

MANAGEMENT OF A MODERN IT COMPANY: theoretical and technological aspects



Edited by
Maksym Yevlanov

MANAGEMENT OF A MODERN IT COMPANY: THEORETICAL AND TECHNOLOGICAL ASPECTS

Collective monograph

Published in November 2025
by Scientific Route OÜ®
Parda tn 4, Kontor 526, Tallinn, Harju maakond Estonia, 10151
www.route.ee

Management of a modern IT company: theoretical and technological aspects

Maksym Yevlanov (Editor)

This book contains information obtained from authentic and highly regarded sources. Reasonable efforts have been made to publish reliable data and information, but the author and publisher cannot assume responsibility for the validity of all materials or the consequences of their use. The authors and publishers have attempted to trace the copyright holders of all material reproduced in this publication and apologize to copyright holders if permission to publish in this form has not been obtained. If any copyright material has not been acknowledged please write and let us know so we may rectify in any future reprint.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made.

The Open Access version of this book, available at monograph.route.ee, has been made available under a Creative Commons Attribution 4.0 International License.

Cover photo: "Tech icons over a green server background" © Canva.com. The cover was created using a Canva's Content License.

Trademark Notice: Product or corporate names may be trademarks or registered trademarks, and are used only for identification and explanation without intent to infringe.

DOI: 10.21303/978-9908-9706-8-4

ISBN 978-9908-9706-8-4 (eBook)

ISBN 978-9908-9706-9-1 (ePub)

This publication has been peer reviewed.

ISBN 978-9908-9706-8-4 (eBook)
ISBN 978-9908-9706-9-1 (ePub)
© The Author(s) of individual chapters, 2025

This is an open access book under the
Creative Commons Attribution
4.0 International License (CC BY 4.0)



AUTHORS

Chapter 1

Viktor Levykin

Doctor of Technical Sciences, Professor
Department of Information Control System
Kharkiv National University of Radio Electronics
ORCID: <https://orcid.org/0000-0002-7929-515X>

Maksym Yevlanov

Doctor of Technical Sciences, Professor
Department of Information Control System
Kharkiv National University of Radio Electronics
ORCID: <https://orcid.org/0000-0002-6703-5166>

Oleksandr Petrychenko

Senior Researcher
Department of Information Control System
Kharkiv National University of Radio Electronics
ORCID: <https://orcid.org/0000-0002-1319-5041>

Olga Neumyvakina

Senior Researcher, Leading Engineer
Department of Information Control System
Kharkiv National University of Radio Electronics
ORCID: <https://orcid.org/0000-0001-6936-6543>

Chapter 2

Maksym Yevlanov

Doctor of Technical Sciences, Professor
Department of Information Control System
Kharkiv National University of Radio Electronics
ORCID: <https://orcid.org/0000-0002-6703-5166>

Viktor Levykin

Doctor of Technical Sciences, Professor
Department of Information Control System
Kharkiv National University of Radio Electronics
ORCID: <https://orcid.org/0000-0002-7929-515X>

Borys Moroz

Doctor of Technical Sciences, Professor
Department of Software Engineering
Dnipro University of Technology
ORCID: <https://orcid.org/0000-0002-5625-0864>

Dmytro Moroz

PhD, Associate Professor
Department of Software Engineering
Dnipro University of Technology
ORCID: <https://orcid.org/0000-0003-2577-3352>

Ivan Iuriev

PhD, Associate Professor
Department of Information Control System
Kharkiv National University of Radio Electronics
ORCID: <https://orcid.org/0000-0002-5178-519X>

Chapter 3

Tetiana Borysenko

PhD, Associate Professor
Department of Information Control System
Kharkiv National University of Radio Electronics
ORCID: <https://orcid.org/0000-0001-6915-6861>

Maksym Yevlanov

Doctor of Technical Sciences, Professor
Department of Information Control System
Kharkiv National University of Radio Electronics
ORCID: <https://orcid.org/0000-0002-6703-5166>

Konstantin Petrov

Doctor of Technical Sciences, Professor,
Head of Department
Department of Information Control System
Kharkiv National University of Radio Electronics
ORCID: <https://orcid.org/0000-0003-1973-711X>

Viktor Borysenko

PhD, Associate Professor
Department of Information Control System
Kharkiv National University of Radio Electronics
ORCID: <https://orcid.org/0000-0001-9372-6449>

Chapter 4

Maksym Yevlanov

Doctor of Technical Sciences, Professor
Department of Information Control System
Kharkiv National University of Radio Electronics
ORCID: <https://orcid.org/0000-0002-6703-5166>

Nataliia Vasylytsova

PhD, Associate Professor
Department of Information Control System
Kharkiv National University of Radio Electronics
ORCID: <https://orcid.org/0000-0002-4043-487X>

Iryna Panforova

PhD, Associate Professor
Department of Information Control System
Kharkiv National University of Radio Electronics
ORCID: <https://orcid.org/0000-0001-7032-9109>

Dmytro Osypchuk

Independent Researcher
ORCID: <https://orcid.org/0009-0005-2808-2145>

Chapter 5

Maksym Yevlanov

Doctor of Technical Sciences, Professor
Department of Information Control System
Kharkiv National University of Radio Electronics
ORCID: <https://orcid.org/0000-0002-6703-5166>

Nataliia Vasylytsova

PhD, Associate Professor
Department of Information Control System
Kharkiv National University of Radio Electronics
ORCID: <https://orcid.org/0000-0002-4043-487X>

Iryna Panforova

PhD, Associate Professor
Department of Information Control System
Kharkiv National University of Radio Electronics
ORCID: <https://orcid.org/0000-0001-7032-9109>

Anastasiia Popova

Independent Researcher
ORCID: <https://orcid.org/0009-0007-4780-7556>

ABSTRACT

The monograph is devoted to solving scientific and applied problems that arise in the process of managing modern IT companies. The rapid development of IT products in the 21st century has caused significant changes in the processes and management system of an IT company as a commercial enterprise providing services. These changes have led to the perception of a modern IT company as an enterprise that combines the most successful practices, models and methods of planned and project management in its activities. But this view of an IT company is detailed only at the conceptual level. The issues of formal description, design and development of systems and technologies that would be based on this concept remain largely unresolved. The vast majority of modern IT products used to manage an IT company and its processes are focused on the concept of exclusively project management and do not take into account the features of planned practices of modern service management.

The monograph considers the solution of theoretical and theoretical-applied issues related to the formulation and solution of important tasks of managing the general activities of an IT company and its individual processes. Such tasks include the task of managing the operation of web-based information systems, the task of assigning tasks to IT company employees, as well as the task of assessing the costs of IT company resources during its activities. The research results are formal models, methods, and elements of information technology that can be used to build management systems for modern IT companies. The research used mathematical tools of category theory and set theory; models and methods of statistical analysis and Data Mining, as well as existing models and methods of assessing the costs of efforts and resources in IT projects.

The obtained research results and their experimental verification indicate the possibility of improving the processes of modern IT companies. Such improvement can be carried out through the consistent development of individual elements of the IT company management system, without global reengineering of systems and technologies for storing and processing data currently used in the IT company.

This monograph can be useful to researchers, teachers, graduate students and higher education students in the field of computer science, information management systems and IT project management.

Keywords

IT project, information system, operations management, description of a system element, incomplete information, task, performer, evaluation, localization, video game, change, Beckhard and Harris model, descriptor, sprint.

CIRCLE OF READERS AND SCOPE OF APPLICATION

The first chapter presents the results of the development of elements of the methodology, models and formal formulation of the problem of effective management of the life cycle of the operation of a web-based information system (IS). These results can be used to build an IS of multi-level management of IT services provided by an IT company. Such use is appropriate when developing a strategy for creating and gradually improving an IT company management system. The materials of the chapter will be useful to scientists in the fields of computer science, information systems, as well as management of the activities of an IT company and its individual IT projects.

The second chapter presents the results of solving the problem of modeling losses when designing a description of an IS element in conditions of incompleteness or lack of information about this element. These results can be used in the process of managing IT project decisions in conditions of partial uncertainty. Such use is appropriate for automated assessment of possible costs in the process of forming and changing task backlogs for IT project performers. The materials of the chapter will be useful to scientists and specialists in IT project management of the development of IS management of enterprises and organizations.

The third chapter proposes the results of solving the problem of assigning IT project tasks to its performers. These results can be used to automatically assign IT project tasks to its performers, taking into account the previous work experience of these performers. Such use is appropriate for creating services that can interact with both existing automated management tools for an IT company and those being developed. The materials of the chapter will be useful to scientists and practitioners in the fields of IT project team management and the use of artificial intelligence tools for managing IT projects.

The fourth chapter proposes the results of solving the problem of assessing the complexity of an IT project for video game localization. These results can be used when planning IT projects for video game localization. Such use is appropriate as a method of automated assessment of efforts to implement an IT project for video game localization. The materials of the chapter will be useful to scientists and practitioners in the fields of IT project management for the localization of various computer games.

The fifth chapter proposes the results of solving the problem of quantitative assessment of changes that occur during the implementation of a long-term IT project. These results can be used to manage changes that occur in IT projects that last more than a year. Such use is appropriate for automated assessment of time spent on implementing changes during individual iterations of an IT project. The materials of the chapter will be useful to scientists and practitioners in managing teams of performers and changes in IT projects.

CONTENTS

List of Tables.....	x
List of Figures	xii
Introduction.....	1
CHAPTER 1 Basics of the effective management methodology of web-based information systems	4
1.1 Introduction	5
1.2 Analysis of modern research in the field of effective management of web-based information systems.....	7
1.3 Research concepts and assumptions	11
1.4 Results of the development of the elements of the methodology for effective operation management of a web-based information system	16
1.4.1 Results of the development of the concept and ontology for effective operation management of a web-based information system	16
1.4.2 Results of development of a set of theoretical and categorical models of effective operation of a web-based information system	20
1.4.3 Results of the development of a formal statement of the problem of effective management of the operation of a web-based information system.....	25
1.5 Results of the development of a theoretical and categorical model of the property of an information system	28
1.6 Discussion of the research results	36
1.7 Conclusions	39
References	41
CHAPTER 2 Cost management of an information system IT project in the conditions of incomplete information about its elements	45
2.1 Introduction	46
2.2 Analysis of current research in the field of IT project decision management	47
2.3 Research models	50

2.4	Results of adapting the causal scheme of production losses to the features of the design processes of an information system element.....	54
2.5	Results of developing detailed models of losses from lack of information	57
2.6	Experimental verification of the obtained results	66
2.6.1	Description of the initial data.....	66
2.6.2	Description of the assessment of losses due to the replacement of incomplete information with information about a similar element of the information system at the stage of IT project initiation.....	70
2.7	Discussion of the results of the loss assessment for the considered solution options	79
2.8	Conclusions	82
	References	83

CHAPTER 3 Method of automatic assignment of tasks to IT project performers ...86

3.1	Introduction	87
3.2	Overview of the current state of solving the problem of automated planning of IT projects.....	88
3.3	Materials and methods	94
3.4	Results of developing a method for solving the problem of assigning IT project tasks to its performers	100
3.4.1	Results of adapting the polynomial naive Bayes classifier to the specifics of the problem of assigning IT project tasks to its performers	100
3.4.2	Description of the results of developing an algorithm for implementing the developed method.....	104
3.5	Experimental verification of the improved method	111
3.6	Discussion of the research results	120
3.7	Conclusions	122
	References	124

CHAPTER 4 Method for assessing the complexity of an IT project for video game localization 128

4.1	Introduction	129
4.2	Analysis of current research in the field of software and video game localization.....	130
4.3	Basic research method.....	136

- 4.4 Results of the development of a method for assessing the complexity of an IT project for video game localization 140
 - 4.4.1 Identification of factors affecting the complexity of video game localization work..... 140
 - 4.4.2 Results of determining the content of the stages and activities of the method for assessing the complexity of an IT project for video game localization 146
- 4.5 Experimental verification of the obtained results 153
- 4.6 Discussion of the results of the development of a method for assessing the complexity of an IT project for video game localization... 159
- 4.7 Conclusions 166
- References 167

CHAPTER 5 The task of quantitative assessment of changes in the long-term IT project management system..... 170

- 5.1 Introduction 171
- 5.2 Analysis of the approach and modern methods of project change management 172
- 5.3 Materials and methods 179
- 5.4 Results of improving the method of quantitative assessment of changes in a long-term IT project 181
 - 5.4.1 Classification of descriptors that describe tasks and changes in an IT project..... 181
 - 5.4.2 Development of an improved method for quantitative assessment of changes 182
- 5.5 Elements of information technology for automated solution of the problem of quantitative assessment of changes 188
- 5.6 Description of an example of solving the problem of quantitative assessment of changes during the implementation of a long-term IT project 194
 - 5.6.1 Description of the features of the long-term IT project "Web Constructor" 194
 - 5.6.2 Overview of the causes of changes in the long-term "Web Constructor" IT project..... 196
 - 5.6.3 Description of the progress and results of experimental verification of the obtained results 200
- 5.7 Discussion of the research results 208
- 5.8 Conclusions 210
- References 211

LIST OF TABLES

1.1	Definition of the "operated information system" concept at different levels of representation	14
2.1	Results of the search for functional modules and individual functions of KSASK suifor reuse in the IT project for the development of the "Electronic CMTPL Policy" service	69
2.2	Values of the parameters of the simplified functional point method and the rollback coefficient k and the COCOMO model	71
2.3	Results of calculating the characteristics of the IT project for developing the "Electronic CMTPL Policy" service "from scratch"	71
2.4	Adjusted results of calculations of the characteristics of the IT project of the development the "Electronic CMTPL Policy" service "from scratch"	72
2.5	Comparison of the names of the functions of the "Electronic CMTPL Policy" service and the "Online Store" module	73
2.6	An example of the result of using the Porter stemmer for function names	75
2.7	Results of calculations of $FP_{Add,z}$ values for all functions, the reuse of which is proposed in the IT project	76
2.8	Values of the elements of formula (2.47), selected for the quantitative assessment of possible losses	76
2.9	Results of the calculations of the $q_z(T)$ values for all functions which reuse is proposed in the IT project	76
2.10	Values of the elements of formula (2.24), selected for quantitative assessment of possible losses	77
2.11	The values of the elements of formula (2.50), selected for quantitative assessment of possible losses	78
2.12	The values of the elements of formula (2.20) selected for the calculation of the estimates of the values of the replaced information	78
2.13	Results of the comparison of the characteristics of the proposed solution options for the IT project	80
3.1	Characteristics of performers	112
3.2	List of completed tasks and their performers	113
3.3	Number of occurrences of tokens in the names of the tasks of each of the performers	113
3.4	List of IT project tasks, ordered by priority values, planned for current performing	115

List of Tables

3.5	Parameters for calculating the value of the classification rule (3.7)	116
3.6	Results of the calculations of the posterior probability for each task and performer	117
3.7	Results of applying logarithmic transformation of posterior probability calculations for each task and performer	118
3.8	Task distribution results	119
3.9	Results of using the developed method	120
4.1	Comparison of translation speed indicators proposed in [16–19]	137
4.2	Results of the comparison of the complexity factors, taken into account by variations of the method based on the number of words	138
4.3	Developed scale for assessing the complexity of the translation of translation units	148
4.4	Characteristics of localization files	155
4.5	Localization file characteristics	155
4.6	Predicted and actual translation speed and duration	156
4.7	Characteristics of the localization files of the video game "Sid Meier's Civilization® VI"	158
4.8	Results of comparing the predicted and real values of translation speeds and durations (additional stage)	158
4.9	Results of the comparison of the errors of the developed method and existing methods for examples of localization files	162
4.10	Results of comparing the accuracy of the forecasts of the duration of the IT project for video game localization and the actual duration of this IT project	164
5.1	Comparison of existing methods of change management in projects	175
5.2	Additional classification of the "Web Constructor" project	194
5.3	Assessment of employees' attitude to the method based on the Beckhard and Harris's model	201
5.4	Descriptors for tasks performed during the last five sprints	202
5.5	Descriptors for tasks performed during the sixth sprint	203
5.6	Performance indicators for the change model	205
5.7	Calculation of the change implementation time required for the WEBCO_2360 task due to changes in the number of characteristics	205
5.8	Calculation of the ratio of the predicted value of time to the actual value of time	206
5.9	Models calculated using Huber regression and their fitness indicators	207

5.10	Calculation of the estimate of released days under the condition of changing the number of characteristics of the task WEBCO_2361	208
5.11	Calculation of the ratio of the predicted value to the actual value for task WEBCO_2361	208
5.12	Results of comparing the values of indicators of the change assessment process	209

LIST OF FIGURES

1.1	Description of the architecture of a web-based system	6
1.2	Aspect-process representation of the System Lifecycle Management approach	12
1.3	Diagram of the relationships between the main concepts of the ontology of the methodology for managing IT services of web-based IS	18
1.4	Diagram of relationships between elements of a generalized model of effective management of the operation of a web-based information system	24
1.5	Structure diagram of the theoretical-categorical model L_{Prop} which describes the concept of "Property"	35
2.1	Cause-and-effect scheme of production losses due to lack of information	51
3.1	Scheme of relationships between initiatives, epics and tasks of an IT project	96
3.2	UML activity diagram describing the algorithm for solving the classification problem using a polynomial Bayesian classifier	99
3.3	UML activity diagram describing a modified algorithm for solving a classification problem using an adapted polynomial Bayesian classifier	103
3.4	UML activity diagram describing the algorithm for solving the problem of assigning IT project tasks to its performers	105
4.1	Diagram of the extended video game value chain	131
4.2	Structure of an indie game development company based on field research in Sweden and China	132
4.3	Classification scheme of the main factors that influence the progress and results of the assessment of the complexity of an IT project for video game localization	141
4.4	Example of a Translation Unit in the original language from the "Don't Starve Together" game	154
4.5	Results of comparison of relative errors	163
4.6	Results of comparison of squares of relative errors	163
5.1	Stages and key activities of the improved quantitative change assessment method	183
5.2	Description of the architecture of the information technology of quantitative change assessment	189
5.3	A fragment of the program code for calculating the values of the Gaussian density function	190

5.4	A fragment of the program code for plotting the calculated Gaussian density function and checking the results obtained using the three-sigma method	191
5.5	A fragment of the program code for creating a model using the least squares method	192
5.6	A fragment of the program code for creating a model using the Huber regression method	192
5.7	A fragment of the program code for calculating the performance indicator of the model "Determination coefficient R^2 "	192
5.8	A fragment of the program code for calculating the performance indicator of the model "Forecasting coefficient Q^2 "	192
5.9	A fragment of the program code for calculating the performance indicator of the model "Forecasting coefficient Q_{LOOCV}^2 "	193
5.10	A fragment of the program code for calculating the performance indicator of the model "Standard deviation (σ)"	193
5.11	A fragment of the program code for calculating the performance indicator of the model "F-criterion"	193
5.12	Gaussian distribution for the data on the number of changed characteristics collected for the first five sprints	204
5.13	Change model built using the OLS method for data from the first five sprints	204
5.14	Gaussian distribution for descriptor values collected during the previous six sprints	206
5.15	Model built using Huber regression, for data from the first six sprints	207

INTRODUCTION

A modern IT company is an enterprise that stands out among all other types of enterprises. The main goal of an IT company is to provide services for the development, implementation, support, maintenance and development of IT products for various purposes on behalf of various stakeholders. Therefore, an IT company is forced to combine the best practices of planned management of IT services and project adaptation of these IT services to the requirements of individual customers during its own activities. Such a combination requires, in turn, the creation and development of an appropriate methodology, models, methods, systems and technologies for managing an IT company and its individual processes.

Unfortunately, the current state of science allows to describe in detail the methodological issues of creating and using an IT company management system and its processes mainly at the conceptual level. A significant number of scientific studies in recent years are aimed at solving only individual issues of managing the activities of an IT company. This can be explained by the complexity of the problem of building a methodological and formal basis for such a management system. Such a lag is also observed in the field of development and use of specialized information systems and technologies for managing an IT company. The vast majority of modern IT products used to manage an IT company and its processes are focused on the concept of exclusively project management and do not take into account the peculiarities of planned practices of modern service management. Some work on the creation of information systems and technologies for multi-level management of IT services was started in the early 2010s. But still the main results of these works are too large IT products, the cost and complexity of which significantly limits their distribution.

Solving the considered problems of managing a modern IT company requires a large amount of scientific and applied work that cannot be performed by a single team of researchers. But the gradual solution of scientific and applied issues, coordinated with the modern concept of IT company management and with the results obtained by individual research groups, can provide the governing bodies of IT companies with results that have significant applied value. Therefore, conducting research in this area is relevant from a theoretical and applied point of view.

The monograph offered to your attention contains the results of research by employees of the Department of Information Control Systems of the Kharkiv National University of Radio Electronics, the Department of Computer Systems Software of the National Technical University "Dnipro Polytechnic" and individual

independent researchers working in the field of management of IT companies and individual IT projects. This monograph considers the solution of theoretical and theoretical-applied issues related to the formulation and resolution of important problems and tasks of managing the general activities of an IT company and its individual processes, namely:

- problems of effective management of the operation of web-based information systems (Chapter 1);
- problems of managing the costs of an IT project of an information system in conditions of incomplete information about its elements (Chapter 2);
- tasks of assigning tasks to IT project performers (Chapter 3);
- tasks of assessing the complexity of an IT project for video game localization (Chapter 4);
- tasks of quantitative assessment of changes in the long-term IT project management system (Chapter 5).

During the research described in the chapters of the monograph, mathematical tools of category theory and set theory, models and methods of statistical analysis and Data Mining, as well as existing models and methods for estimating the costs of efforts and resources in IT projects were used.

The main theoretical results of the research presented in this monograph are:

- the concept of the problem of effective management of the life cycle of an operated information system as an IT service system;
- a set of models for effective management of the operation of a web-based information system;
- a formal description of the problem of effective management of the life cycle of an operated information system;
- a theoretical and categorical model of the properties of an information system;
- detailed models of losses in the activities of an IT company due to the lack of information;
- a method for solving the problem of assigning IT project tasks to its performers;
- a new method for assessing the complexity of IT projects for video game localization;
- an improved method based on the Beckhard and Harris model for quantitative assessment of changes in a long-term IT project.

The obtained theoretical results were verified during the implementation of relevant IT projects or on data from IT projects implemented using other methods and techniques. The results of the tests confirm that the application of the models and methods proposed by the authors of the monograph is better than the use of existing models, methods and tools for solving similar problems of IT company management.

Introduction

This monograph may be useful to researchers, teachers, postgraduate students and higher education students in the field of computer science, information management systems and IT project management.

The authors express special gratitude to the first reader of the monograph – PhD, Leading Researcher of the Department of Information Control Systems of the Kharkiv National University of Radio Electronics, Olha Neumyvakina – for valuable recommendations, attention, and versatile erudition, repeatedly revealed during the creation of the monograph.

CHAPTER 1

Basics of the effective management methodology of web-based information systems

Viktor Levykin
Maksym Yevlanov
Oleksandr Petrychenko
Olga Neumyvakina

Abstract

The object of the study is the processes of functioning and maintenance, which together determine the stage of operation of a web-based information system for managing an enterprise or organization.

During the study, the problem of developing elements of the methodology for lifecycle management of a web-based information system was solved. Modern research in this area is mainly aimed at analyzing and developing individual aspects of managing the operation of complex systems. The main attention is paid to solving individual tasks of managing the operation of systems based on IoT or DevOps. General issues of creating and improving the System Lifecycle Management approach remain little studied.

During the study, the main provisions of the effective management concept of the life cycle of a web-based information system were proposed. This concept is based on a multi-level representation of the information system and the possibility of effective management of the system according to its properties. For further formal description of the proposed concept, the main concepts were defined and the ontology of this concept was developed. Based on this ontology, a generalized model of effective management of the operation of a web-based information system and a set of theoretical and categorical models that detail the description of the elements of this model were developed.

The resulting generalized model was taken as a basis for developing a formal problem statement of effective life cycle management of an operated information system as an IT service system. A formal description of the objective function and constraints of the problem of classical (permanent) life cycle management of an

operated information system was proposed. On its basis, a formal description of the objective function and constraints of the problem of effective life cycle management of an operated web-based information system was proposed. It was determined that the problem of classical (permanent) life cycle management of an operated information system is a special case of the proposed problem of effective management.

The practical application of the obtained research results allows improving systems for life cycle management of an operated information system without global reengineering of existing systems and technologies for data storage and processing.

Keywords

Web-based information system, system property, operation management, ontology of effective management, theoretical-categorical model, attributive model.

1.1 Introduction

The first quarter of the 21st century is characterized by the rapid development of web-based information systems (IS) for managing enterprises and organizations. The term "web-based IS" will be used here and further to refer to IS that uses web technologies to provide information and services to users or other information systems/applications [1]. One of the most important consequences of this definition is the perception of web-based IS as systems, which functions can be accessed from anywhere in the world (provided there is a working Internet connection) via a browser.

The modern view of the architecture of web-based IS recognizes the existence of several different approaches to describing this architecture. However, to reflect the above-mentioned perception of web-based IS, it is best to use the description of the architecture of such a system proposed in [2] (**Fig. 1.1**).

In **Fig. 1.1**, the following designations are adopted for the elements of the description of the architecture of web-based IS: DNS – Domaine Name System, a domain name system that ensures the operation of the Internet; CDN – Content Delivery System, a content delivery system that provides acceleration of the delivery of static HTML, CSS, JavaScript files and images (dotted lines separate the elements of the web-based system that are located on the system's web servers and can be located practically anywhere on the globe). Detailed descriptions of these elements are given in [2].

Creation, support and development of web-based IS with the architecture described in **Fig. 1.1** significantly complicates the solution of the problem of managing such IS within their life cycle (LC). Such management is necessary for any IT company that promotes web-based IS of its own production on the market of IT products

intended for the management of enterprises and organizations. The goal of such management in general should be to increase the competitiveness of web-based IS by maintaining the characteristics of efficiency and quality of this system at the level most desirable for all stakeholders.

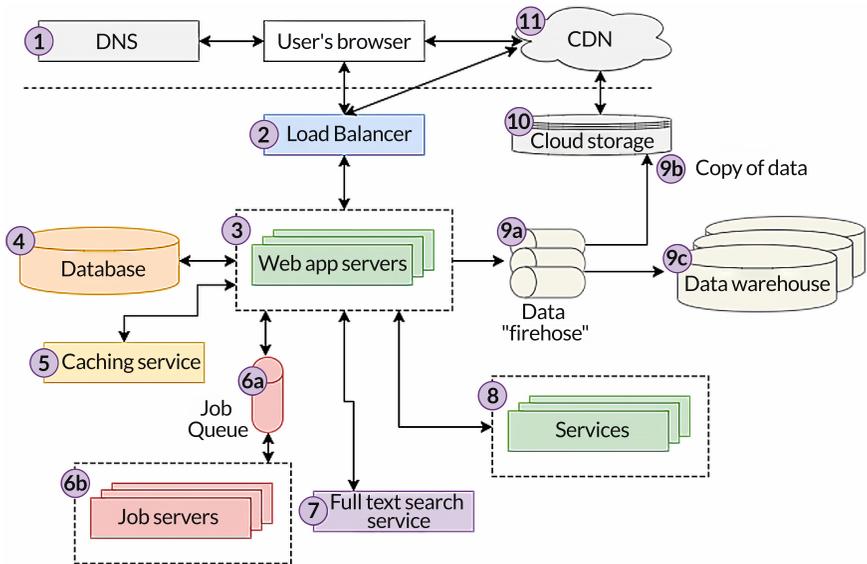


Fig. 1.1 Description of the architecture of a web-based system
Source: [2]

The modern approach to the creation and functioning of IS management of enterprises and organizations conceptually considers such systems as a type of system for providing IT services to operators and users [3]. Here and further, the term "service" will be understood as the performance of actions, work or duties, which is self-sufficient, consistent, discrete and can be composed of other services [4]. Based on this definition, the term "IT service" was proposed to be used to refer to an independent functional task in IS, the use of which is economically and technically feasible for performing a separate activity of the process of an enterprise or organization or for managing this activity [5].

The results of implementing this approach in the IT industry are [6]:

- the concept of an IT service management system as a system for managing and controlling the activities of managing IT objects of an IT company;

- an approach "Product Lifecycle Management" (PLM);
- an approach "Application Lifecycle Management" (ALM).

The term "PLM" will be understood as an approach to managing processes and methods that are used or created during the LC of products or systems. Its scope mainly covers the hardware parts of the system [7]. The term "ALM" will be understood as an approach focused on the software LC. It is designed to monitor, control and manage artifacts and processes that arise during the program LC. These processes include requirements development, source code management, test management, release management, etc. [8].

To solve the problem of managing services during their LC, a separate type of management system was created – service management systems. According to the modern understanding, such systems should be perceived as systems for managing and controlling the activities of the service provider from the point of view of service management [3]. However, although the PLM and ALM approaches arose on the basis of the same concept of an IT service management system, the ISs created within these approaches developed in different ways. In order to eliminate the difference between these ISs, a new approach was developed, the main difference of which is the integration of PLM and ALM into the interdisciplinary management of the system's LC (SLM) [9]. Unfortunately, the creation and development of the SLM approach occurs mainly at the conceptual level. The reasons for this should be considered different approaches to the structuring of hardware and software, as well as to the integration of elements of this software into systems. In addition, the phases of the software and hardware LCs also differ, even if there are similarities [10]. Therefore, conducting theoretical and theoretical-applied research devoted to the development of methodological foundations of the SLM approach at the conceptual and formal levels is relevant.

1.2 Analysis of modern research in the field of effective management of web-based information systems

Modern information technologies (IT) and IS based on PLM are aimed at managing large amounts of information and artifacts that arise and accumulate during the product LC [7]. Such PLM-IT and PLM-IS are considered as a center that ensures the existence of compatible data flows (for example, data from CAD products, modeling or architecture data, etc.) and solving the problems of managing these data flows. This is due to the fact that PLM grew out of CAD (Computer Aided Design) and PDM (Product Data Management). For this reason, PLM-IT and PLM-IS are only limitedly suitable for displaying and supporting the software LC [9].

The use of PLM-IT and PLM-IS for managing the IS LC in enterprises and organizations is complicated by the need to manage changes that arise during the creation and operation of such IS. Thus, in [11] the reasons for these complications are the lack of compatibility of product change notifications, low data quality, and the lack of direct traceability and tracking of changes to the final product. To eliminate these complications, [11] proposes to integrate PLM-IS and IT asset administration (Asset Administration Shell). Such integration should improve cooperation between PLM-IS and IS of enterprise and organization management, improve data quality, ensure traceability and increase efficiency within engineering processes. However, this approach is not very suitable for LC management, because it focuses mainly on hardware management.

Unlike PLM, ALM was originally considered as an approach that can theoretically be used without the use of appropriate tools. ALM was mainly proposed to be used to integrate all processes, methods and data used and generated within software development [8]. The modern vision distinguishes three main aspects within ALM: governance, development and operations [9]. The governance aspect considers the tasks of conducting software business analysis, project portfolio management and program portfolio management. The development aspect considers the tasks of software generation (from requirements definition through design and source code development to testing and release). The operation aspect considers the tasks of software monitoring and management [12].

Modern research suggests understanding SLM as a technical and organizational basis for LC management of all system artifacts that are created or improved during the existence of this system, and the tools used to work with these artifacts. This basis provides clear tracking of all system elements. Thus, interdisciplinary SLM is considered the basis for complex digital engineering [13]. But IT products that are the results of this complex digital engineering differ significantly from each other. Examples of such IT products include:

- Microsoft products (Microsoft System Center 2016, Microsoft Azure Automation & Control in Microsoft Operations Management Suite);
- specialized products of individual IT companies (Cortex, ActiveBatch Workload Automation, Ignio, VMware Vrealize Orchestrator, etc.);
- products that are descendants of CASE systems and visual modeling environments (CA Process Automation).

As can be seen, the main difference between IT products designed to implement the SLM approach is caused mainly by the features and requirements of specific IT companies, on which orders these products were created. It should be noted that there is no fixed classification of IT products that can be used to implement the SLM approach.

But it should be recognized that these and similar IT products generally meet the requirements for service management systems defined in [3]. In general, the set of functions of such products is aimed at achieving the following goals:

- increasing the efficiency of the functioning of IT processes of the enterprise-consumer of IT services (technical and economic goal);
- reducing the operational risks of the enterprise-consumer of IT services (technical and economic goal);
- reducing the costs of IT processes of the enterprise-consumer of IT services throughout the entire LC (economic goal);
- compensation for the negative consequences of the complexity of managing the IT services system of the consumer enterprise, in particular, eliminating the effect of "IT blindness" (economic goal);
- increasing the level of compliance with standards, including to facilitate further certification (economic and organizational and methodological goal).

Unlike applied developments, most theoretical research in the field of PLM, ALM and SML is aimed at developing individual aspects of IT service management of large industrial systems based on the Internet of Things. For example, in [14] a description of the information layer of the reference architectural model RAMI 4.0 is proposed, which is supported by the Zachman Framework concept. The proposed description, according to the authors, should ensure the practical use of RAMI 4.0, allowing stakeholders to use model-based systems engineering, on the one hand, and include information engineering, on the other. However, the description proposed in [14] can be used to develop only individual aspects of industrial systems and does not cover the task of industrial IS management as a whole. Similar research is being conducted in the IT industry. Thus, in [15] the results of mixed modeling and modeling of continuous service delivery pipeline scenarios as a separate aspect of DevOps are described. But these studies also do not cover the overall task of managing IS and IT, which an IT company uses in its activities.

The formation of integral assessments of the progress and results of the functioning of individual IS services currently does not involve the use of formal models [16]. At the same time, studies have shown that the success of projects aimed at improving individual processes is not a coincidence, but a predictable result with clearly defined and measurable characteristics [17]. To solve various IT service management problems, approaches based on solving multi-criteria optimization problems are proposed in [18]. However, the difficulties of applying such approaches in managing real web-based IS force to look for other options for solving such problems. Thus, in [19] it is proposed to consider an approach to web service management based on knowledge. However, this approach is not yet fully developed and requires additional research.

A significant number of studies consider the problem of managing the IS LC mainly through the prism of tools for its solution. In this case, the multi-criteria decision-making process is usually proposed as the main tool. To implement this process, decision support systems based on multidimensional data warehouses are proposed. An example of such a platform designed to automate the management of the stages of the LC describing the development of modern software is considered in [20]. A significant part of the publications is devoted to describing the results of research into problems that arise when using tools for solving such multi-criteria optimization problems. Algorithms [21] and tools [22] for analyzing and optimizing hyperparameters are considered as similar tools in modern research. Studies devoted to the use of simulation modeling tools for LC management of individual aspects of IS [23] are not left without attention.

The following directions for further research on the development of SLM are indicated in [9]:

- improving the implementation of the knowledge management process in the business (in particular, ensuring the interconnection of business goals and indicators with current processes to identify awareness of complaints and the potential for business improvement [24]);
- developing an integrated or common ALM-PLM data model (in particular, to identify and minimize differences between the descriptions of the hardware and software development processes [25]);
- extending the requirements and use cases of the ALM-PLM system to other stages of the Life Cycle;
- further detailing and refining the SLM, PLM and ALM approaches, developing these approaches through new methods, ideas and processes to support the development of software and hardware;
- defining KPIs for ALM-PLM integration (in particular, to further evaluate and control the use of these approaches to assess their success or potential for improvement [26]).

But the SLM development is impossible without eliminating an important contradiction in the field of IT service management. The essence of this contradiction is as follows. A typical IT service management system, defined in [1], recognizes the goal of management as maintaining user satisfaction with existing IT services at the level desired by the Provider and Consumer of IT services. At the same time, the main quantitative criterion by which it is possible to determine the degree of achievement of this goal is the indicator of the number of incident reports that remain unresolved. But these incident reports arise as a result of the interaction of operators and users with existing IT services and the IT infrastructure that ensures the operation of these

IT services. Thus, the IT service management system defined in [3] and the IT products that are created on the basis of the requirements for this system are aimed at maintaining stable and time-invariant states of the software and hardware of the IT service system. Therefore, compliance with the requirements for a typical service management system defined in [3] does not allow optimizing the set of IT services taking into account the technical and economic features of changing business processes of individual consumers of IT services. Therefore, the study of the general formal formulation of the problem of managing the IS LC is relevant both from a theoretical and practical point of view.

The aim of the study is to develop the foundations of a methodology for effective management of the LC of an operated web-based IS as an IT service system taking into account the technical and economic features of business processes subject to automation. This will allow automating the solution of the tasks of managing the IS LC for enterprises and organizations, taking into account the features of SLM and ALM-PLM integration.

To achieve this aim, it is proposed to solve the following objectives:

- develop a concept for effective operation management of web-based IS;
- develop a set of models for effective operation of web-based IS;
- develop a formal statement of the problem of effective operation management of web-based IS;
- develop a model of the properties of the operated web-based IS.

1.3 Research concepts and assumptions

By analogy with the modern vision of ALM, the SLM approach in this study is proposed to be considered as a set of three main aspects: system governance, system development, and system operations. However, unlike the tasks considered in [9] and [12], which comprise the ALM aspects, it is proposed to present each SLM aspect as a set of typical processes of the system's LC, defined in [4]. Such a representation is shown in **Fig. 1.2**. The use of typical processes allowed, instead of defining individual management tasks characteristic of each individual aspect of the approach, to establish a set of works and activities, the implementation of which is most important for a specific aspect.

Based on the proposed aspect-process representation of the SLM approach, a conclusion was made about the significant differences in the methodologies for managing web-based IS under development and web-based IS under operation. Based on this conclusion, a decision was made on the feasibility of conducting research in

the field of developing the main elements of the methodology for managing the operation of web-based IS. This decision was also a consequence of the following reasons:

- the minimum (compared to other aspects) number of typical LC processes of the system, which in this aspect are the most important;
- recognition by modern IT companies of LC processes, which describe the operation of IT products, as critical business processes, the activities of which bring the main percentage of the IT company's profit;
- unlike the aspects of system management and development, the aspect of system operation remains the least studied from a theoretical and applied point of view.

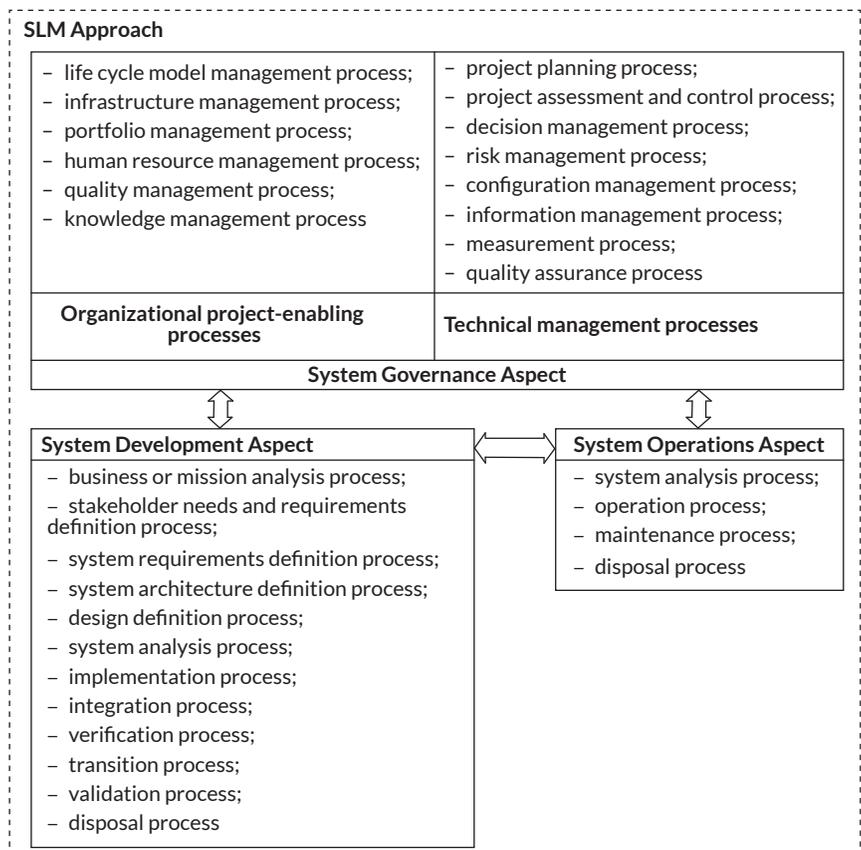


Fig. 1.2 Aspect-process representation of the System Lifecycle Management approach

Therefore, the object of this study was chosen to be the processes of functioning and maintenance of IS, which together determine the stage of operation of the IS LC. The subject of this study is the main methodological elements that allow formalizing the formulation and solution of the problem of managing the operation of web-based IS.

The development of the main elements of the methodology for managing the operation of web-based IS was proposed to be carried out on the basis of the following initial assumptions and recommendations [27]:

- Assumption 1: the operated IS should be considered as a set of interconnected representations at different levels (business level, IT service level, IT service level, IT infrastructure level);

- Assumption 2: the main conditional "unit" of describing data processing actions in managed IS of any level of representation is a transaction;

- Assumption 3: the dimensionality of the problem of managing the functioning of the IS can be reduced provided that individual properties of the IS are identified, which characterize the manifestations of individual qualities of this system;

- Assumption 4: each individual property of the IS is determined by a set of requirements (functional and non-functional) and a set of requests for change (requests for change, RFC) of these requirements and individual elements of the system.

Assumption 1 arose from the need to eliminate one of the main shortcomings of the existing methodology for managing the operation of the system as a set of IT services. This methodology uses one level of representation to describe any IS – as a set of individual IT services and their heterogeneous components [3] without taking into account their belonging to the Supplier or Consumer of IT services. But for the Consumer and the Provider of IT services, the description of any IS implies the simultaneous existence of several levels of representation – as a single product, as a set of functions, and as a set of IT services that implement these functions and are at different stages of their LC. Therefore, to eliminate this contradiction, first of all, it is necessary to coordinate the descriptions of web-based IS at different levels of representation of this system. Similar levels were identified in [5] for the processes of forming and analyzing system requirements. Based on the solution proposed in [5], it is proposed to identify the following levels of representation of the operated IS:

- business level (combining the level of managed objects and/or processes proposed in [5] and the general system level);

- IT accommodation (or functional service) level;

- IT service level;

- IT infrastructure level.

The features of the representations of the concept of "operated IS" at each of the proposed levels are given in **Table 1.1** [27].

Table 1.1 Definition of the "operated information system" concept at different levels of representation

Presentation layer	Definition of the concept of "operated information system"
Business layer	IS – IT product, which: <ul style="list-style-type: none"> – has financial and legal value; – is one of the mechanisms of the managed object and/or process; – is simultaneously a system consisting of personnel and a complex of automation tools; – forms and reflects a single holistic information representation of this object and/or process in accordance with the goals set for this product
IT accommodation (or functional service) level	IS – a system of IT services (functional tasks, functions), which: <ul style="list-style-type: none"> – is operated on the basis of the Consumer's decisions on the economic or technical feasibility of this action; – is aimed at forming and reflecting a single holistic information representation of the managed object or process in accordance with the goals set for the system
IT service layer	IS – a system of IT services, which: <ul style="list-style-type: none"> – implements a set of operations for providing or processing data; – is aimed at forming and reflecting a single holistic information representation of the object or process in accordance with the goals set for the system
IT infrastructure layer	IS – a system of interconnected IT infrastructure elements necessary and sufficient for the operation of all IT services of IS, which: <ul style="list-style-type: none"> – consists of complexes of automation tools; – is aimed at the formation and display of a single holistic information representation of an object or process in accordance with the goals set for the system

Source: [27]

To harmonize the IS descriptions at the levels of representation considered in **Table 1.1**, it is necessary to introduce an additional concept that allows describing the behavior of IS and its elements during operation. For this purpose, Assumption 2 was formulated, according to which it was proposed to use the term "transaction" as such a concept. Each abstract transaction should have the following properties [28]:

- atomicity;
- correctness;
- isolation;
- durability.

These properties allow considering a transaction as the basis for a formal description of any possible action performed by a web-based IS and its individual elements during operation.

To describe the operated web-based IS and its elements at the levels of representation considered in **Table 1.1** levels of representation, it was proposed to distinguish the following types of transactions [27]:

- business transaction - an agreed change in the state of relations between two or more parties, where each party is ready for this change and knows that it will be agreed and approved by all parties [29];
- functional transaction - a logical unit of the scenario for performing an IS function (service), usually including several service transactions;
- service transaction (also transaction) - a logical unit of work, usually including several database operations [28];
- infrastructure transaction - a logical unit of work at the upper (application) levels of the network models used [30], usually including several operations of the lower levels of a specific network model.

Thus, any IS (including web-based IS) can be described at any level of representation using transactions that transfer the IS and its elements from the initial state to the final state. At the same time, to unify the descriptions of heterogeneous IS elements during its operation, it was proposed to use the concept of "configuration item" (CI) introduced in [3] - an element that must be controlled to provide one or more services.

However, such a representation of the operated IS seriously complicates its modeling during operation management. The application of the existing methodology for the operation management of IS leads to the fact that the growth rate of the number of indicators characterizing the IS and its CIs significantly exceeds the growth rate of the number of CIs in the IS. The consequence of this is the impossibility of scaling models and methods for solving management problems during the operation management of small, medium and large IS due to the sharp increase in the dimensionality of such problems. It should be noted that existing methods for solving large-dimensional problems are effective only in some cases and do not solve this problem radically.

To overcome this complication, it was proposed to apply elements of the theory of potential efficiency. This theory allows to assess the efficiency of a complex system by using relatively simple models that describe individual qualities of this system. In this case, the efficiency of the system is represented as the probability of a beneficial exchange between the system and the environment [31]. This definition does not contradict the definition of the concept of efficiency adopted in the existing methodology and allows to present the degree of implementation of planned activities and achievement of planned results in service management as the probability of a profitable exchange of CIs and resources necessary for their creation or modification between the IS, its Supplier and Consumer.

The proposed approach to organizing a formal solution to IS operation management tasks requires the introduction of an additional concept of "system property". At the same time, for solving IS operation management tasks, only those system properties that characterize the manifestations of individual qualities of this system are of interest. IS qualities, in turn, are determined by a set of requirements (functional and non-functional) put forward to the IS before the start of its operation, as well as by a set of RFCs of the IS and its individual CIs that are formed during the operation of the IS. In this case, the effectiveness of the operated IS will be determined as the degree of achievement of planned results for each individual IS property and its CIs [27].

The use of the theory of potential efficiency and Assumptions 3 and 4 formulated on its basis allowed to determine the main approach to the formal description of the operated web-based IS, its CIs, as well as the formulation of the problem of LC management of the operated web-based IS. To detail this approach, it was proposed to use the concept of "Digital Twin". This concept proposes to use their electronic representation – a digital analogue that is actually no different from such an object – to solve the problems of managing complex real-world objects [32]. Although the concept of "Digital Twin" is used mainly for managing material objects (aircraft, cars, power plants, industrial robots, etc.), a number of modern studies recognize the expediency of applying this concept in software engineering [33].

1.4 Results of the development of the elements of the methodology for effective operation management of a web-based information system

1.4.1 Results of the development of the concept and ontology for effective operation management of a web-based information system

The solution to the problem of managing the web-based IS operation largely depends on the Consumer and Supplier's point of view on the IS operation processes. Different data of the points of view are determined mainly by the difference in the global goals of the Consumer and Supplier, considered in [5, 34]. Therefore, it is necessary to recognize the simultaneous existence of two such points of view:

- an approach to managing the operation of web-based IS from the Consumer's point of view (an approach to managing operations within the framework of the web-based IS operation process);
- an approach to managing the operation of web-based IS from the Supplier's point of view (an approach to managing operations within the framework of the web-based IS maintenance process).

Recognition of these points of view allowed to formulate the basic concept of effective management of the operation of web-based IS as a set of the following provisions [27]:

- effective management of the operation of IS and its CIs from the consumer's point of view is considered as a set of design and/or operational measures, the result of which is the achievement of global IS efficiency from the point of view of all employees of the Consumer while maintaining the costs of these measures at the level desired by the Consumer;
- effective management of the operation of IS and its CIs from the Supplier's point of view is considered as a set of design and/or operational measures, the result of which is the achievement of global IS efficiency from the point of view of all employees of the Supplier while maintaining the costs of these measures at the level desired by the Supplier;
- effective management of the IS operation as a whole while respecting the interests of the Consumer and the Supplier of this IS is considered as a special case of the multi-criteria optimization problem, the solution of which will be Pareto-optimal IS;
- effective management of the operation of individual CIs of the IS while respecting the interests of the Consumer and the Supplier of this IS is considered as a set of design and/or operational measures to transform a set of current values of the indicators of the properties of the CIs of the IS (operational characteristics of individual CIs of the IS) into a set of planned values of these CIs, at which the IS as a whole will be Pareto-optimal.

In this study, the main attention is paid to the development of elements of the methodology for effective operation management of web-based IS precisely from the point of view of the Supplier. The role of such a Supplier can be any IT company that provides (sells or leases) its own web-based IS for operation by enterprises or organizations acting as Consumers.

Based on the developed concept of operation management of web-based IS, an ontology of the methodology for effective management of IT services of web-based IS was proposed. This ontology was obtained by improving the ontology of the existing service management methodology. The diagram of the relationships between the main concepts of the ontology of effective IT service management is shown in **Fig. 1.3** [27].

To improve perception, **Fig. 1.3** does not show the main slots of frames describing the main concepts. In red in **Fig. 1.3**, concepts that are introduced into the ontology as a result of improving the service management methodology are highlighted. In italics in **Fig. 1.3**, concepts that expand the service management methodology in the course of solving applied service management problems are highlighted.

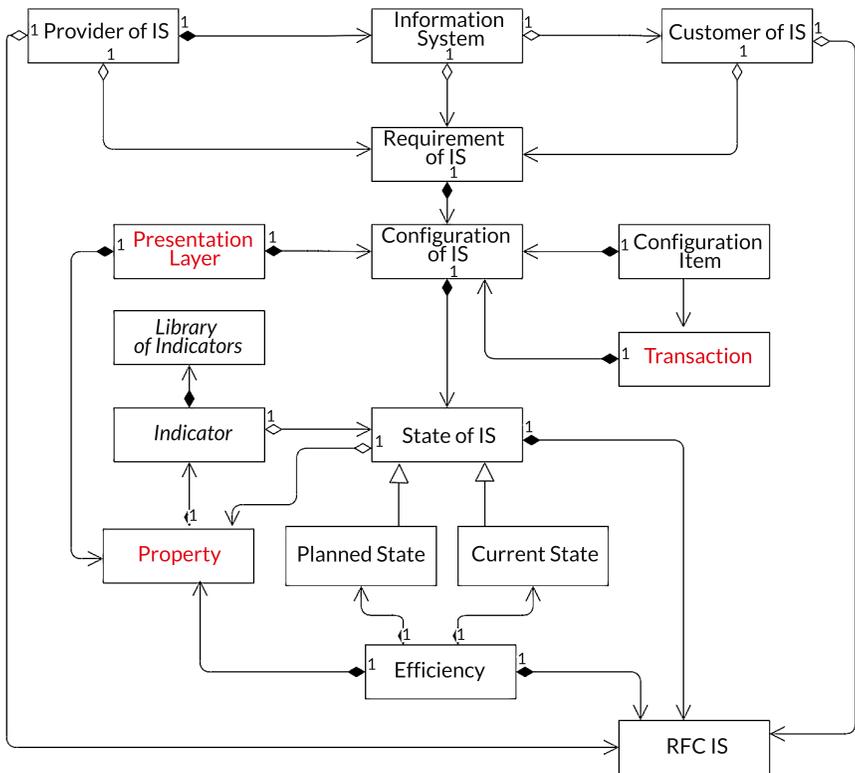


Fig. 1.3 Diagram of the relationships between the main concepts of the ontology of the methodology for managing IT services of web-based IS
Source: [27]

The scheme shown in **Fig. 1.3** allows to distinguish groups of main entities of the subject area, on the basis of which it is possible to form the following descriptions [27]:

- description of the operated IS at different levels of representation;
- description of the states of the operated IS;
- description of indicators that characterize the operated IS and its CIs;
- description of the efficiency of the operation of the IS by its individual properties and in general.

The description of the operated IS at different levels of representation will consist of the following concepts shown in **Fig. 1.3** [27]:

- "Information System" concept;

- "Provider of IS" concept;
- "Customer of IS" concept;
- "Requirement of IS" concept;
- "RFC IS" concept;
- "Configuration of IS" concept;
- "Presentation Layer" concept;
- "Configuration Item" concept;
- "Transaction" concept.

These concepts define the main points of view that should be taken into account during a detailed formalized description of the operated IS at different levels of representation as a set of individual CIs and transactions created to meet the requirements of the IS Supplier and Consumer.

The description of the states of the operated IS will consist of the following concepts, shown in **Fig. 1.1** [27]:

- "Configuration of IS" concept;
- "Presentation Layer" concept;
- "Configuration Item" concept;
- "Transaction" concept;
- "State of IS" concept;
- "Planned state" concept;
- "Current state" concept.

These concepts define the main points of view that should be taken into account during a detailed formalized description of the operated IS at different levels of representation of both planned and current states of individual CIs and transactions.

The description of indicators that characterize the operated IS and its CIs will consist of the following concepts, shown in **Fig. 1.3** [27]:

- "State of IS" concept;
- "Indicator" concept;
- "Library of Indicators" concept.

These concepts define the main points of view that should be taken into account during a detailed formalized description of the operated IS by a set of individual indicators based on a specific library of indicators.

The description of the efficiency of the IS operation by its individual properties and as a whole will consist of the following concepts shown in **Fig. 1.3** [27]:

- "State of IS" concept;
- "Planned state" concept;
- "Current state" concept;
- "Indicator" concept;

- "Presentation Layer" concept;
- "Property" concept;
- "Efficiency" concept;
- "RFC IS" concept.

These concepts define the main points of view that should be taken into account during a detailed formalized description of the efficiency of the operated IS at different levels of representation.

Partial duplication of concepts in different descriptions is necessary to ensure the connections of these descriptions with each other.

1.4.2 Results of development of a set of theoretical and categorical models of effective operation of a web-based information system

Based on the groups of concepts of the subject area identified in **Subsection 1.4.1**, it becomes possible to develop generalized mathematical models of effective management of the operation of web-based IS. Since these descriptions are sets of individual concepts defined in a different way, they can be formally represented in the form of the following categories [27]:

- description of the operated IS at different levels of representation will be represented by the L_{IS} category;
- description of the states of the operated IS will be represented by the L_{St} category;
- description of indicators characterizing the operated IS and its CIs will be represented by the L_{ind} category;
- description of the efficiency of the IS operation by its individual properties and in general will be represented by the L_{Eff} category.

The L_I category, taking into account the concepts of the ontology of web-based IS service management and the connections between these concepts, has the form

$$L_{IS} = \left[\begin{array}{l} Ob_{IS}, Ob_{Pr}, Ob_{Cust}, Ob_{rec}, Ob_{RFC}, Ob_{Conf}, Ob_{P_L}, Ob_{CI}, Ob_{Tr}, H_{Ob_{IS}}^{Ob_{Pr}}, H_{Ob_{Cust}}^{Ob_{IS}}, \\ H_{Ob_{rec}}^{Ob_{IS}}, H_{Ob_{rec}}^{Ob_{Pr}}, H_{Ob_{Cust}}^{Ob_{IS}}, H_{Ob_{rec}}^{Ob_{Conf}}, H_{Ob_{P_L}}^{Ob_{Conf}}, H_{Ob_{CI}}^{Ob_{Conf}}, H_{Ob_{Tr}}^{Ob_{Conf}}, H_{Ob_{CI}}^{Ob_{Tr}}, H_{Ob_{RFC}}^{Ob_{Pr}}, H_{Ob_{RFC}}^{Ob_{Cust}}, H_{1_{Ob}} \end{array} \right], \quad (1.1)$$

where Ob_{IS} – a subset of L_{IS} category objects that describes the "Information System" concept; Ob_{Pr} – a subset of L_{IS} category objects that describes the "Provider of IS" concept; Ob_{Cust} – a subset of L_{IS} category objects that describes the "Customer of IS" concept; Ob_{rec} – a subset of L_{IS} category objects that describes the "Requirement of IS" concept; Ob_{RFC} – a subset of L_{IS} category objects that describes the "RFC IS" concept; Ob_{Conf} – a subset of L_{IS} category objects that describes the "Configuration of IS"

concept; Ob_{p_L} – a subset of L_{IS} category objects that describes the "Presentation Layer" concept; Ob_{Cl} – a subset of L_{IS} category objects that describes the "Configuration Item" concept; Ob_{Tr} – a subset of L_{IS} category objects that describes the "Transaction" concept; $H_{Ob_{IS}}^{Ob_{Pr}}$ – a subset of L_{IS} category morphisms defined between subsets Ob_{Pr} and Ob_{IS} ; $H_{Ob_{Cust}}^{Ob_{IS}}$ – a subset of L_{IS} category morphisms defined between subsets Ob_{IS} and Ob_{Cust} ; $H_{Ob_{rec}}^{Ob_{Pr}}$ – a subset of L_{IS} category morphisms defined between subsets Ob_{IS} and Ob_{rec} ; $H_{Ob_{rec}}^{Ob_{Pr}}$ – a subset of L_{IS} category morphisms defined between subsets Ob_{Pr} and Ob_{rec} ; $H_{Ob_{IS}}^{Ob_{Cust}}$ – subset of L_{IS} category morphisms defined between subsets Ob_{Cust} and Ob_{IS} ; $H_{Ob_{Conf}}^{Ob_{rec}}$ – subset of L_{IS} category morphisms defined between subsets Ob_{rec} and Ob_{Conf} ; $H_{Ob_{Conf}}^{Ob_{p_L}}$ – subset of L_{IS} category morphisms defined between subsets Ob_{p_L} and Ob_{Conf} ; $H_{Ob_{Conf}}^{Ob_{Cl}}$ – subset of L_{IS} category morphisms defined between subsets Ob_{Cl} and Ob_{Conf} ; $H_{Ob_{Conf}}^{Ob_{Tr}}$ – subset of L_{IS} category morphisms defined between subsets Ob_{Tr} and Ob_{Conf} ; $H_{Ob_{Tr}}^{Ob_{Cl}}$ – subset L_{IS} category morphisms defined between subsets Ob_{Cl} and Ob_{Tr} ; $H_{Ob_{RFC}}^{Ob_{Pr}}$ – a subset of morphisms of the L_{IS} category defined between the subsets Ob_{Pr} and Ob_{RFC} ; $H_{Ob_{RFC}}^{Ob_{Cust}}$ – a subset of morphisms of the L_{IS} category defined between the subsets Ob_{Cust} and Ob_{RFC} ; $H_{1_{ob}}$ – a subset of single morphisms defined on selected subsets of objects of the L_{IS} category.

The L_{St} category, taking into account the concepts of the web-based IS service management ontology identified in [27] and the relationships between these concepts, has the form

$$L_{St} = \left[Ob_{Conf}, Ob_{p_L}, Ob_{Cl}, Ob_{Tr}, Ob_{St}, Ob_{pl_St}, Ob_{c_St}, H_{Ob_{Conf}}^{Ob_{p_L}}, H_{Ob_{Conf}}^{Ob_{Cl}}, H_{Ob_{Conf}}^{Ob_{Tr}}, H_{Ob_{Conf}}^{Ob_{St}}, H_{Ob_{Conf}}^{Ob_{pl_St}}, H_{Ob_{Conf}}^{Ob_{c_St}}, H_{1_{ob}} \right], \quad (1.2)$$

where Ob_{St} – a subset of objects of the L_{St} category, which describes the concept of "State of IS" concept; Ob_{pl_St} – a subset of objects of the L_{St} category, which describes the concept of "Planned State" concept; Ob_{c_St} – a subset of objects of the L_{St} category, which describes the concept of "Current State" concept; $H_{Ob_{St}}^{Ob_{Conf}}$ – a subset of morphisms of the L_{St} category, defined between subsets Ob_{Conf} and Ob_{St} ; $H_{Ob_{pl_St}}^{Ob_{St}}$ – a subset of morphisms of the L_{St} category, defined between subsets Ob_{St} and Ob_{pl_St} ; $H_{Ob_{c_St}}^{Ob_{St}}$ – a subset of morphisms of the L_{St} category, defined between subsets Ob_{St} and Ob_{c_St} ; $H_{1_{ob}}$ – a subset of single morphisms, defined on selected subsets of objects of the L_{St} category.

The L_{Ind} category, taking into account the concepts of the web-based IS service management ontology highlighted in [27] and the connections between these concepts, has the form

$$L_{Ind} = \left[Ob_{St}, Ob_{Ind}, Ob_{Lib}, H_{Ob_{St}}^{Ob_{Ind}}, H_{Ob_{Lib}}^{Ob_{Ind}}, H_{1_{ob}} \right], \quad (1.3)$$

where Ob_{Ind} – a subset of objects of the L_{Ind} category, which describes the "Indicator" concept; Ob_{Lib} – a subset of objects of the L_{Ind} category, which describes the "Library of Indicators" concept; $H_{Ob_{St}}^{Ob_{Ind}}$ – a subset of morphisms of the L_{Ind} category, defined between the Ob_{Ind} and Ob_{St} subsets; $H_{Ob_{Lib}}^{Ob_{Ind}}$ – a subset of morphisms of the L_{Ind} category, defined between the Ob_{Ind} and Ob_{Lib} subsets; $H_{1_{Ob}}$ – a subset of single morphisms, defined on selected subsets of objects of the L_{Ind} category.

The L_{Eff} category, taking into account the concepts of the web-based IS service management ontology highlighted in [27] and the connections between these concepts, has the form

$$L_{Eff} = \left[\begin{array}{l} Ob_{St}, Ob_{Pl_{St}}, Ob_{C_{St}}, Ob_{Ind}, Ob_{P_{L}}, Ob_{Pr_{op}}, Ob_{Eff}, Ob_{RFC}, H_{Ob_{Pl_{St}}}^{Ob_{St}}, H_{Ob_{C_{St}}}^{Ob_{St}}, \\ H_{Ob_{St}}^{Ob_{Ind}}, H_{Ob_{Pr_{op}}}^{Ob_{St}}, H_{Ob_{Pr_{op}}}^{Ob_{P_{L}}}, H_{Ob_{Pr_{op}}}^{Ob_{Eff}}, H_{Ob_{Pl_{St}}}^{Ob_{Eff}}, H_{Ob_{C_{St}}}^{Ob_{Eff}}, H_{Ob_{RFC}}^{Ob_{St}}, H_{1_{Ob}} \end{array} \right], \quad (1.4)$$

where $Ob_{Pr_{op}}$ – a subset of objects of the L_{Eff} category that describes the "Property" concept; Ob_{Eff} – a subset of objects of the L_{Eff} category that describes the "Efficiency" concept; $H_{Ob_{Pr_{op}}}^{Ob_{St}}$ – a subset of morphisms of the L_{Eff} category defined between the Ob_{St} and $Ob_{Pr_{op}}$ subsets; $H_{Ob_{Pr_{op}}}^{Ob_{P_{L}}}$ – a subset of morphisms of the L_{Eff} category defined between the $Ob_{P_{L}}$ and $Ob_{Pr_{op}}$ subsets; $H_{Ob_{Pl_{St}}}^{Ob_{Eff}}$ – a subset of morphisms of the L_{Eff} category defined between the Ob_{Eff} and $Ob_{Pl_{St}}$ subsets; $H_{Ob_{C_{St}}}^{Ob_{Eff}}$ – a subset of morphisms of the L_{Eff} category defined between the Ob_{Eff} and $Ob_{Pl_{St}}$ subsets; $H_{Ob_{RFC}}^{Ob_{St}}$ – a subset of morphisms of the L_{Eff} category defined between the Ob_{St} and Ob_{RFC} subsets; $H_{1_{Ob}}$ – a subset of unit morphisms defined on selected subsets of objects of the L_{Eff} category.

According to the provisions of [3], the management of effective operation of web-based IS is based on the cycle "Plan – Do – Check – Act" (PDCA). In this management cycle:

- at the "Plan" stage, the development of web-based IS operation plans is carried out;
- at the "Do" stage, work is carried out to implement web-based IS operation plans;
- at the "Check" stage, work is carried out to verify the results of the implementation of web-based IS operation plans;
- at the "Act" stage, work is carried out to correct web-based IS operation plans.

Then the generalized model of effective management of web-based IS operation can be represented as a supercategory of the following form [27]

$$M_O = \left[L_p, L_D, L_{Ch}, L_A, F_{L_D}^{L_p}, F_{L_{Ch}}^{L_D}, F_{L_D}^{L_{Ch}}, F_{L_A}^{L_{Ch}}, F_{L_p}^{L_A} \right], \quad (1.5)$$

where L_p – a supercategory that formally describes the "Plan" stage; L_D – a supercategory that formally describes the "Do" stage; L_{Ch} – a supercategory that formally describes the "Check" stage; L_A – a supercategory that formally describes the

"Act" stage; $F_{L_D}^{L_P}$ – a single covariant functor that establishes a connection between the L_P and L_D supercategories; $F_{L_{Ch}}^{L_D}$ – a single covariant functor that establishes a connection between the L_D and L_{Ch} supercategories; $F_{L_D}^{L_{Ch}}$ – a single covariant functor that establishes a connection between the L_{Ch} and L_D supercategories; $F_{L_A}^{L_{Ch}}$ – a single covariant functor that establishes a connection between the L_{Ch} and L_A supercategories; $F_{L_P}^{L_A}$ – a single covariant functor that establishes a connection between the L_A and L_P supercategories.

The functors $F_{L_D}^{L_P}$, $F_{L_{Ch}}^{L_D}$, $F_{L_A}^{L_{Ch}}$ and $F_{L_P}^{L_A}$ connect the supercategories L_P , L_D , L_{Ch} and L_A according to the control cycle of the operated web-based IS. The functor $F_{L_D}^{L_{Ch}}$ establishes a connection between the supercategories L_{Ch} and L_D to describe situations when the operation of the web-based IS is carried out according to the developed plans and there is no need to adjust these plans.

To detail the formal description of the supercategories L_P , L_D , L_{Ch} and L_A , the category of description of the states of the operated IS L_{St} (1.2) was divided into two subcategories: the subcategory of description of the planned state of the operated IS L_{St}^P and the subcategory of description of the current state of the operated IS L_{St}^C . The L_{St}^P subcategory has the form [6, 27]

$$L_{St}^P = \left[\begin{array}{l} Ob_{Conf}, Ob_{P_L}, Ob_{Cl}, Ob_{Tr}, Ob_{St}, Ob_{Pl_St}, H_{Ob_{Conf}}^{Ob_{P_L}}, H_{Ob_{Conf}}^{Ob_{Cl}}, \\ H_{Ob_{Conf}}^{Ob_{Tr}}, H_{Ob_{Conf}}^{Ob_{Cl}}, H_{Ob_{St}}^{Ob_{Conf}}, H_{Ob_{Pl_St}}^{Ob_{St}}, H_{1_{Ob}} \end{array} \right], \quad (1.6)$$

and the subcategory L_{St}^C has the form [6, 27]

$$L_{St}^C = \left[\begin{array}{l} Ob_{Conf}, Ob_{P_L}, Ob_{Cl}, Ob_{Tr}, Ob_{St}, Ob_{C_St}, H_{Ob_{Conf}}^{Ob_{P_L}}, H_{Ob_{Conf}}^{Ob_{Cl}}, \\ H_{Ob_{Conf}}^{Ob_{Tr}}, H_{Ob_{Conf}}^{Ob_{Cl}}, H_{Ob_{St}}^{Ob_{Conf}}, H_{Ob_{C_St}}^{Ob_{St}}, H_{1_{Ob}} \end{array} \right]. \quad (1.7)$$

Then, taking into account the previously developed categorical descriptions of the exploited ISS, the L_P supercategory is proposed to be described as follows [6, 27]

$$L_P = \left[L_{IS}, L_{St}^P, L_{Ind}, F_{L_{St}^P}^{L_{IS}}, F_{L_{Ind}}^{L_{St}^P} \right], \quad (1.8)$$

where $F_{L_{St}^P}^{L_{IS}}$ – a single covariant functor that establishes a connection between the LIS categories L_{IS} and L_{St}^P ; $F_{L_{Ind}}^{L_{St}^P}$ – a single covariant functor that establishes a connection between the categories L_{St}^P and L_{Ind} .

The L_D supercategory is proposed to be described as follows [6, 27]

$$L_D = \left[L_{IS}, L_{St}^C, L_{Ind}, F_{L_{St}^C}^{L_{IS}}, F_{L_{Ind}}^{L_{St}^C} \right], \quad (1.9)$$

where $F_{L_{St}^{L_{IS}}}^{L_{IS}}$ – a single covariant functor that establishes a connection between the LIS categories L_{IS} and L_{St}^P ; $F_{L_{Ind}^{L_{St}}}^{L_{St}^P}$ – a single covariant functor that establishes a connection between the supercategories and L_{Ind} .

The L_{Ch} supercategory is proposed to be described as follows [6, 27]

$$L_{Ch} = \left[L_{IS}, L_{St}^P, L_{St}^C, L_{Ind}, F_{L_{St}^P}^{L_{IS}}, F_{L_{St}^C}^{L_{IS}}, F_{L_{Ind}^{L_{St}}}^{L_{St}^P}, F_{L_{Ind}^{L_{St}}}^{L_{St}^C} \right]. \quad (1.10)$$

The L_A supercategory is proposed to be described as follows [27]

$$L_A = \left[L_{IS}, L_{Eff}, F_{L_{Eff}}^{L_{IS}} \right], \quad (1.11)$$

where $F_{L_{Eff}}^{L_{IS}}$ – a single covariant functor that establishes a connection between the supercategories L_{IS} and L_{Eff} .

The diagram of the relationships between the elements of the generalized model of effective management of the operation of web-based IS (1.5) is shown in Fig. 1.4 [6].

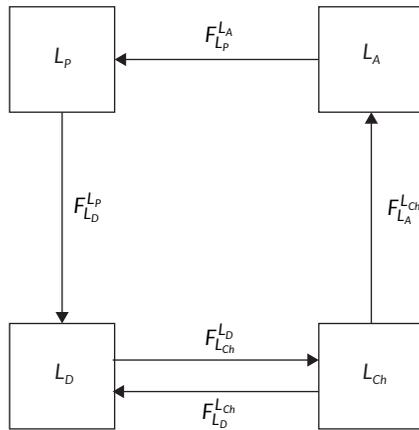


Fig. 1.4 Diagram of relationships between elements of a generalized model of effective management of the operation of a web-based information system

This diagram establishes transitions between the main states of the cycle "Plan – Do – Check – Act" of effective management of the operation of a web-based IS. In this diagram, states are defined as supercategories L_p , L_D , L_{Ch} and L_A , which describe the corresponding stages of the mentioned cycle. Transitions between these

states are shown as functors $F_{L_D}^{L_p}$, $F_{L_{Ch}}^{L_D}$, $F_{L_D}^{L_{Ch}}$, $F_{L_A}^{L_{Ch}}$ and $F_{L_p}^{L_A}$. Thus, model (1.5) can be represented as a finite state machine, transitions between the states of which are associated with changing the alphabets of the description of these states, although they retain their structural features.

1.4.3 Results of the development of a formal statement of the problem of effective management of the operation of a web-based information system

The use of the developed set of theoretical and categorical models (1.1)–(1.11) allowed to formally describe the global problem of managing the operation of a web-based IS as an IT service system. In doing so, it is possible to proceed from the following assumption.

Assumption 5: the main approach to managing an IT service system is the "Plan – Do – Check – Act" cycle.

As noted above, classical management of the operation of an IT service system is management aimed at minimizing the number of incident reports that remain unresolved. Therefore, it is proposed to consider the target state of the operated IT service system as a state in which the number of incident reports and RFC of the managed IT service system tends to 0. Let's call such a state of the operated IS as an IT service system stable.

The aspiration of the operated web-based IS to a stable state is proposed to be described by an objective function, which has the form [6]

$$\left| (Tr_i)_{i=1,n} \right| + \left| (RFC_j)_{j=1,m} \right| \rightarrow 0, \quad (1.12)$$

where Tr_i – the transaction, the execution of which generates the i -th incident; n is the number of incidents that occur during the operation of the web-based IS; RFC_j – the j -th RFC; m – the number of RFCs that occur during the operation of the web-based IS.

The objective function (1.12) of the classical (permanent) LC management r of the operation of a web-based IS is limited by the following conditions [6]:

– for each incident during its elimination, the planned and current states in which the operated IS may be during the control of its operation must be determined

$$\forall Tr_i \in Ob_{Tr} \in L_{IS} \in L_D \exists (L_{St}^P \in L_{Ch}, L_{St}^C \in L_{Ch}); \quad (1.13)$$

– for each RFC during its implementation, the planned and current states in which the operated IS may be during the control of its operation must be determined

$$\forall RFC_i \in Ob_{RFC} \in L_{IS} \in L_D \exists (L_{St}^P \in L_{Ch}, L_{St}^C \in L_{Ch}); \quad (1.14)$$

– the current state of the IS in the process of controlling its operation must fully correspond to its planned state (based on the features of the "Plan - Do - Check - Act" cycle and model (1.5))

$$|(L_{St}^P \in L_{Ch} - L_{St}^C \in L_{Ch}) \cap (L_{St}^C \in L_{Ch} - L_{St}^P \in L_{Ch})| = 0. \quad (1.15)$$

Such a formulation of the task of managing the operation of a web-based IS will allow obtaining an IS that will be best adapted to the requirements of its users. But such adaptability does not guarantee effective operation of the IS, because the set of requirements and RFC itself may not be aimed at achieving effective planned and current states of the operated IS. Therefore, an attempt to achieve goal (1.12) may lead to an increase in RFC. This increase is due to the fact that the number of RFCs that will arise during the operation of the IS can increase only because previous solutions to the management task will not give the desired coincidence of planned and current states. This situation will be especially pronounced in cases where the business processes of the automation object, the IS itself and the IT infrastructure in which the IS is operated are in a state of constant change. Unlike the considered concept of classical (constant) management, the proposed concept of effective management of the operation of web-based IS uses an additional concept of "system property" [31]. At the same time, the interest for the Consumer and the Supplier of IS is only those properties of the system that characterize the manifestations of individual qualities of the web-based IS. The qualities of the web-based IS, in turn, are determined by a set of requirements (functional and non-functional) that are imposed on the IS before the start of its operation, as well as a set of RFCs of the IS and its individual CIs formed during the operation of the IS. In this case, effective management of the operation of the web-based IS will be defined as actions to achieve the planned results for each individual property of the IS and its CIs.

Then the task of achieving the global goal of effective management of the operation of the web-based IS will be considered as the task of achieving optimal characteristics of this IS for each of its specific properties and the minimum probability of the existence of unsolved incidents and RFCs during the operation stage of this IS. The formal description of the objective function of this task will be as follows [6]

$$\begin{cases}
 f_{\text{Eff}}^1(\text{Ob}_{\text{Prop}}^1 \in L_{\text{Eff}}, \text{Ob}_{\text{Eff}} \in L_{\text{Eff}}) \rightarrow \text{opt}^1, \\
 f_{\text{Eff}}^2(\text{Ob}_{\text{Prop}}^2 \in L_{\text{Eff}}, \text{Ob}_{\text{Eff}} \in L_{\text{Eff}}) \rightarrow \text{opt}^2, \\
 \dots\dots\dots \\
 f_{\text{Eff}}^z(\text{Ob}_{\text{Prop}}^z \in L_{\text{Eff}}, \text{Ob}_{\text{Eff}} \in L_{\text{Eff}}) \rightarrow \text{opt}^z, \\
 \sum_{i=1}^n P(\text{Tr}_i) + \sum_{j=1}^m P(\text{RFC}_j) \rightarrow \min,
 \end{cases} \quad (1.16)$$

where 1, 2, ..., z – identifiers of individual properties of the operated IS; $P(\text{Tr}_i)$ – probability of no beneficial exchange between the operated IS and its environment as a result of transaction Tr_i ; $P(\text{RFC}_j)$ – probability of no beneficial exchange between the operated IS and its environment as a result of RFC RFC_j implementation.

The objective function of effective management of web-based IS operation (1.16) is limited by the following conditions [6]:

– for any specified k -th property of the operated IS, there must be a set of states of this IS, being in which the operated IS will be considered effective for this property

$$\forall \text{Ob}_{\text{Eff}}^k \in \text{Ob}_{\text{Eff}} \in L_{\text{Eff}} \in L_A \rightarrow \text{opt}^k \exists (\text{Ob}_{\text{St}} \in L_{\text{IS}} \in L_D, \text{Ob}_{\text{Ind}} \in L_{\text{IS}} \in L_D), \quad (1.17)$$

where $\text{Ob}_{\text{Ind}} \in L_{\text{IS}} \in L_D$ – a set of indicators by which the fact of this IS being in an effective state is established during the operation of the web-based IS;

– for each incident during its elimination, planned and current states must be determined, being in which the operated IS can be considered effective

$$\forall \text{Tr}_i \in \text{Ob}_{\text{Tr}} \in L_{\text{IS}} \in L_D \exists (L_{\text{St}}^p \in L_{\text{Ch}}, L_{\text{St}}^c \in L_{\text{Ch}}, L_{\text{Eff}} \in L_A); \quad (1.18)$$

– for each RFC during its implementation, planned and current states must be determined, being in which, the operated web-based IS can be considered effective

$$\forall \text{RFC}_j \in \text{Ob}_{\text{RFC}} \in L_{\text{IS}} \in L_D \exists (L_{\text{St}}^p \in L_{\text{Ch}}, L_{\text{St}}^c \in L_{\text{Ch}}, L_{\text{Eff}} \in L_A); \quad (1.19)$$

– the current state of the web-based IS in the process of controlling its operation must approach its planned state (based on the features of the "Plan – Do – Check – Act" cycle and model (1.5)) with some permissible small deviation

$$|(L_{\text{St}}^p \in L_{\text{Ch}} - L_{\text{St}}^c \in L_{\text{Ch}}) \cap (L_{\text{St}}^c \in L_{\text{Ch}} - L_{\text{St}}^p \in L_{\text{Ch}})| \leq \varepsilon. \quad (1.20)$$

The developed formulation of the problem of effective management of the operation of web-based IS allows to consider the problem of classical (permanent) management of the operation of web-based IS as a separate case of problem (1.16)–(1.20). For this case, the objective function (1.16) will have the form:

$$\left\{ \begin{array}{l} f_{\text{Eff}}^1 (Ob_{\text{Prop}}^1 \in L_{\text{Eff}}, Ob_{\text{Eff}} \in L_{\text{Eff}}) \rightarrow opt^1, \\ f_{\text{Eff}}^2 (Ob_{\text{Prop}}^2 \in L_{\text{Eff}}, Ob_{\text{Eff}} \in L_{\text{Eff}}) \rightarrow opt^2, \\ \dots\dots\dots \\ f_{\text{Eff}}^z (Ob_{\text{Prop}}^z \in L_{\text{Eff}}, Ob_{\text{Eff}} \in L_{\text{Eff}}) \rightarrow opt^z, \\ \sum_{i=1}^n P(Tr_i) + \sum_{j=1}^m P(RFC_j) \rightarrow 0. \end{array} \right. \quad (1.21)$$

The proposed formal description of the task of effective management of the operation of a web-based IS will allow obtaining an IS that will be during its operation:

- best adapted to the requirements of its users and the features of the IT infrastructure and business processes of the automation object existing during operation;
- be in those states that will be recognized as effective by both the Consumer and the Supplier (including, based on their strategies and goals within the IS LC [5]).

1.5 Results of the development of a theoretical and categorical model of the property of an information system

The proposed formal statement of the problem of effective management of the operation of web-based IS requires specifying the formal description of the concept "Property" and its possible connections with the "State of IS", "Indicator", "Presentation Layer" and "Efficiency" concepts.

Based on the formal description of the objective function (1.16), effective management of the operation of web-based IS requires the use of sets of parameters (or estimates of their values), which are elements of sets of objects of the categorical models of the concepts "Property" and "Efficiency". Therefore, to specify the formal model of the property of IS, it is proposed to use the following assumption.

Assumption 6: the main principles of effective management at individual stages of web-based IS LC should be identical to each other.

Using this assumption allows to formulate the main principle of effective management of the operation of web-based IS by analogy with the main principle of managing IS requirements described in [5]. This principle consists in the gradual

transformation of the set of initial values of attributes describing the requirement into the set of desired values of the same attributes. The desired value should be understood as the value that the attribute acquires when describing the implemented requirement, which is verified by the appropriate tests [5].

Based on the proposed concept of effective management of the operation of the IS, it was concluded that the main principle of managing the operation of individual CIs of the IS consists in the cyclical gradual transformation of the set of attribute values describing the property of the IS into the set of effective values of the same attributes. The effective value should be understood as the value that the attribute of the IS property acquires when describing the Pareto-optimal (desired) state of the process of the operated IS, provided that this state is agreed upon by the IS stakeholders.

From this principle it follows that to describe different properties of the operated web-based IS, the same set of property attributes should be used, as well as a universal model for evaluating the property of the IS and its processes by these attributes.

It should be noted that in general, the problem of assessing the studied processes is currently solved at the level of ISO 15504 and ISO 330XX standard groups (in particular, ISO 33001-33004, ISO 33021, etc.). However, these standards are focused on assessing the levels of capability of the LC processes of systems and software products. To solve the problem of assessing the processes occurring in the systems and IT products themselves, and the levels of capabilities of these processes in the specified standards, it is recommended to adapt the models proposed in them to the specifics of these processes [35, 36]. Modern IT companies recognized the need for practical application of the provisions of the standard [35] (in connection with the provisions of the ISO 20000 standard) almost immediately after its publication [37]. But modern research is mainly concerned with solving individual problems [38] that arise during the practical application of the provisions of standards [35, 36], or conceptual frameworks for assessing the maturity of individual IS and IT products [39]. Therefore, the problem of adapting the model for assessing process capability levels to the specifics of the task of assessing web-based IS processes by individual properties requires a separate study.

In the general case, the process assessment model should connect the basic model of the studied process and the measurement scheme, which consists of the following components [35]:

- a set of capability levels;
- a set of process attributes;
- a rating scale.

For such a connection, the process assessment model should establish a mapping of the measurement scheme into the process capability scale, which forms the ordinate axis. To form the abscissa axis, the process assessment model should determine the result of mapping the basic process model into a set of process categories. The ordered elements of this set form the abscissa axis. Then the evaluation model should establish the result of determining the correspondence of the studied process category and its specific base model to a specific quantitative value of the capability scale. Based on this value, the evaluation model provides the opportunity to establish quantitative values of process attributes and determine the current level of capabilities of the studied process [36]. In a generalized form, the model of the concept "Property" is proposed to be represented as a category $L_{Prop} = [Ob_{Prop}; Mor_{Prop}]$, the set of objects of which has the form

$$Ob_{Prop} = (Ob_{P_G}, Ob_{PMF}, Ob_{AM}), \quad (1.22)$$

where Ob_{P_G} – a subset of objects of the L_{Prop} category that describe the general characteristics of the IS property; Ob_{PMF} – a subset of objects of the L_{Prop} category that describe the process measurement scheme by the IS property; Ob_{AM} – a subset of objects of the L_{Prop} category that describe the process measurement model by the IS property.

For the formal description of subsets of objects Ob_{P_G} , Ob_{PMF} and Ob_{AM} , it is proposed to use the approach that was already used in [5] for the formal description of IS requirements design patterns. According to this approach, any subset of objects of the L_{Prop} category can be formally represented as a tuple consisting of such subsets

$$Ob_A = \langle M_A^B, M_A^{Add} \rangle, \quad (1.23)$$

where M_A^B – the basic attribute model that formally describes the tuple of attributes that characterize the mandatory elements of the Ob_A subset; M_A^{Add} – an additional attributive model that formally describes a tuple of attributes characterizing the elements of the subset Ob_A , the addition of which can improve the technological implementation of the developed methodology for effective management of the operation of web-based IS.

Then the subset of objects Ob_{P_G} selected in (1.22) should be formally represented as follows

$$Ob_{P_G} = \langle M_{P_G}^B, M_{P_G}^{Add} \rangle. \quad (1.24)$$

The basic attributive model of the general characteristics of the IS property is proposed to be represented by a tuple, which has the form

$$M_{P_G}^B = \langle P_Name, P_Des, P_Date_B, P_Date_E \rangle, \quad (1.25)$$

where P_Name – an attribute that describes the name of the IS property; P_Des – an attribute that describes the textual definition of the IS property; P_Date_B – an attribute that describes the start date of the application of this property for effective management of the operation of web-based IS; P_Date_E – an attribute that describes the end date of the application of this property for effective management of the operation of web-based IS.

For a detailed description of the process measurement scheme for the IS property, it is proposed to represent a subset of the Ob_{PMF} objects of the L_{Prop} category as a set of the following subsets

$$Ob_{PMF} = (Ob_{AbL}, Ob_{PA}, Ob_{RS}), \quad (1.26)$$

where Ob_{AbL} – a subset of the L_{Prop} category objects that describe the levels of possibilities of the IS property; Ob_{PA} – a subset of the L_{Prop} category objects that describe the process attributes used to measure the value of the level of possibilities of the IS property; Ob_{RS} – a subset of objects of the L_{Prop} category that describe the rating scale used to measure the value of the capability level of the IS property.

Detailed descriptions of these subsets are also determined by the approach formally described in (1.23).

Then, the basic attributive model of capability levels is proposed to be represented by a tuple that has the form

$$M_{AbL}^B = \left\langle \left\langle L_Name, L_M, L_Des \right\rangle_i \right\rangle_{i=1, \dots, n}, \quad (1.27)$$

where $\langle L_Name, L_Des \rangle_i$ – a tuple of attributes that describe the i -th capability level; n – the number of capability levels determined by the results of choosing a specific maturity model (Capable Maturity Model, SPICE, etc.); L_Name – an attribute that describes the name of the capability level; L_M – an attribute that describes the name of the maturity model selected as the source of describing the capability levels; L_Des – an attribute that describes the textual definition of the capability level.

The basic attributive model of process attributes is proposed to be represented by a tuple that has the form

$$M_{PA}^B = \left\langle \left\langle PA_Name, PA_Des, PA_Date_B, PA_Date_E \right\rangle_j \right\rangle_{j=1, \dots, m}, \quad (1.28)$$

where $\langle PA_Name, PA_Des, PA_Date_B, PA_Date_E \rangle_j$ – a tuple of attributes that describe the j -th process attribute; m – the number of process attributes, which is determined by the results of choosing a specific maturity model (Capable Maturity Model, SPICE, etc.); L_Name – an attribute that describes the name of the process attribute; L_Des – an attribute that describes the textual definition of the process attribute.

The basic attributive model of the rating scale is proposed to be represented by a tuple that has the form

$$M_{RS}^B = \left\langle \left\langle RS_Name, RS_Des, RS_Date_B, RS_Date_E \right\rangle, \left\langle RS_DB, RS_DT, RS_Ch \right\rangle_k \right\rangle_{k=1, \dots, p}, \quad (1.29)$$

where RS_Name – attribute describing the name of the rating scale; RS_Des – attribute describing the textual definition of the rating scale; RS_Date_B – attribute describing the date of the start of the rating scale application; RS_Date_E – attribute describing the date of the end of the rating scale application; $\langle RS_DB, RS_DT, RS_Ch \rangle_k$ – tuple of attributes describing the k -th range of the rating scale; p – number of ranges of the rating scale, determined by the results of the selection of a specific maturity model (for example, according to the ISO 15504 standard and the ISO 330XX group of standards $p = 4$); RS_DT – attribute describing the minimum quantitative limit of the k -th range of the rating scale; RS_DT – attribute describing the maximum quantitative limit of the k -th range of the rating scale; RS_Ch – attribute describing the symbolic definition of the k -th range of the rating scale.

For a detailed description of the process measurement model by the IS property, it is proposed to represent a subset of Ob_{AM} objects of the L_{Prop} category as a set of the following subsets

$$Ob_{AM} = (Ob_{PAM}, Ob_{Tr}^{As}, Ob_{Ind}^{As}), \quad (1.30)$$

where Ob_{PAM} – a subset of L_{Prop} category objects that describe the attributes of processes used during measurement; Ob_{Tr}^{As} – a subset of L_{Prop} category objects that describe transactions as IS CIs, the state of which is measured; Ob_{Ind}^{As} – a subset of L_{Prop} category objects that describe indicators used to measure the state of a transaction as IS CIs.

Detailed descriptions of these subsets are also determined by the approach formally described in (1.23).

Then, it is proposed to represent the basic attributive model of process attributes used during measurement by a tuple that has the form

$$M_{PAM}^B = \left\langle \left\langle PA_Name, \langle PA_Val, PA_DT_As \rangle_i \right\rangle_j \right\rangle_{j=1, \dots, m; i=1, \dots, s}, \quad (1.31)$$

where $\langle PA_Name, \langle PA_Val, PA_DT_As \rangle_i \rangle_j$ – a tuple of attributes that describe the j -th attribute of the process used in measurement; PA_Name – an attribute that describes the name of the process attribute used in the measurement; $\langle PA_Val, PA_DT_As \rangle_i$ – a tuple of attributes that describe the result of the i -th measurement of the value of the j -th process attribute; PA_Val – an attribute that describes the value of the j -th process attribute, which is the result of the i -th measurement; PA_DT_As – an attribute that describes the date and time of the i -th measurement of the j -th process attribute; s – the number of measurements planned within the implementation of the PDCA cycle iteration.

The basic attributive model of a transaction as a IS CI, the state of which is measured, is proposed to be represented by a tuple that has the form

$$M_{Tr}^{B,As} = \left\langle \left\langle Tr_Name^{As}, Tr_DT_As, \langle CI_Name^{As} \rangle_c \right\rangle_t \right\rangle_{t=1, \dots, t_{act}; c=1, \dots, c_{act}}, \quad (1.32)$$

where $\langle Tr_Name^{As}, Tr_DT_As, \langle CI_Name^{As} \rangle_c \rangle_t$ – a tuple of attributes describing the t -th transaction as a IS CI participating in the measurement; Tr_Name^{As} – an attribute describing the name of the transaction participating in the measurement; Tr_DT_As – an attribute describing the date and time of the transaction's participation in the measurement; $\langle CI_Name^{As} \rangle_c$ – a tuple of attributes describing the list of IS CIs as elements of the transaction participating in the measurement; CI_Name^{As} – an attribute describing the name of the IS CI as an element of the transaction participating in the measurement; c – a designation of the IS CI that is an element of the t -th transaction; c_{act} – the maximum number of IS CIs that make up the t -th transaction on the date and time of the measurement; t – a designation of the transaction as a IS CI participating in the measurement; t_{act} – the maximum number of transactions as CIs that make up the IS configuration option that is current on the date and time of the measurement.

The basic attribute model of indicators used to measure the status of a transaction as an IS CI is proposed to be represented by a tuple, which has the form

$$M_{Ind}^{B,As} = \left\langle \left\langle CI_Name^{As}, Ind_Name^{As}, Ind_DT_M, Ind_Val \right\rangle_q \right\rangle_{q=1, \dots, Ind_{act}}, \quad (1.33)$$

where $\langle CI_Name^{As}, Ind_Name^{As}, Ind_DT_M, Ind_Val \rangle_q$ – a tuple of attributes that describe the q -th indicator used to measure the status of a transaction as a IS CI;

CI_Name^{As} – an attribute that describes the name of the IS CI (transaction or its individual element) participating in the measurement; Ind_Name^{As} – an attribute that describes the name of the q -th indicator used to measure the CI status; Ind_DT_M – an attribute that describes the date and time of measurement of the value of the q -th indicator; Ind_{act} – an attribute that describes the value of the q -th indicator as the result of the measurement, Ind_{act}^{max} – the maximum number of indicators used to measure the status of a transaction as a IS CI.

Based on the representations of the set of objects of the L_{Prop} category (1.22), (1.24), (1.26) and (1.30), it is proposed to represent the set of morphisms of the L_{Prop} category as a collection of the following subsets

$$Mor_{Prop} = (H_{P_G}, H_{Abl}, H_{PAM}, H_{AM}^{PMF}), \quad (1.34)$$

where H_{P_G} – a subset of morphisms that determine the subordination of descriptions of the process measurement scheme and model by the property of the IS to descriptions of the general characteristics of this property; H_{Abl} – a subset of morphisms that determine the subordination of descriptions of process attributes and rating scales to descriptions of the levels of capabilities of the IS property; H_{PAM} – a subset of morphisms that determine the subordination of descriptions of transactions as IS CIs, the state of which is measured, and indicators of these transactions to descriptions of process attributes used during measurement; H_{AM}^{PMF} – a subset of morphisms that determine the subordination of descriptions of the process measurement model to descriptions of the process measurement scheme.

The H_{P_G} subset of morphisms generally consists of the following morphisms

$$H_{general} = (H_{PMF}^{P_G}, H_{AM}^{P_G}), \quad (1.35)$$

where $H_{PMF}^{P_G}$ – a morphism that determines the hierarchy of subordination of descriptions of the process measurement scheme by the IS property to descriptions of the general characteristics of this property; $H_{AM}^{P_G}$ – a morphism that defines the hierarchy of subordination of descriptions of the process measurement model by the IS property to descriptions of the general characteristics of this property.

H_{Abl} morphisms subset in the general case consists of the following morphisms

$$H_{Abl} = (H_{PA}^{Abl}, H_{RS}^{Abl}), \quad (1.36)$$

where H_{PA}^{Abl} – a morphism that defines the hierarchy of subordination of descriptions of process attributes to descriptions of the corresponding capability level;

H_{RS}^{ABL} – a morphism that defines the hierarchy of subordination of descriptions of the rating scale to descriptions of the corresponding capability level.

H_{PAM} morphisms subset in the general case consists of the following morphisms

$$H_{PAM} = (H_{Tr}^{PAM}, H_{Ind}^{PAM}), \tag{1.37}$$

where H_{Tr}^{PAM} – a morphism that defines the hierarchy of subordination of descriptions of transactions as IS CIs, the state of which is measured, to descriptions of process attributes used during measurement; H_{Ind}^{PAM} – a morphism that defines the hierarchy of subordination of descriptions of indicators used to measure the state of a transaction as IS CI descriptions of process attributes used during measurement.

H_{AM}^{PMF} morphisms subset in the general case consists of a H_{PAM}^{PA} morphism that defines the subordination of descriptions of process attributes used during measurement to descriptions of process attributes of the measurement scheme.

A diagram of the structure of the category-theoretic model L_{Prop} , which describes the "Property" concept, is shown in Fig. 1.5.

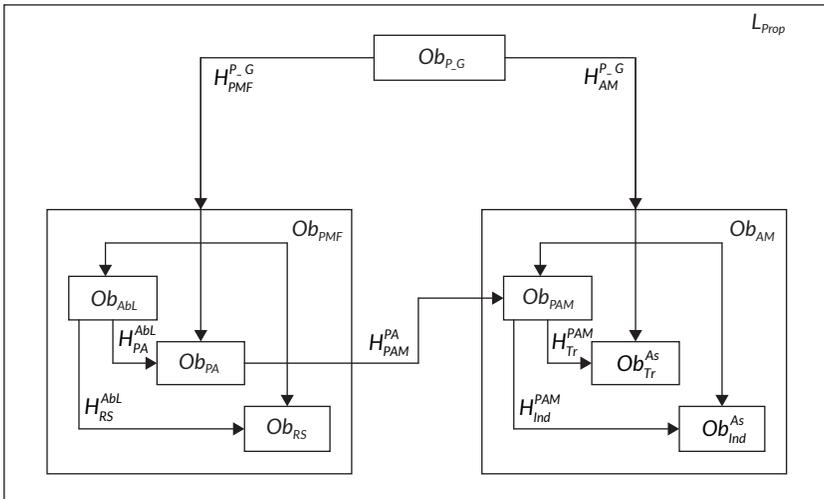


Fig. 1.5 Structure diagram of the theoretical-categorical model L_{Prop} , which describes the concept of "Property"

Modern standards and IT are focused on solving the problem of classical (permanent) management of IS operations, in which deviations between the planned and

current values of state indicators of individual IS CIs are measured. The application of the L_{Prop} model in the process of solving the problem of effective management of the operation of web-based IS made it possible to formulate and formally solve the following variants of this problem:

- the problem of minimizing deviations in the values of the levels of capabilities of IS transactions at any of the levels of representation of this IS (the problem of optimizing the levels of maturity of IS and its elements);
- the problem of minimizing deviations in the values of the attributes of IS transaction processes at any of the levels of capabilities and levels of representation of this IS (the problem of optimizing the states of IS and its elements within the selected maturity levels);
- the task of minimizing deviations in the values of IS transaction indicators and their CI within individual process attributes at any of the levels of capabilities and levels of representation of this IS (the task of classical (constant) IS operation management).

1.6 Discussion of the research results

In the process of the research, a concept of effective management of the LC of an operated web-based IS was proposed. The proposed concept consists in a multi-level representation of a web-based IS as a product with a set of individual properties. In this case, each property of the IS is considered as a manifestation of the individual qualities of this system. Each quality, in turn, is proposed to be considered as the result of fulfilling a set of requirements and RFCs of the IS and its SI. This concept allows to formally describe the task of effective management of the LC of an operated IS as a task of achieving optimal characteristics of this IS for each of its specific properties and the minimum probability of the existence of unresolved incidents and RFCs during the operation stage of this IS.

For a formal description of the proposed concept, a generalized model of effective management of the operation of web-based IS (1.5)-(1.11) and a complex of theoretical and categorical models (1.1)-(1.4) were developed, which detail the description of the elements of this model. The use of the apparatus of category theory allows to further present the generalized model (1.5)-(1.11) as a finite state machine, the transitions between the states of which are associated with the change of the alphabets of the description of these states, although they retain their structural features. Such a representation allows to consider the developed models (1.1)-(1.11) in further studies as the basis for a formal description of the

automated effective management of the operation of web-based IS based on the "Plan – Do – Check – Act" cycle. This description can exist provided that it is possible to prove the possibility of transforming the corresponding finite state machine into a consistent set of simpler finite state machines that describe IT processing of data structures at each stage of the cycle [6].

Based on the developed generalized model (1.5)–(1.11), a formal description (1.12) of the purpose of classical (permanent) LC management of the operated IS and limitations (1.13)–(1.15) was developed. Taking into account these formal descriptions and the results of the development of the concept of effective LC management of an operated web-based IS, a formal description of the objective function of effective management of the LC of an operated IS (1.16) and systems of constraints (1.17)–(1.20) was developed. The obtained results allow to define the problem of classical (permanent) LC management of an operated IS (1.21) as a special case of the problem of effective management.

To clarify the formal statement of the problem of effective LC management of an operated web-based IS, a theoretical-categorical model of the properties of an operated web-based IS (1.22)–(1.37) was developed. The use of the developed theoretical-categorical model L_{Prop} allowed to significantly change the features of the statement and solution of the problem of effective management of the operation of a web-based IS. Solving this problem using the developed model (1.22)–(1.37) will allow not only to maintain the operated IS in a stable state, but also to determine the optimal levels of maturity of the IS and its individual transactions from the point of view of the Consumer and the Provider of IT services. This allows stakeholders to plan and manage the recognition of the feasibility and implementation of the following measures during the operation of a web-based IS:

- standardization (typification) of individual IS transactions and IS as a whole at any of the representation levels;
- to ensure the manageability of individual transactions of the IS and the IS as a whole (in real time or time close to real);
- to ensure the optimal operation of individual transactions of the IS and the IS as a whole.

It should be noted that a significant number of modern studies in the field of PLM, ALM and SLM use the mathematical apparatus of graph theory for the formal description of the tasks to be solved. Examples of such studies are [40, 41]. In contrast, in this study, the mathematical apparatus of category theory was used for the formal description of the tasks of LC management of the operated IS. The theoretical and categorical models of the operated IS and the tasks of LC management of this IS used and developed in this study are based on the ontology of web-based

IS services management proposed by one of the authors of the study in [27] and allow to establish:

- the boundaries of domains and data structures used to describe individual concepts of the ontology;
- the rules for transforming domains and data structures when creating connections between different concepts of the ontology.

The applied application of the models developed in this study makes it possible to transition to effective LC management of an operated IS without global reengineering of systems and technologies for data storage and processing. The vast majority of IT products for LC management of an operated IS are highly specialized and relatively new products that occupy a small market segment. Therefore, systems for LC management of an operated IS are most often the result of integrating a large number of such highly specialized products. The proposed theoretical and categorical models allow in such cases [6]:

- to determine domains and data structures for technologies for integrating heterogeneous IT products into a single system;
- to ensure the transition from classical to effective LC management of an IT service system without serious changes in the list of highly specialized products in operation.

Also important from an applied point of view is the proposed presentation of the existing problem of classical (permanent) LC management of an operated IS as a separate case of the problem of effective management. This representation allows to consider the design and implementation of an effective LC management system of operated IS as a gradual evolutionary development of the previously created and already operating system for the classical LC management of IT services.

The main limitation of this study is the need to use complete descriptions of all elements of this IS to solve the problem of effective LC management of an operated IS. This means that during at least one iteration of the "Plan - Do - Check - Act" cycle, IS administrators must interact with a complete digital model of this IS at all levels of representation to solve individual tasks of LC management. The development and maintenance of such a model is a complex and costly task. But this limitation is methodological in nature and is valid for any work in the field of change management of developed or operated IS (which is confirmed, for example, by studies [42, 43]).

Another feature of the obtained results is their orientation on the LC management of the operated IS in conditions of constant changes in business processes, functions and elements of IT infrastructure. This feature is a consequence of the generalization of applied experience in supporting and ensuring the functioning of IS management of enterprises and organizations in conditions of martial law, which is in force in Ukraine [6].

Although the obtained research results are methodological in nature, they should be recognized as important for further theoretical and applied research in the field of creating PLM-, ALM- and SLM-IS and IT. The main further directions of development of this research are [6]:

- determination of the set of properties of the operated IS and efficiency indicators for these properties;
- development of new and improvement of existing models, detailing the formal description of the task of effective management;
- development and implementation of methods and algorithms for solving problems of effective management of various IS and IT options.

1.7 Conclusions

The concept of the task of effective LC management of an operated IS as an IT service system is formulated. The formulated concept allows to formally describe the task of effective LC management of an operated IS as the task of achieving optimal characteristics of this IS for each of its specific properties and the minimum probability of the existence of unresolved incidents and RFCs during the operation stage of this IS. The result obtained takes into account the points of view of the Supplier and Consumer of IT services as the main stakeholders of web-based IS. During further research, the main attention was paid to the development of elements of the methodology for effective management of the operation of web-based IS precisely from the point of view of the Supplier as an IT company that provides (sells or leases) its own web-based IS for operation by enterprises or organizations acting as Consumers.

A generalized model of effective management of web-based IS operation (1.5)–(1.11) and a set of theoretical and categorical models (1.1)–(1.4) that detail the description of the elements of this model have been developed. The developed models are a formal basis for creating IS and IT for effective management of web-based IS operation without global reengineering of systems and technologies for data storage and processing. Unlike existing models of a similar purpose, the developed models allow establishing the boundaries of domains and data structures used to describe the elements of the generalized model (1.5)–(1.11), as well as the rules for transforming domains and data structures when creating connections between different elements of this model.

A formal statement of the problem of classical (permanent) LC management of operated IS as an IT service system has been developed. A formal description of

the objective function (1.12) and constraints (1.13)–(1.15) of this problem is proposed using the apparatus of category theory and set theory. The developed formal formulation is based on the existing definition of the goal of such management as maximizing the satisfaction of IS users based on the results of each iteration of the "Plan – Do – Check – Act" cycle of LC management of the operated IS. It is proposed to consider the IS state as a formal sign of such satisfaction as the number of incident reports and RFCs of the operated IS tending to 0. The results obtained allow to formally describe the task of LC management of the operated IS in conditions of stable (unchangeable) business processes and IT infrastructure of enterprises and organizations as objects of automation.

A formal description of the objective function (1.16) and constraints (1.17)–(1.20) of the task of effective LC management of the operated IS is proposed using the apparatus of category theory, set theory and elements of probability theory. It is determined that the task of classical (permanent) LC management of the operated IS is a special case of the proposed task of effective management. Practical application of the proposed formal description of the task of effective LC management of an operated IS allows improving SLM-systems for LC management of an operated IS without global reengineering of existing systems and technologies for data storage and processing.

A theoretical and categorical model of the IS property (1.22)–(1.37) has been developed. This model allows formally describing any IS property taking into account the current requirements set by modern standards for the implementation of the processes of assessing the LC of the system. Using the developed model (1.22)–(1.37) allows solving tasks for automating effective planning and management of the operation of web-based IS.

Conflict of interest statement

The authors declare that there is no conflict of interest in relation to this paper, as well as the published research results, including the financial aspects of conducting the research, obtaining and using its results, as well as any non-financial personal relationships.

Use of artificial intelligence statement

The authors declare that they did not use artificial intelligence tools in preparing this manuscript.

References

1. What is Web (Based) Information System. IGI Global. Available at: <https://www.igi-global.com/dictionary/web-based-information-system/32189#:~:text=An%20information%20system%20that%20utilizes,or%20other%20information%20systems%2Fapplications>
2. Fulton, J. (2017). Web Architecture 101. Medium. Available at: <https://medium.com/storyblocks-engineering/web-architecture-101-a3224e126947>
3. ISO/IEC 20000-1. Information technology – Service management – Part 1: Service management system requirements (2018). Geneva: ISO Copyright Office, 96.
4. ISO/IEC/IEEE Standard No 15288:2015 (2015). Systems and software engineering – System life cycle processes. ISO/IEC/IEEE International Standard. <https://doi.org/10.1109/IEEESTD.2015.7106435>
5. Levykin, V. M., Evlanov, M. V., Kernosov, M. A. (2014). Patterny proektirovaniia trebovaniia k informatcionnym sistemam: modelirovanie i primenenie. Kharkiv: OOO "Kompaniia "Smit", 320.
6. Levykin, V., Evlanov, M., Levykin, I., Petrychenko, O. (2025). Development of a concept for the task of life cycle effective management of an operated information system. *Technology Audit and Production Reserves*, 2 (2 (82)), 66–73. <https://doi.org/10.15587/2706-5448.2025.326479>
7. Stark, J. (2020). *Product Lifecycle Management (Vol. 1)*. Cham: Springer International. <https://doi.org/10.1007/978-3-030-28864-8>
8. Schwaber, C. (2006). *The Changing Face of Application Life-Cycle Management*. Forrester Research Inc. Available at: <https://www.yumpu.com/en/document/view/13866040/download-the-changing-face-of-application-life-cycle-mks>
9. Wyrwich, F., Kharatyan, A., Dumitrescu, R. (2024). Interdisciplinary system life-cycle management – a systematic literature review. *Proceedings of the Design Society*, 4, 2765–2774. <https://doi.org/10.1017/pds.2024.280>
10. Rizzo, S. (2016). *Why ALM and PLM need each other*. Siemens Whitepaper. Available at: <https://polarion.plm.automation.siemens.com/hubfs/Docs/Whitepapers/why-alm-and-plm-need-each-other-whitepaper.pdf>
11. Liepert, C., Stary, C., Lamprecht, A., Zügn, D.; Elstermann, M., Lederer, M. (Eds.) (2025). *Interoperable Product Change Management Within Engineering: A Digital Twin Approach*. *Subject-Oriented Business Process Management. Models for Designing Digital Transformations*. S-BPM ONE 2024. *Communications in Computer and Information Science*. Vol. 2206. Cham: Springer. https://doi.org/10.1007/978-3-031-72041-3_17

12. Chappell, D. (2010). What is Application Lifecycle Management? David Chappell and Associates. Available at: http://davidchappell.com/writing/white_papers/What_is_ALM_v2.0--Chappell.pdf
13. Eigner, M. (2021). System Lifecycle Management: Digitalisierung des Engineering. Berlin, Heidelberg: Springer Vieweg. <https://doi.org/10.1007/978-3-662-62183-7>
14. Binder, C., Neureiter, C., Lüder, A. (2022). Towards a domain-specific information architecture enabling the investigation and optimization of flexible production systems by utilizing artificial intelligence. *The International Journal of Advanced Manufacturing Technology*, 123 (1-2), 49–81. <https://doi.org/10.1007/s00170-022-10141-2>
15. Colantoni, A., Berardinelli, L., Garmendia, A., Bräuer, J. (2022). Towards Blended Modeling and Simulation of DevOps Processes: the Keptn Case Study. *MODELS'22: Proceedings of the 25th International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings*, Association for Computing Machinery. New York, 784–792. <https://doi.org/10.1145/3550356.3561597>
16. Gulzar, K., Ruusu, R., Sierla, S., Aarnio, P., Karhela, T., Vyatkin, V. (2018). Automatic Generation of a Lifecycle Analysis Model from a First Principles Industrial Process Simulation Model. *2018 IEEE 16th International Conference on Industrial Informatics (INDIN)*. Danvers, 741–746. <https://doi.org/10.1109/indin.2018.8471980>
17. Calderon, N. N., Kajko-Mattsson, M., Nolan, A. J. (2015). Successful process improvement projects are no accidents. *Journal of Software: Evolution and Process*, 27 (11), 896–911. <https://doi.org/10.1002/smr.1738>
18. Reiff-Marganiec, S., Tilly, M. (Eds.) (2012). *Handbook of Research on Service-Oriented Systems and Non-Functional Properties: Future Directions*. IGI Global. <https://doi.org/10.4018/978-1-61350-432-1>
19. Driss, M., Aljehani, A., Boulila, W., Ghandorh, H., Al-Sarem, M. (2020). Servicing Your Requirements: An FCA and RCA-Driven Approach for Semantic Web Services Composition. *IEEE Access*, 8, 59326–59339. <https://doi.org/10.1109/access.2020.2982592>
20. Kienzle, J., Combemale, B., Mussbacher, G., Alam, O., Bordeleau, F., Burgueno, L. et al. (2022). Global Decision Making Over Deep Variability in Feedback-Driven Software Development. *Proceedings of the 37th IEEE/ACM International Conference on Automated Software Engineering*. New York. <https://doi.org/10.1145/3551349.3559551>
21. Moosbauer, J., Binder, M., Schneider, L., Pfisterer, F., Becker, M., Lang, M. et al. (2022). Automated Benchmark-Driven Design and Explanation of

- Hyperparameter Optimizers. *IEEE Transactions on Evolutionary Computation*, 26 (6), 1336–1350. <https://doi.org/10.1109/tevc.2022.3211336>
22. Garouani, M., Ahmad, A., Bouneffa, M., Hamlich, M., Bourguin, G., Lewandowski, A. (2022). Using meta-learning for automated algorithms selection and configuration: an experimental framework for industrial big data. *Journal of Big Data*, 9 (1). <https://doi.org/10.1186/s40537-022-00612-4>
 23. Bush, B., Stright, D., Huggins, J., Newes, E. (2022). Simulation process and data flow for a large system dynamics model. *Simulation*, 98 (9), 823–833. <https://doi.org/10.1177/00375497221093381>
 24. Ebert, C. (2013). Improving engineering efficiency with PLM/ALM. *Software & Systems Modeling*, 12 (3), 443–449. <https://doi.org/10.1007/s10270-013-0347-3>
 25. Deuter, A., Imort, S. (2020). PLM/ALM Integration With The Asset Administration Shell. *Procedia Manufacturing*, 52, 234–240. <https://doi.org/10.1016/j.promfg.2020.11.040>
 26. Deuter, A., Otte, A., Höllisch, D. (2017). Methodisches Vorgehen zur Entwicklung und Evaluierung von Anwendungsfällen für die PLM/ALM-Integration. *Wissenschaftsforum Intelligente Technische Systeme (WInTeSys)*. Paderborn, 211–222. <https://doi.org/10.17619/UNIPB/1-93>
 27. Petrichenko, O. V. (2021). Improving enterprise IT-service management methodology. *Management Information System and Devises*, 177, 4–12. <https://doi.org/10.30837/0135-1710.2021.177.004>
 28. Date, C. J. (2003). *Introduction to Database Systems*. Pearson, 1040.
 29. Dilovi tranzaktsii. Microsoft Build. Available at: <https://docs.microsoft.com/uk-ua/dynamics365/project-operations/psa/basic-business-transactions>
 30. Tanenbaum, A., Wetherall, D. (2010). *Computer Networks*. Pearson, 960.
 31. Fleishman, B.; Patten, B., Jorgenson, S. (Eds). (1995). *Stochastic Theory of Complex Ecological Systems*. *Complex Ecology*. New Jersey: Prentice Hall PTP, Prentice Hall Inc, 166–224.
 32. Kanaga Priya, P., Reethika, A.; Mishra, A., El Barachi, M., Kumar, M. (Eds.) (2024). A Review of Digital Twin Applications in Various Sectors. *Transforming Industry using Digital Twin Technology*. Cham: Springer, 239–258. https://doi.org/10.1007/978-3-031-58523-4_12
 33. Guinea-Cabrera, M. A., Holgado-Terriza, J. A. (2024). Digital Twins in Software Engineering – A Systematic Literature Review and Vision. *Applied Sciences*, 14 (3), 977. <https://doi.org/10.3390/app14030977>
 34. Ievlanov, M. V., Neumyvakina, O. E., Karamysheva, A. Iu. (2012). Global goals of IT-services supplier and consumer. *Eastern-European Journal of Enterprise*

- Technologies, 5 (2 (59)), 12–17. Available at: <https://journals.uran.ua/eejet/article/view/4137>
35. ISO/IEC/IEEE Standard No 15504-5 (2012). Information technology – Process assessment – An exemplar software life cycle process assessment model ISO/IEC/IEEE International Standard.
 36. ISO/IEC/IEEE Standard No 33061 (2021). Information technology – Process assessment – Process assessment model for software life cycle processes. ISO/IEC/IEEE International Standard.
 37. Mesquida, A. L., Mas, A.; O'Connor, R. V., Pries-Heje, J., Messnarz, R. (Eds.) (2011). ISO/IEC 15504-5 Best Practices for IT Service Management. Systems, Software and Service Process Improvement. Springer 14–24. https://doi.org/10.1007/978-3-642-22206-1_2
 38. Jung, H.-W., Ting, K.-F. (2017). An empirical evaluation of ISO/IEC 15504-5 capability measures: Reflective or formative. *Computer Standards & Interfaces*, 53, 123–130. <https://doi.org/10.1016/j.csi.2017.03.002>
 39. Monla, Z., Assila, A., Beladjine, D., Zghal, M.; De Paolis, L. T., Arpaia, P., Sacco, M. (Eds.) (2023). A Conceptual Framework for Maturity Evaluation of BIM-Based AR/VR Systems Based on ISO Standards. *Extended Reality*. Springer, 139–156. https://doi.org/10.1007/978-3-031-43401-3_9
 40. Zimmermann, T. C., Konietzko, E., Lindow, K. (2024). Graph-based Parameter Management for Configuration Controlled Multi-level Modeling of Cyber-physical Systems. 2024 19th Annual System of Systems Engineering Conference (SoSE). Tacoma, 270–274. <https://doi.org/10.1109/sose62659.2024.10620964>
 41. Katzung, S., Cinkaya, H., Kizgin, U. V., Savinov, A., Baschin, J., Vietor, T. (2024). AI-based analysis and linking of technical and organisational data using graph models as a basis for decision-making in systems engineering. *Proceedings of the Design Society*, 4, 2625–2634. <https://doi.org/10.1017/pds.2024.265>
 42. Rostami, K., Stammel, J., Heinrich, R., Reussner, R. (2017). Change Impact Analysis by Architecture-based Assessment and Planning. *Lecture Notes in Informatics, Proceedings – Series of the Gesellschaft fur Informatik*. Hannover, 267, 69–70.
 43. Rostami, K., Heinrich, R., Busch, A., Reussner, R. (2017). Architecture-Based Change Impact Analysis in Information Systems and Business Processes. 2017 IEEE International Conference on Software Architecture (ICSA). Gothenburg, 179–188. <https://doi.org/10.1109/icsa.2017.17>

CHAPTER 2

Cost management of an information system IT project in the conditions of incomplete information about its elements

Maksym Yevlanov
Viktor Levykin
Borys Moroz
Dmytro Moroz
Ivan Iuriev

Abstract

The object of the study is the process of managing IT project decisions.

During the study, the problem of modeling losses during the design of a description of an information system (IS) element in conditions of incompleteness or lack of complete information about this element was solved. Research in this area is devoted to the development and improvement of models for making design decisions in conditions of complete certainty, risk and partial uncertainty. The formation and making decisions on managing IT projects in conditions of partial uncertainty remain almost unexplored.

It is proposed to use for the IT industry a cause-and-effect scheme for the formation of losses in industrial production due to the lack of information. According to this scheme, the consequences of the situation of the lack of information about the description of the IS element and the loss models that characterize these consequences were analyzed. The proposed scheme was adapted to the features of the design and implementation processes of IS elements. Detailed loss models were developed for the four main consequences of the situation of incompleteness or lack of information about the description of the IS element.

Experimental verification of the obtained results was carried out on the basis of the IT project for the creation of the "Electronic Compulsory Motor Third Party Liability Insurance Policy" service. The consequences of identifying an incomplete description of this service during the design process were considered. Two options for replacing the incomplete description of this service were proposed: developing the service "from scratch"; using the description of the "Online Store" module

functions selected from the IT project backlog. For the first option, the estimate of possible losses is \$123300.12. For the second option, the estimate of possible losses is \$50000.

During the analysis of the obtained results, it was noted that the obtained estimates characterize losses only on the IT project iteration that is planned. To increase the accuracy of the assessment of losses from eliminating the situation of lack of information about the description of the IS element, further research was deemed necessary. It was also deemed appropriate to conduct prospective research on the creation of IS and information technologies to support the assessment and decision-making of the management of an IT company and its IT projects in a time regime close to real time.

Keywords

IT project, description of a system element, incomplete information, cause-and-effect scheme, loss model, information system function.

2.1 Introduction

The decision management process is one of the technical project management processes [1]. The processes in this group are used to establish and deploy plans, execute plans, assess actual achievement and progress against plans, and manage project performance. The scope of these processes is the technical management of a project or its products for inclusion in a system. Therefore, individual technical management processes can be applied at any time in the life cycle and at any level in the project hierarchy, as required by plans or due to unforeseen events [1].

The purpose of the decision management process is defined in [1] as providing a structured, analytical framework for objectively identifying, characterizing, and evaluating multiple solution alternatives at any point in the system life cycle and selecting the most advantageous course of action. This process is used to solve technical or design problems and respond to solution requests that arise during the system life cycle, and to determine the alternative that provides the best results in a given situation.

However, the problems of decision-making in project management are far from being completely solved. Thus, in [2] the scope of project execution "Uncertainty" is defined, which covers operations and functions related to risk and uncertainty. The allocation of this scope confirms the understanding of the importance of using models and methods for solving decision-making problems

under risk and partial uncertainty in project management systems. But when it comes to specific recommendations for the implementation of such models and methods, in [2] only the following options for responding to uncertainty are proposed:

- collecting additional information;
- preparing for several final results;
- design based on a set;
- developing resilience.

All of these options require additional time during project management, which is undesirable, especially for such a type of project as IT projects. The reasons for such undesirability lie mainly in the possibility of significant time losses as a result of exceeding the deadlines for the implementation of the IT project. The options for responding to uncertainty specified in [2] leave open the question of the possibility of using already known models and decision-making methods in IT project management. It should be recognized that such models and methods were mainly developed for other types of management than IT project management. Therefore, there is a need to conduct scientific and applied research in the field of developing new and using existing models and decision-making methods under uncertainty to solve IT project management problems.

2.2 Analysis of current research in the field of IT project decision management

Modern analysis of the application of traditional project management models and methods recognizes as the main challenge and the main direction of research in this field the complexity of project processes, due to which they do not meet the initial deadlines, cost, quality and business goals [3]. Based on this, in [3] the following main difficulties in project management are highlighted: delays in the project implementation schedule; lack of clearly defined goals and support from management/company; changes in scope; insufficient resources; poor risk management and measurement of project effectiveness; lack of communication between project participants. All these difficulties lead to solving project management problems under conditions of risk or partial uncertainty. Such conditions, in turn, necessitate the use of formal models and decision-making methods in the decision management process discussed above. The importance of using decision-making models and methods based on a scientific approach in project management is also confirmed by experimental studies. Thus, in [4], data from two

randomized controlled studies involving 382 entrepreneurs were analyzed, who were divided into two groups:

- a group of "scientific" entrepreneurs, who were trained in a scientific approach to decision-making regarding project selection before the experiments began;
- a group of "ordinary" entrepreneurs.

The results of the analysis presented in [4] allow to draw the following conclusions:

- the increased probability of project termination by "scientific" entrepreneurs is associated with greater accuracy in determining the cost of the project;
- "scientific" entrepreneurs are faster to adjust their expectations regarding the cost of the project downwards before making a decision to terminate the project;
- "scientific" entrepreneurs generate a greater number of new ideas, and a greater proportion of their projects end with the opening of a business;
- in the long term (up to five years), the initial difference in termination rates between "scientific" and "ordinary" entrepreneurs is equalized;
- the control group, which consisted of "regular" entrepreneurs, shows a higher rate of project abandonment in the long term (up to five years).

These findings generally support the idea that "scientific" entrepreneurs are not overly critical in their evaluation of their projects; rather, by abandoning projects with lower potential earlier, they can free up resources to reallocate elsewhere [4].

It should be recognized that in general, scientific methods and decision-making models are used in the management of a wide variety of projects. Thus, according to the results of the study [5], it is proposed that outsourcing decisions, including the choice of outsourcing service and contractor, be made by an expert group. [5] also considers constant monitoring and control of the quality of external units and the satisfaction of stakeholders to be important. [6] proposes to use a resource optimization model based on a genetic algorithm to solve the problem of resource optimization in the management of research projects. [7] proposes to use object information modeling technology to visually display the cost of the project, generalize and integrate various costs and, thus, obtain a total budget for the cost of the project to find optimization solutions in order to facilitate analysis and decision-making by project managers and relevant personnel. However, all the studies considered are based on the assumption that decision-making in project management is carried out under conditions of certainty.

Issues of decision support in project management under risk and uncertainty are also highlighted in modern research. Thus, the main structural components, elements and functions of an intelligent software system are described in [8], which is a tool for supporting management decisions in the implementation of practical projects in the field of economic management. This system is built on the principles

of the methodology for optimizing adaptive project management using network economic and mathematical modeling and the feedback principle. The use of this methodology allows taking into account the incompleteness of information associated with the failure or delay of specific project operations.

In [9], a review of solutions for computer-aided decision support for offshore software outsourcing in the global context of the development of such systems is provided. In particular, in [9] ten main key factors related to the decision-making process in the IT industry were identified, namely: human communication, cost reduction, organizational and professional maturity, project management methods, IT infrastructure support, language restrictions, knowledge-based support, changes in requirements, legal issues and cultural diversity. However, the data obtained in [9] show that the software industry lacks effective and efficient decision-making models that take into account the specifics of the IT industry and, in particular, the features of management in IT projects. The conclusion about the lack of application of effective and efficient models of decision-making and decision-making in IT project management is also confirmed by the results of other studies. Thus, in [10] an approach to decision analysis is proposed, which allows for the implementation of structured, reproducible and group execution of relevant procedures using a decision support system. However, this approach, as shown in [10], has only a methodological basis. In [11], a study was conducted to support decision-making on the implementation of test automation in the context of Agile-based Software Development.

In total, the study identified twenty-one factors that significantly affect test automation in the specified context. But the final result of applying the identified factors in [11] is only a conceptual model designed to help managers practicing Agile make decisions on the implementation of test automation in the specified context.

The results of the analysis allow to conclude that it is appropriate to conduct scientific and applied research in the field of developing new and improving existing formal models and decision-making methods for IT project management. These studies, taking into account the results obtained in [4], are proposed to be based on the following assumptions:

- the greatest effect from the application of these models and methods should be expected when making decisions on current and operational management of work at the early stages of the project;
- it is desirable to use such models and decision-making methods from various aspects of project management that would allow to assess the impact of the studied alternatives of the decision being formed and adopted on the overall assessment of

the project cost and, accordingly, on the estimates of the costs of performing the work of this project.

These assumptions made it possible to determine that the purpose of this study is to develop loss models when designing a description of the elements of the enterprise management information system (IS) in conditions of incompleteness or lack of complete information about these elements. The use of these models will allow formalizing the procedure for choosing a rational option for using personnel and IT infrastructure of an IT company during the design or implementation of IS elements, the descriptions of which are characterized by incomplete information.

To achieve this aim, it is proposed to solve the following research objectives:

- adapt the situational-consequential scheme of production losses due to lack of information to the features of the design processes of IS elements;
- develop and implement detailed loss models for the main consequences of production losses and decision-making on managing iterations of the implementation of the IT project of creating IS;
- perform experimental verification of the results obtained.

2.3 Research models

To determine the mechanism of loss formation during the design and implementation of IS elements, it is proposed to apply the existing causal scheme of loss formation in production due to the lack of information (**Fig. 2.1**) [12]. It should be noted that causal schemes are considered by some modern researchers as one of the means of overcoming uncertainty in project management [13].

In **Fig. 2.1**, the following designations are adopted: Ex_1 - "Replacing missing information with other, for example, scattered or indirect" consequence; Ex_2 - "Concentration of production resources on performing work not for their direct purpose" consequence; Ex_3 - "Transferring production resources to performing work that corresponds to their direct purpose" consequence; Ex_4 - "Refusal to use resources due to the lack of information for making management decisions" consequence; Ex_5 - "Any measures to maintain the required duration of the production cycle are not taken" consequence; Ex_6 - "Intensification of the production process after the elimination of uncertainty" consequence; P_1-P_8 and $P'_9-P'_{13}$ - losses characterizing the corresponding consequences; P_{sum} - total losses of the consequence Ex_1 ; W_1-W_3 - penalties for failure to complete the planned work on time. A detailed description of these elements is given in [12].

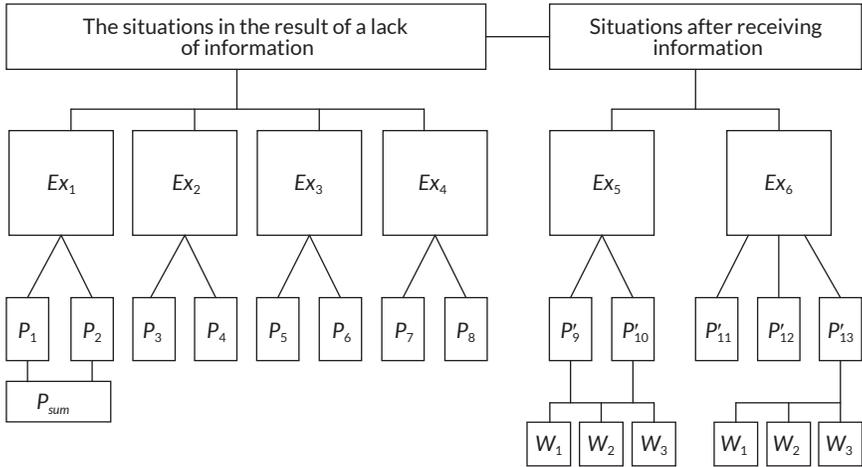


Fig. 2.1 Cause-and-effect scheme of production losses due to lack of information

The occurrence of a situation where there is no information can lead to one of the consequences Ex_1, Ex_2, Ex_3 and Ex_4 . The consequence Ex_1 attracts the most attention among them, because it allows replacing the description of an IS element for which complete information is missing with reusable descriptions of other IS elements being designed or other IS. The loss models P_1 and P_2 , which characterize the consequence Ex_1 , have the form [12]:

$$P_1 = S_{add}R(H)k, \tag{2.1}$$

$$P_2 = \Delta SR(H)k, \tag{2.2}$$

where P_1 – losses from the absence of information in the event of the need for further processing with replacement information; S_{add} – the average cost of additional processing of replacement information; $R(H)$ – the average number of occurrences of situations of absence of information for the analyzed period of time T with a certain composition of information $H = \{h_1, h_2, \dots, h_n\}$; k – the coefficient of quality of management decision-making; P_2 – losses from the absence of information in the presence of a difference in the cost of information being replaced and information being replaced; ΔS – the average difference in the cost of information being replaced and information being replaced. In some cases, as shown in the scheme (Fig. 2.1), these losses are summed up.

According to the result of Ex_2 , production resources are concentrated on performing work not for their direct purpose. For this case, the loss models P_3 and P_4 , which characterize the consequence of Ex_2 , have the form [12]

$$P_3 = \beta S_{res_{sv}}^{t_{st}} \times l t_{indef}(H) R(H) k_1 \quad (2.3)$$

provided $t_{indef}(H) \geq t_c$;

$$P_4 = \beta S_{res_{sv}}^{t_{st}} \times l t_c(H) R(H) k_1 \quad (2.4)$$

provided that $t_{indef}(H) < t_c$, where P_3, P_4 – losses from uncertainty of the situation due to the lack of management information when transferring resources to perform work not for their direct purpose; b – coefficient of losses on the cost of resources when they perform work not for their direct purpose; $S_{res_{sv}}^{t_{st}}$ – average cost of a resource unit for a normalized period of time t_{st} ; l – number of conditional units of resources; $t_{indef}(H)$ – average time of uncertainty of the situation with a certain information composition H ; t_c – average cycle of performing a normalized unit of work; k_1 – coefficient of quality of work performance.

The consequence Ex_3 is characterized by the transfer of production resources to perform work that corresponds to their direct purpose, which is possible in the case when the production process has additional components that ensure the employment of resources for a time not less than the time of uncertainty of the situation. If, to eliminate forced downtime of resources, the work is selected from a random set of unfinished work (if such is always available) and at the same time the creation of any special reserve is not required, then the losses can be calculated by the formula [12]

$$P_5 = \left(\frac{R(H) t S_{av.nc.w} t_{indef}(H)}{T t_c} + W_{sum} \right) - S_{unfin.w}, \quad (2.5)$$

where t – the time period under consideration; $S_{av.nc.w}$ – the average cost of an unperformed unit of work as a result of transferring resources; W_{sum} – the total value of penalties for failure to complete scheduled work on time as a result of transferring resources to reserve work; $S_{unfin.w}$ – the cost of completing unfinished work.

If a special reserve of work is created, then the losses are [12]

$$P_6 = \left(\frac{R(H) t S_{av.nc.w} t_{indef}(H)}{T t_c} + W_{sum} + S_{reserve} \right) - S_{compl.w.reserve}, \quad (2.6)$$

where $S_{reserve}$ – the cost of creating a special reserve of work; $S_{compl.w.reserve}$ – the cost of the completed reserve of work.

Consequence Ex_4 is a situation when, due to the lack of information for making management decisions, resources are not used. Losses in this case are proposed to be estimated in [12] as follows

$$P_7 = S_{res_{sv}}^{t_{st}} \times It_{indef}(H)R(H) \quad (2.7)$$

provided $t_{indef}(H) \geq t_c$;

$$P_8 = S_{res_{sv}}^{t_{st}} \times It_c R(H) \quad (2.8)$$

provided $t_{indef}(H) < t_c$.

After receiving the information necessary for making management decision information, consequences Ex_5 and Ex_6 arise. They are the results of consequences Ex_2 , Ex_3 and Ex_4 . Let's define consequence Ex_5 as a consequence in which no measures are taken to maintain the required production cycle duration. Naturally, in [12] it is assumed in this case that t_c will increase by $t_{indef}(H)$ from the moment of time at which the cycle was interrupted.

For the consequence Ex_5 , if the lack of information does not lead to an increase in the volume of work in progress, the losses are estimated as follows [12]

$$P'_9 = W_{sum} \quad (2.9)$$

In the case of an increase in the volume of work in progress, losses are determined as follows [12]

$$P'_{10} = \frac{S_{unfin_{sv}} t_{indef}(h)R(H)}{T} + W_{sum} \quad (2.10)$$

where $S_{unfin_{sv}}$ – the average cost of work in progress.

The losses associated with the penalty for the consequence Ex_5 , in the case of an increase in t_c , may have different values. If the increase in t_c does not lead to an increase in the critical period, $W_1 = 0$; if t_c increases and exceeds the critical period by an amount not greater than that specified in the relevant regulatory documents, then

$$W_2 = \alpha S_{unfin_{sv}}^{compl} R(H), \quad (2.11)$$

and in the case of an excess of t_c over the critical period by an amount greater than that specified in the standards,

$$W_3 = \alpha_1 S_{unfin_{ov}}^{compl} R(H), \quad (2.12)$$

where α , α_1 – the corresponding penalty coefficients for the specified cases; $S_{unfin_{ov}}^{compl}$ – the average total cost of work in progress.

The consequence Ex_6 is characterized by the fact that the production process after the elimination of uncertainty intensifies. If the increase in t_c due to the uncertainty of the situation can be eliminated without additional costs, for example, by reducing the inter-operational lag, the losses are zero ($P'_{11} = 0$). A possible option is when, in order to reduce t_c , the process is intensified by overtime use of resources [12].

Then, if $\Delta t \geq t_{indef}(H)$, losses $P'_{12} = \gamma S_{res_{ov}}^{t_{st}} \times t_{indef}(H) R(H)$, where γ – the coefficient that increases the cost of resources in the case of their use in overtime; Δt – the maximum possible reduction in the cycle duration. When [12]

$$P'_{13} = \gamma S_{res_{ov}}^{t_{st}} \times \Delta t R(H) + \frac{S_{unfin_{ov}} R(H)(t_{indef}(H) - \Delta t)}{T} + W_{sum}. \quad (2.13)$$

The losses due to penalties when increasing t_c are determined by formulas (2.11), (2.12) [13].

The scheme considered in Fig. 2.1 was developed for manufacturing enterprises and does not take into account the peculiarities of the processes of designing and implementing IS elements. In addition, the loss and penalty models (2.1)–(2.13) are not detailed, which makes their application in managing an IT project for creating or improving IS much more difficult.

2.4 Results of adapting the causal scheme of production losses to the features of the design processes of an information system element

In the course of adapting the causal scheme of production losses due to lack of information [12], it was decided to take into account the following features of the design and implementation processes of IS elements in an IT company:

- each specific IS corresponds to a separate IT project, which is performed by IT company employees within a predetermined time frame;
- an IT company can simultaneously perform several different IT projects;

- management of IS element development teams is carried out iteratively based on Agile IT project management methodologies;
- the description of each IS element being created is performed by one or more developers using a uniform structured template of this description H ;
- in the process of forming descriptions of individual IS elements, developers strive to achieve the maximum possible reuse of descriptions of existing elements of similar IS;
- the content of the task backlog for the development team as a set of IS element descriptions and even the duration T of each iteration (sprint, etc.) are set at the beginning of execution;
- the content of the task backlog rarely changes during the iteration being performed.

Here and now, the developers of IS elements will be understood as IT service providers who are employees of an IT company that takes responsibility for creating IS according to the requirements of IT service consumers [14] during all relevant stages of the IS life cycle [1].

These features allowed to formulate the definition of the consequences highlighted in the scheme as follows:

- consequence Ex_1 : "Replacement of missing information with information about a similar IS element that is reused";
- consequence Ex_2 : "Concentration of IT company personnel and IT infrastructure elements on the design or development of other IS elements";
- consequence Ex_3 : "Transfer of IT company personnel and IT infrastructure elements to the design or development of other elements of the same IS";
- consequence Ex_4 : "Refusal to use the personnel and elements of the IT infrastructure of the IT company due to the lack of information about the IS elements";
- consequence Ex_5 : "Failure to take any measures to maintain the required duration of the iteration T ";
- consequence Ex_6 : "Intensification of the processes of implementing the IT project of creating the IS after eliminating uncertainty".

Then the elements of the generalized loss formulas (2.1) and (2.2) are proposed to be interpreted as follows: P_1 - losses from the lack of information in the event of the need for additional processing with information from the description of the reused IS element; S_{add} - the average cost of additional processing of information, which is replaced, from the reused description of the IS element; $R(H)$ - the average number of occurrences of situations of lack of information for the period of time T under consideration, with a priori established template of the description of the IS element $H = \{h_1, h_2, \dots, h_n\}$; k - the coefficient of quality of management decision-making;

P_2 – loss from the absence of information in the presence of a difference in the value of the information being replaced and the information being replaced from the reused description of the IS element; ΔS – average difference in the value of the information being replaced and the information being replaced from the reused description of the IS element.

The elements of the generalized loss formulas (2.3) and (2.4) are proposed to be interpreted as follows: P_3, P_4 – losses from the lack of management information when transferring personnel and IT infrastructure elements of an IT company to the design or development of other IS elements; b – coefficient of losses on the cost of personnel and IT infrastructure elements of an IT company when they are used during the design or development of other IS elements; $S_{res,av}^{t_{st}}$ – average cost of a unit of personnel and IT infrastructure elements of an IT company for a normalized period of time t_{st} ; l – number of conditional units of resources (personnel and IT infrastructure elements) of an IT company that are used during the design or development of other IS elements; $t_{indef}(H)$ – average duration of the uncertainty of the description of an IS element with a priori established template of this description $H = \{h_1, h_2, \dots, h_n\}$; t_c – average duration of one iteration of the IT project work on the design and implementation of an IS element, $t_c = T$.

The elements of the generalized loss formulas (2.5) and (2.6) are proposed to be interpreted as follows: P_5 – losses from the transfer of personnel and IT infrastructure elements of the IT company to perform work on the design or development of IS elements in the case when this work is selected from a random set of unfinished work (if such is always available) and the creation of any special reserve is not required; P_6 – losses from the transfer of personnel and IT infrastructure elements of the IT company to perform work on the design or development of IS elements in the case when this work is selected from a created special reserve of work; $S_{av.nc.w}$ – the average cost of a unit of work not performed as a result of the transfer of personnel and IT infrastructure elements of the IT company; W_{sum} – the total value of penalties for failure to perform scheduled work on time as a result of the transfer of personnel and IT infrastructure elements of the IT company to reserve work.

The elements of the generalized loss formulas (2.7) and (2.8) are proposed to be interpreted as follows: P_7, P_8 – losses from non-use of personnel and IT infrastructure elements of the IT company due to the lack of information for making management decisions.

Similarly, the semantics of costs P'_9 and P'_{10} was clarified for the consequence Ex_5 and costs P'_{11}, P'_{12} and P'_{13} , and for the consequence Ex_6 . However, this clarification slightly changed the descriptions and interpretations of these costs.

2.5 Results of developing detailed models of losses from lack of information

The results of adapting the situational-consequence scheme of loss generation to the features of the design processes of IS elements, considered in **Section 2.4**, leave open the question of a detailed method of calculating these losses. Therefore, to apply the proposed mechanism of loss generation during the design and implementation of IS elements, it is necessary to develop detailed loss models that would allow obtaining quantitative estimates of losses for the consequences $Ex_1 - Ex_6$.

In the process of formulating a detailed description of the loss models (2.1) and (2.2), it was established that the main problem lies in clarifying the formal description of the parameters S_{add} and ΔS . Based on the results of developing parametric models for estimating labor costs (the modern term is efforts) for the implementation of the COCOMO II IT project [15], the following assumptions were formulated:

- Assumption 1: at different stages of the IS creation life cycle, different models are used to estimate efforts;
- Assumption 2: the need for personnel to perform the estimated work during an iteration with a duration of T can be determined by dividing the obtained effort estimate by the value of the parameter T ;
- Assumption 3: the costs of operating IT infrastructure elements in the most common conditions of an IT company's activity can be considered constant and evenly distributed over iterations with a duration of T .

These assumptions were proposed to be used as the basis for developing detailed loss models P_1 and P_2 .

In the case of using the P_1 (2.1) loss model to estimate losses in the case of consequence Ex_1 , the most difficult thing is to determine the value of the parameter S_{add} . Taking into account the above-mentioned features of the design and implementation processes of IS elements, as well as the results of adapting the cause-and-effect scheme of losses in production due to the lack of information, this parameter is proposed to be described as follows

$$S_{add} = \sum_{z=1}^{R(H)} \left(\sum_{i=1}^m k_{zi} q_{zi}(T) St_{zi}(T) + \sum_{j=1}^n k_{zj} q_{zj}(T) Cl_{zj}(T) \right), \quad (2.14)$$

where k_{zi} – the normative value of the cost of wages of a developer who is engaged in the process of creating the z -th IS element in the time period T in the i -th position; $q_{zi}(T)$ – the number of developers who are engaged in the process of creating the z -th IS element in the time period T in the i -th position; $St_{zi}(T)$ – the operator that

establishes the fact that developers are in the process of creating the z -th IS element in the time period T in the i -th position; k_{zj} – the normative value of the cost of operating the j -th IT infrastructure element in the process of creating the z -th IS element in the time period T ; $q_{zj}(T)$ – the number of j -th IT infrastructure elements operated in the process of creating the z -th IS element in the time period T ; $C_{Cj}(T)$ – an operator that establishes the fact of operation of the j -th IT infrastructure elements that are operated in the process of creating the z -th IS element in the time period T .

Taking into account Assumption 3, expression (2.14) can be written in a simplified form

$$S_{add} = \sum_{z=1}^{R(H)} \sum_{i=1}^m k_{zi} q_{zi}(T) St_{zi}(T) + \sum_{z=1}^{R(H)} C_{Cj}(z), \quad (2.15)$$

where $C_{Cj}(z)$ – the constant value of the costs of operating IT infrastructure elements in the process of creating the z -th IS element in the time period T .

Thus, it is necessary to determine the number of developers $q_{zj}(T)$, which is required to eliminate the situation that arose as a result of the lack of complete information about the description of the IS element in the IT project iteration of duration T , which is planned or implemented. Based on the COCOMO model, this parameter is proposed to be calculated in the following way

$$q_{zi}(T) = P_z / T, \quad (2.16)$$

where P_z – the effort to perform additional processing of the replacing information.

When using the P_2 (2.2) loss model to estimate losses in the case of the consequence Ex_1 , the most difficult thing is to determine the value of ΔS . To do this, it is first necessary to establish the peculiarity of the difference in the values of the information being replaced and the information being replaced. This peculiarity arises as a result of using the mechanism for reusing descriptions of previously developed IS elements. According to this mechanism, reuse of the description of an IS element is possible in one of the following cases:

- case I: the description was developed and implemented during one of the previous iterations;
- case II: the description is planned to be developed and implemented during the same iteration as the replaced description of the IS element, but one or more days earlier.

In case I, the value of the information value S_{repm} , which is replaced, is known to the developers for this element quite accurately from the reports on the progress of the previous iterations. In case II, the developers can only operate with an estimate

of the value of the information value \hat{S}_{repm} , which is replaced. This estimate, by analogy with (2.14) and (2.15), is proposed to be defined as follows:

$$\hat{S}_{repm} = \sum_{i=1}^m k_{ri} q_{ri}(T) St_{ri}(T) + \sum_{j=1}^n k_{rj} q_{rj}(T) Cl_{rj}(T), \quad (2.17)$$

$$\hat{S}_{repm} = \sum_{i=1}^m k_{ri} q_{ri}(T) St_{ri}(T) + C_{Cl}(z), \quad (2.18)$$

where r – the IS element, the description of which will be developed and reused in the planned iteration. The value of the quantity q_{ri} is set by a formula similar to (2.16).

As for the value of the information being replaced, developers can also operate only with its estimate \hat{S}_{repd} . The value of this estimate is proposed to be calculated, by analogy with (2.17) and (2.18), as follows:

$$\hat{S}_{repd} = \sum_{i=1}^m k_{di} q_{di}(T) St_{di}(T) + \sum_{j=1}^n k_{dj} q_{dj}(T) Cl_{dj}(T), \quad (2.19)$$

$$\hat{S}_{repd} = \sum_{i=1}^m k_{di} q_{di}(T) St_{di}(T) + C_{Cl}(d), \quad (2.20)$$

where d – the IS element, the description of which is incomplete or missing at the time of planning the iteration.

Then the value in (2.2) is proposed to be determined as follows

$$\Delta S = \sum_{r=1}^{R(H)} S'_{repm,r} - \sum_{d=1}^{R(H)} \hat{S}_{repd,d}, \quad (2.21)$$

where

$$S'_{repm,r} = \begin{cases} S_{repm,r} & \text{if (I),} \\ \hat{S}_{repm,r} & \text{if (II).} \end{cases} \quad (2.22)$$

Here (I) is case I of reuse of the r -th IS element; (II) is case II of reuse of the r -th IS element.

Based on formulas (2.14)–(2.22), a detailed description of the loss model P_1 (2.1) for the case of IT project management of IS creation is proposed to be presented as follows

$$P_1 = k \times \sum_{z=1}^{R(H)} \left(\sum_{i=1}^m k_{zi} q_{zi}(T) St_{zi}(T) + \sum_{j=1}^n k_{zj} q_{zj}(T) Cl_{zj}(T) \right), \quad (2.23)$$

or in a simplified case

$$P_1 = k \times \left(\sum_{z=1}^{R(H)} \sum_{i=1}^m k_{zi} q_{zi}(T) St_{zi}(T) + \sum_{z=1}^{R(H)} C_{Cl}(z) \right). \quad (2.24)$$

A detailed description of the loss model P_2 (2.2) for the case of IT project management of IS creation is proposed to be presented as follows

$$P_2 = k \times \left(\sum_{r=1}^{R(H)} S'_{repm,r} - \sum_{d=1}^{R(H)} \hat{S}_{repd,d} \right). \quad (2.25)$$

Expressions (2.14)–(2.25) allowed to state that the choice of the consequence Ex_1 will be appropriate only when the condition

$$P_{sum} = P_1 + P_2 \leq \sum_{d=1}^{R(H)} \hat{S}_{repd,d}, \quad (2.26)$$

is met, i.e. in situations where the costs of reusing the information being replaced do not exceed the costs of designing and implementing "from scratch" IS elements, the descriptions of which are incomplete or absent.

In the process of formulating a detailed description of the loss models (2.3) and (2.4), it was established that the main problem lies in clarifying the formal description of the parameter b . Based on the above-defined features of the processes of designing and implementing IS elements in an IT company, the following assumptions were formulated:

- Assumption 4: the transfer of personnel and elements of the IT infrastructure of an IT company from the IT project of creating IS A to the IT project of creating IS B is considered as replacing the backlog of tasks for developing IS elements A with the backlog of tasks for developing IS elements B;
- Assumption 5: the general mechanism for replacing information is single and unchanged for any IT project within an IT company.

Based on these assumptions, the value of the coefficient b is proposed to be calculated in the case of replacing the description of one IS element A with the description of one IS element B as follows

$$\beta = \hat{S}_{repd}^{IS_A} - \hat{S}_{repm}^{IS_B} / \hat{S}_{repd}^{IS_A}, \quad (2.27)$$

where $\hat{S}_{repd}^{IS_A}$ – the estimated value of information about the descriptions of the elements of IS A, which is being replaced; $\hat{S}_{repm}^{IS_B}$ – the estimated value of information about the descriptions of the elements of IS B, which is being replaced.

The value $\hat{S}_{repl,d}^{IS_A}$ for one d -th element of IS A, the description of which is absent or insufficient for the performance of work on the design and implementation of this element, is proposed to be calculated by formulas (2.19) or (2.20). The value $\hat{S}_{repl,r}^{IS_B}$ for one r -th element of IS B, the description of which is used during the formation of the team backlog instead of the description of the d -th element of IS A, is proposed to be calculated by formulas (2.17) or (2.18). Based on these estimates, expression (2.27) for the case of replacing the descriptions of two or more elements of IS A with descriptions of two or more elements of IS B was modified as follows

$$\beta = \left(\sum_{d=1}^{R(H)} \hat{S}_{repl,d}^{IS_A} - \sum_{r=1}^{R(H)} \hat{S}_{repl,r}^{IS_B} \right) / \sum_{d=1}^{R(H)} \hat{S}_{repl,d}^{IS_A}. \quad (2.28)$$

The product $S_{res_{ov}}^{t_{st}} \times I$ described the total resource costs for the normalized time period t_{st} . For the case of transferring resources involved in the design and implementation of the d -th element of IS A to perform work on the design and implementation of IS B, this product is proposed to be described as follows:

$$S_{res_{ov},d}^{t_{st}} \times I_d = \hat{S}_{repl,d}^{IS_A} = \sum_{i=1}^m k_{di}^{IS_A} q_{di}^{IS_A}(T) St_{di}^{IS_A}(T) + \sum_{j=1}^n k_{dj}^{IS_A} q_{dj}^{IS_A}(T) Cl_{dj}^{IS_A}(T), \quad (2.29)$$

$$S_{res_{ov},d}^{t_{st}} \times I_d = \hat{S}_{repl,d}^{IS_A} = \sum_{i=1}^m k_{di}^{IS_A} q_{di}^{IS_A}(T) St_{di}^{IS_A}(T) + C_{Cl}(d). \quad (2.30)$$

Then, in the general case, the product $S_{res_{ov}}^{t_{st}} \times I$ as a set of resources that were planned to be used to perform $R(H)$ of the IS A elements, but due to the uncertainty of the descriptions of these elements, it is recommended to transfer them to the IS B IT project, is proposed to be described as follows

$$S_{res_{ov}}^{t_{st}} \times I = \hat{S}_{repl}^{IS_A} = \sum_{d=1}^{R(H)} \hat{S}_{repl,d}^{IS_A}. \quad (2.31)$$

Based on formulas (2.27)–(2.31), a detailed description of the loss model P_3 (2.3) for the case of managing IT projects for the creation of IS A and IS B is proposed to be presented as follows

$$P_3 = \frac{\sum_{d=1}^{R(H)} \hat{S}_{repl,d}^{IS_A} - \sum_{r=1}^{R(H)} \hat{S}_{repl,r}^{IS_B}}{\sum_{d=1}^{R(H)} \hat{S}_{repl,d}^{IS_A}} \times \sum_{d=1}^{R(H)} \hat{S}_{repl,d}^{IS_A} \times t_{indef}(H) k_1 \quad (2.32)$$

provided that $t_{indef}(H) \geq T$.

A detailed description of the loss model P_4 (2.4) for the case of managing IT projects for the creation of IS A and IS B is proposed to be presented as follows

$$P_4 = \frac{\sum_{d=1}^{R(H)} \hat{S}_{repl,d}^{IS_A} - \sum_{r=1}^{R(H)} \hat{S}_{repl,r}^{IS_B}}{\sum_{d=1}^{R(H)} \hat{S}_{repl,d}^{IS_A}} \times \sum_{d=1}^{R(H)} \hat{S}_{repl,d}^{IS_A} \times Tk_1 \quad (2.33)$$

provided that $t_{indef}(H) < T$.

In the process of formulating a detailed description of the loss models (2.5) and (2.6), the following assumptions were made:

- Assumption 6: for any IT project to create an IS, there is always a project backlog as a set of tasks, the design and implementation of which is planned within the framework of a set of iterations of this IT project;
- Assumption 7: any backlog of a development team can be described as a set consisting of two disjoint subsets – a subset of main tasks, which is necessarily non-empty, and a subset of reserve tasks, which may be empty.

The elements of the subset of mandatory tasks of the team backlog are tasks planned for design and implementation by the team in the current iteration. The elements of the subset of reserve tasks of the same backlog are tasks that are provided for design and implementation to team representatives in the event of either their full completion of all assigned mandatory tasks, or recognition of the impossibility of performing assigned mandatory tasks due to the absence or uncertainty of the description of these tasks.

Based on these assumptions, the definition of losses arising as a result of EX_3 is proposed to be specified as follows: P_5 – losses from transferring personnel and IT infrastructure elements of the IT company to perform the task of designing and implementing an IS element in the case when it is selected from the project backlog; P_6 – losses from transferring personnel and IT infrastructure elements of the IT company to perform the task of designing and implementing an IS element in the case when it is selected from the subset of reserve tasks of the development team backlog.

Initially, the situation of P_5 losses was considered. In this case, a situation arises of replacing one or more tasks for the design and implementation of an IS element, the descriptions of which are missing or not defined with sufficient accuracy, with the corresponding number of tasks from the project backlog, the descriptions of which are fully defined. Then the estimate of the average cost $S_{av.nc.w}$ of a unit of work not performed as a result of the transfer of resources, by analogy with formulas (2.19) and (2.20), should be described as follows:

$$\hat{S}_{av.nc.w} = \sum_{i=1}^m k_{di} q_{di}(T) St_{di}(T) + \sum_{j=1}^n k_{dj} q_{dj}(T) Cl_{dj}(T), \quad (2.34)$$

$$\hat{S}_{av.nc.w} = \sum_{i=1}^m k_{di} q_{di}(T) St_{di}(T) + C_{Cl}(d), \quad (2.35)$$

where d – an IS element, the description of which is incomplete or missing at the time of iteration planning.

The estimate of the cost $\hat{S}_{unfin.w}$ of performing unfinished work, to which the personnel and IT infrastructure elements of the IT company were transferred, selected from a random set of unfinished work of the IS creation IT project, by analogy with formulas (2.17) and (2.18), should be described as follows:

$$\hat{S}_{unfin.w} = \sum_{i=1}^m k_{ri} q_{ri}(T) St_{ri}(T) + \sum_{j=1}^n k_{rj} q_{rj}(T) Cl_{rj}(T), \quad (2.36)$$

$$\hat{S}_{unfin.w} = \sum_{i=1}^m k_{ri} q_{ri}(T) St_{ri}(T) + C_{Cl}(z), \quad (2.37)$$

where r – an IS element, the fully defined description of which is selected from the project backlog.

As for the value of the parameter W_{sum} , in the general case it should be determined based on the policy of penalties for failure to perform work within certain time intervals, which is fixed in the contract for the implementation of the IT project. This contract is concluded between an IT company acting as an IT Service Provider (developer, IT project performer, etc.) and an organization or individual acting as an IT Service Consumer (customer, etc.). One of the classic examples of such a policy is the decision to pay a penalty as a predetermined percentage of the cost of an IT project for each working day during which the work remains unperformed. For this example, the parameter W_{sum} will take the value

$$W_{sum} = t k_{\%} P_{project}, \quad (2.38)$$

where t – the time period under consideration; $k_{\%}$ – penalty coefficient, $k_{\%} \in [0, \dots, 1]$, the value of which is fixed in the text of the contract for the implementation of the IT project; $P_{project}$ – cost of the IT project, the value of which is fixed in the text of the contract for the implementation of the IT project.

Based on formulas (2.34)–(2.38), a detailed description of the P_5 (2.5) loss model for the case when the task for replacement is selected from the project backlog is proposed to be presented as follows

$$P_5 = \left(\frac{R(H)t \left(\sum_{i=1}^m k_{di} q_{di}(T) St_{di}(T) + \sum_{j=1}^n k_{dj} q_{dj}(T) Cl_{dj}(T) \right) t_{indef}(H)}{Tt_c} + tk_{\%} P_{project} \right) - \left(\sum_{i=1}^m k_{ri} q_{ri}(T) St_{ri}(T) + \sum_{j=1}^n k_{rj} q_{rj}(T) Cl_{rj}(T) \right) \quad (2.39)$$

provided that $t_{indef}(H) \geq T$. Otherwise, it should be assumed that $t_{indef}(H) = T$.

Next, the situation of P_6 loss occurrence was considered. In this case, a situation arises of replacing one or more tasks for the design and implementation of an IS element, the descriptions of which are absent or not determined with sufficient accuracy, with the corresponding number of tasks from the development team backlog, the descriptions of which are fully determined. The estimate of the average cost $S_{av.nc.w}$ of a unit of work not performed as a result of the transfer of resources, in this case should be calculated using formulas (2.34), (2.35). The value of the parameter W_{sum} should be calculated using formula (2.38).

To calculate the value of the parameter $S_{reserve}$, it was proposed to consider the creation of the development team backlog as a task for the performance of which the appropriate personnel and resources are involved. The result of such a task is the development team backlog as a team activity plan for the time period T . Therefore, in the general case, $S_{reserve}$ should be calculated using formulas:

$$\hat{S}_{reserve} = \sum_{i=1}^m k_{pi} q_{pi}(T) St_{pi}(T) + \sum_{j=1}^n k_{pj} q_{pj}(T) Cl_{pj}(T), \quad (2.40)$$

$$\hat{S}_{reserve} = \sum_{i=1}^m k_{pi} q_{pi}(T) St_{pi}(T) + C_{Cl}(p), \quad (2.41)$$

where p – conditional resources (IT project staff and other resources) that were used to perform the work on planning the activities of the IT project team.

The cost of the completed work reserve $S_{compl.w.reserve}$ in this case is equal to the cost of tasks from the backlog reserve of the development team that were performed instead of tasks, which descriptions were recognized as incomplete. In this case, developers can only operate with an estimate of the value $S_{compl.w.reserve}$. This estimate, by analogy with (2.17), (2.18), is proposed to be determined as follows:

$$\hat{S}_{compl.w.reserve} = \sum_{res=1}^r \sum_{i=1}^m k_{resi} q_{resi}(T) St_{resi}(T) + \sum_{res=1}^r \sum_{j=1}^n k_{resj} q_{resj}(T) Cl_{resj}(T), \quad (2.42)$$

$$\hat{S}_{compl.w.reserve} = \sum_{res=1}^r \sum_{i=1}^m k_{resi} q_{resi}(T) St_{resi}(T) + C_{cl}(z), \quad (2.43)$$

where res – the IS element, the description of which is decided to be selected from the backlog reserve of the development team and used instead of the incomplete description of the IS element, which cannot be implemented during the current iteration. The value of the quantity q_{ri} is set by a formula similar to (2.16).

Based on formulas (2.34), (2.35), (2.38) and (2.40)–(2.43), a detailed description of the loss model P_6 (2.6) for the case when the task for replacement is selected from the backlog of the development team is proposed to be presented as follows

$$P_6 = \left(\frac{R(H)t \left(\sum_{i=1}^m k_{di} q_{di}(T) St_{di}(T) + \sum_{j=1}^n k_{dj} q_{dj}(T) Cl_{dj}(T) \right) t_{indef}(H)}{Tt_c} + \right. \\ \left. + tk_{\%} P_{project} + \sum_{i=1}^m k_{pi} q_{pi}(T) St_{pi}(T) + \sum_{j=1}^n k_{pj} q_{pj}(T) Cl_{pj}(T) \right) - \\ - \sum_{res=1}^r \sum_{i=1}^m k_{resi} q_{resi}(T) St_{resi}(T) - \sum_{res=1}^r \sum_{j=1}^n k_{resj} q_{resj}(T) Cl_{resj}(T) \quad (2.44)$$

provided that $t_{indef}(H) \geq T$. Otherwise, it should be assumed that $t_{indef}(H) = T$.

When developing detailed descriptions of costs P_7 (2.7) and P_8 (2.8), it was taken into account that the product $S_{res,av}^{t_{st}} \times I$, which determined the total resource costs for the normalized time period t_{st}^* was already described by expressions (2.29)–(2.31). In this case, the difference is that the set of resources that were planned to be used to perform $R(H)$ of the IS A is elements not transferred to the IS B IT project and is not used in any of the IT projects. Therefore, it was proposed to present the detailed descriptions of costs P_7 (2.7) and P_8 (2.8) as follows

$$P_7 = \left(\sum_{d=1}^{R(H) \times IS_A} S_{redp,d} \right) \times t_{indef}(H) R(H) \quad (2.45)$$

provided that $t_{indef}(H) \geq T$;

$$P_8 = \left(\sum_{d=1}^{R(H) \times IS_A} S_{redp,d} \right) \times TR(H) \quad (2.46)$$

provided that $t_{indef}(H) < T$.

In this case, the value of each element of the sum $\sum_{d=1}^{R(H)} \hat{S}_{redp,d}^{IS_A}$ is calculated using expressions (2.29) or (2.30).

For a detailed description of losses due to the consequences Ex_5 and Ex_6 , which are the results of the consequences Ex_2 , Ex_3 and Ex_4 , it was necessary to conduct additional research. The main goal of these studies was to study the features of planning multiple iterations of an IT project for creating an IS due to an increase or decrease in the duration of a separate iteration T . Therefore, in this study, a detailed description of losses due to the consequences Ex_5 and Ex_6 was not considered.

2.6 Experimental verification of the obtained results

2.6.1 Description of the initial data

To verify the results obtained, it was proposed to use the data of the IT project for the development of the service "Electronic Compulsory Motor Third Party Liability Insurance Policy" (CMTPL). The sale of this type of insurance policy was officially launched in Ukraine in accordance with the changes in the legislation of February 7, 2018. In this regard, the problem of developing and implementing IT services that allow automating such activities arose.

The IT company Profitsoft, a leading software development company that also has deep expertise in the insurance domain and innovations in the InsurTech industry, was engaged in solving this problem. It has been implementing IT projects for the development and maintenance of software for more than 20 years. In the Ukrainian market, Profitsoft is known as the developer of the IS "Comprehensive System for Automation of Insurance Company Work – Profitsoft" (KSASK), which is successfully used by leading insurance companies [16].

The KSASK development began in 2006, when the functional module "Front-office" was introduced. The following functions were developed within this module [17]: "Calculators", "Sales network", "Forms", "Commission", "Security".

In 2009, the system was supplemented with the functional module "Back-office". The following functions were developed within this module [17]: "Metadata-based interface", "Finance", "Settlement", "Reinsurance".

In 2013, KSASK implemented designers – tools that allow to configure any calculations, documents and output forms without the need to modify the entire system or its individual modules. In addition, the functional modules "Underwriting" and "Import of contracts" [17] were implemented during this period.

In 2014–2019, KSASK gradually transitioned from a modular to a service architecture. In addition, the following modules were developed in the system [17]: "Online-shop" (aka "Internet store"), "Settlement designer", "Integrations" (a module that supports data exchange via API with external platforms (EWA, PrivatBank, etc.)).

By 2021, KSASK had developed as follows [17]:

- calculator designer was developed;
- "Financial monitoring" and "Quote requests" services were developed;
- mobile version of KSASK software was developed;
- transition from a service to a microservice architecture of the system began.

In 2022–2024, KSASK was adapted to the new requirements of the National Bank of Ukraine, which came into force in 2024 in accordance with the Law "On Insurance" (No. 1909-IX). As a result of the adaptation, KSASK began to provide automation of all business processes of insurance companies, in particular, accounting of contracts, financial monitoring, formation and submission of reports in accordance with the requirements of the regulator. The system supports electronic signatures and integration with key state registers and services, ensuring reliability, transparency and control of all operations. KSASK has already been updated to comply with new classifications of insurance products and accounting by business lines, which makes it a reliable solution for meeting the requirements of the National Bank of Ukraine [18].

At the time of the initiation of the IT project for the development of the "Electronic CMTPL Policy" service (2017), the basic version of KSASK consisted of the following functional modules:

- "Insurance Calculator";
- "Accounting for Contracts";
- "Import of Data Lists";
- "Underwriting";
- "Accounting for Forms";
- "Commissions";
- "Finances";
- "Settlement";
- "VMI" (voluntary medical insurance);
- "Reporting to the Motor (Transport) Insurance Bureau of Ukraine (MTBU)";
- "Online Store";
- "Administration";
- "CRM";
- "Business Processes";
- "Releases";
- "Help".

The service "Electronic CMTPL Policy", the development of which was the main goal of the IT project, according to the requirements set for it, was to consist of the following functions:

- "Filling out an electronic policy";
- "Checking an electronic policy";
- "Issuing an electronic policy";
- "Selling an electronic policy";
- "Printing an electronic policy agreement";
- "Checking the validity period of an electronic policy";
- "Paying for an electronic policy";
- "Calculating the commission fee for an electronic policy";
- "Authorization via SMS".

The emergence of a business opportunity to sell an electronic CMTPL policy has led to the emergence of a corresponding business need in insurance companies. Therefore, ProfITsoft has identified the minimization of the time for the development and implementation of this service as an additional condition for the successful implementation of the IT project for the development of the "Electronic CMTPL Policy" service. It was believed that the fulfillment of this condition would allow insurance companies-owners of KSASK to gain a temporary business advantage over competitors (until the implementation of similar services from other developers). The main way to fulfill this condition during the initiation of the project was to reuse the results of the development of other functional KSASK modules. To search for functional KSASK modules and their functions that can be reused in the IT project for the development of the "Electronic CMTPL Policy" service, the information technology of the IT service provision management system was used [19]. The theoretical foundations and features of the implementation of this technology are discussed in detail in [20]. At the moment, this technology in the IT company ProfITsoft has received further development and has become a specialized mechanism "Functional Exchange Fund". This mechanism of interaction between insurance companies-users of KSASK provides the possibility of transferring, exchanging, reusing and collectively developing the modified functionality of the system [21].

The results of the search for functional modules and individual functions of KSASK suitable for reuse in the IT project for the development of the "Electronic CMTPL Policy" service are given in **Table 2.1**.

As a result of the application of information technology for managing the IT service provision system during the initiation of the IT project, an assessment of the profitability of independent service development or reuse of one of the KSASK IT service search results (functions) listed in **Table 2.1** was carried out.

The ISO/IEC 1926:2001 criteria system was used for the assessment [22]. The assessment was carried out according to the IT service selection method proposed in [20], which is based on the hierarchy analysis method.

Table 2.1 Results of the search for functional modules and individual functions of KSASK suitable for reuse in the IT project for the development of the "Electronic CMTPL Policy" service

Search object	IT service search result	Search result for the name/part of the name that matches, IT service function
Full match with the list of functional requirements	No IT services found that fully implement all functional requirements	
IT services that fully implement individual functional requirements	"Online store"	"Authorization via SMS"
IT services that implement individual parts of the functional requirements	"Online store"	- "Filling"; - "Policy registration"; - "Registration"; - "Sale"; - "Payment"; - "Print"; - "SMS"; - "Authorization via SMS"
	"Accounting for Contracts"	- "Verification"; - "Contract verification"; - "Print"; - "Term"; - "Print contract"
	"Commissions"	- "Commissions"; - "Calculation"; - "Print"; - "Verification"
	"Underwriting"	- "Verification"; - "Policy verification"; - "Policy"

Source: [20]

The assessment resulted in the following recommendations [20]:

- the best solution according to the selected criteria is the development of the "Electronic CMTPL Policy" service "from scratch";

– for reuse during the implementation of the IT project for the development of the "Electronic CMTPL Policy" service, the most suitable are the functions of the "Contract Accounting" and "Online Store" modules.

To experimentally verify the results obtained in the study, it was proposed to check the correctness of these recommendations at different stages of the IT project life cycle for the development of the "Electronic CMTPL Policy" service. In particular, it was proposed to consider solutions for the formation and adjustment of IT project plans, based on estimates of losses that may arise as a result of identifying the fact of incompleteness of the description of a separate IT project function.

2.6.2 Description of the assessment of losses due to the replacement of incomplete information with information about a similar element of the information system at the stage of IT project initiation

First of all, the assessment of losses due to the replacement of incomplete information about the functions of the "Electronic CMTPL Policy" service with information about similar functions of the KSASK at the stage of IT project initiation was considered. For comparison, an assessment of the costs of developing the "Electronic CMTPL Policy" service "from scratch" was also carried out.

To calculate the costs of developing the "Electronic CMTPL Policy" service "from scratch", it was proposed to use:

- COCOMO model [15];
- a simplified method of functional points [23], which allows to estimate the efforts for the development of IS elements in conditions of almost complete absence of information about new IS functions;
- a backfiring technique for converting the number of functional points into the number of lines of source code (as an argument of the COCOMO model).

The values of the parameters of the simplified functional point method, the roll-back coefficient k and the COCOMO model are given in **Table 2.2**.

As a result of calculating the number of functional points of the "Electronic CMTPL Policy" service, it was obtained

$$FP = (C_1 + C_2 + C_3)^{2.35} = (9 + 9 + 15)^{2.35} = 33^{2.35} \approx 3702.61 \text{ (functional points).}$$

The results of calculating the estimates of efforts, time costs and personnel requirements for the IT project for developing the "Electronic CMTPL Policy" service "from scratch" using the COCOMO model are given in **Table 2.3**.

Table 2.2 Values of the parameters of the simplified functional point method and the rollback coefficient k and the COCOMO model

Element name	Element value	Explanation
C_1	9	Estimation of the project scale ("Custom application")
C_2	9	Estimation of the users of the design object ("Commercial project")
C_3	15	Estimation of the type of design object ("Software for publicly available services")
k_{DB}	16	Maximum normative number of SQL commands per one functional point
k_{SW}	46	Maximum normative number of Java commands per one functional point
B_d	1.12	Locked mode of operation (application system development)

Source: [23]

Table 2.3 Results of calculating the characteristics of the IT project for developing the "Electronic CMTPL Policy" service "from scratch"

Characteristic name	Characteristic value	Units of measurement
Service database development		
E_{DB}	174.37	Person-months
T_{DB}	17.77	Months
SS_{DB}	9.81	Full-time and part-time employees
Service software development		
E_{SW}	524.49	Person-months
T_{SW}	27.08	Months
SS_{SW}	19.51	Full-time and part-time employees

It should be remembered that the estimates of the characteristics of an IT project at the stage of its initiation may significantly deviate from the real characteristics of the same project. According to the research results given in [15], such estimates may exceed the value of the real characteristic by four times. Based on this, it was proposed to adjust the characteristics of the IT project given in **Table 2.3** by dividing the number of functional points by four (the result is 925.6525 functional points). The results of the adjustment are given in **Table 2.4**.

The data in **Table 2.4** show that even in the case of organizing parallel development of the database and software for the "Electronic CMTPL Policy" service, the duration of the IT project will be approximately 1 year and 4 months. This duration was recognized as unacceptable from the point of view of meeting the

business needs of insurance companies-users of KSASK (even considering that the decision to develop the "Electronic CMTPL Policy" service "from scratch" was recognized as the best according to the criteria for the quality of software product development).

Table 2.4 Adjusted results of calculations of the characteristics of the IT project of the development the "Electronic CMTPL Policy" service "from scratch"

Characteristic name	Characteristic value	Units of measurement
Service database development		
E_{DB}	40.67	Person-months
T_{DB}	10.22	Months
SS_{DB}	3.98	Full-time and part-time employees
Service software development		
E_{SW}	123.28	Person-months
T_{SW}	15.58	Months
SS_{SW}	7.914	Full-time and part-time employees

Next, an assessment of possible losses due to replacing incomplete information about the functions of the "Electronic CMTPL Policy" service with information about similar functions of the "Online Store" module of KSASK at the stage of IT project initiation was carried out. This option was chosen because, according to the search results given in **Table 2.1**, it showed the largest number of matches.

It was decided to exclude from further consideration those search results given in **Table 2.1**, which are parts of other search results for functions of the same module. The result of excluding duplicate search results and further comparison of the names of the functions of the service "Electronic CMTPL Policy" and the functions of the module "Online Store" is given in **Table 2.5**. In addition, **Table 2.5** indicates the estimates of the number of functional points for each of the functions of the service being developed and the value of the number of functional points for the functions of the module.

Since at the stage of IT project initiation, the executors only know the names of individual functions of the service being developed, it was proposed to consider the distribution of efforts for the development of individual functions of the service as uniform. Therefore, for further calculations, the value of the effort spent on the development of each individual function of the service was taken as 102.85 (functional points).

Table 2.5 Comparison of the names of the functions of the "Electronic CMTPL Policy" service and the "Online Store" module

No.	Service function name	Number of service function points	Name of the module function highlighted in the search results	Number of functional points of the module function
1	"Filling out an electronic policy"	102.85	"Filling out an insurance contract"	97
2	"Issuing an electronic policy"	102.85	"Issuing an insurance policy"	84
3	"Selling an electronic policy"	102.85	"Selling an order"	157
4	"Printing an electronic policy agreement"	102.85	"Printing the source document"	48
5	"Paying for an electronic policy"	102.85	"Paying for the insurance contract"	142
6	"Authorization via SMS"	102.85	"Authorization via SMS"	134

For further calculations, it was proposed to use:

- the COCOMO model [15];
- the improved functional points method [24], which allows to estimate the efforts for the development of IS elements in the conditions of reuse of individual system functions;
- the backfiring technique for converting the number of functional points into the number of lines of source code (as an argument of the COCOMO model).

To determine the number of developers $q_{zi}(T)$ required to eliminate the situation under consideration, the expression (16) was used, which in this case took the form

$$q_{zi}(T) = \frac{2.4(KLOC_z)^{B_z}}{T} = \frac{2.4(k_z FP_{add,z} / 1000)^{B_z}}{T}, \quad (2.47)$$

where $KLOC_z$ - the number of thousands of lines of source code, the creation of which is necessary for additional processing with information that replaces; k_z - the rollback coefficient, which determines the normative number of lines of source code for the implementation of one functional point of the z -th IS function; $FP_{add,z}$ - the number of functional points as an estimate of the efforts for additional processing with information that replaces an incomplete description of the z -th IS function; B_z - an indicator, the value of which is determined by the selected mode of additional processing with information that replaces an incomplete description of the z -th IS function.

To calculate the value of $FP_{add,z}$ in [24] it was proposed to use the expression

$$FP_{add,z} = FP_z \times \left(\frac{|Name_d - Name_z|}{|Name_z|} \right), \quad (2.48)$$

where FP_z – the number of functional points, which is an estimate of the efforts to develop the z -th function of the developed functional problem; $Name_d$ – the name of the d -th function of the developed problem, the description of which is incomplete; $Name_z$ – the name of the z -th function of the developed functional problem.

Based on formula (2.48), to calculate the $FP_{add,z}$ value it was necessary to perform stemming of the names of individual service functions and their corresponding module functions with the subsequent removal of stopwords. The results of these operations using the Porter stemmer are given in **Table 2.6**.

The FP_z values for each of the module functions, the descriptions of which are proposed to be reused in the IT project for the development of the "Electronic CMTPL Policy" service, are given in **Table 2.5**. These values, as established for case I, are one of the results of the analysis of previous IT projects of the company Profitsoft.

Then, according to expression (2.48) for $z = 1$, there is the following number of functional points

$$\begin{aligned} FP_{add,1} &= 97 \times \left(\frac{|\text{"Fill electron polici"} - \text{"Fill insur contract"}|}{|\text{"Fill insur contract"}|} \right) = \\ &= 97 \times \left(\frac{2}{3} \right) = 64.67 \text{ (functional points)}. \end{aligned}$$

The results of calculating the $FP_{Add,z}$ values for all functions are given in **Table 2.7**.

The values of the elements of formula (2.47), which were used to experimentally verify the results obtained during the assessment of losses in the situation under consideration, are given in **Table 2.8**.

As a result of the calculation according to formula (2.47) for $z = 1$, it is possible to obtain

$$q_1(T) = \frac{2.4(46 \times 64.67 / 1000)^{1.12}}{2} = \frac{2.4(2.97482)^{1.12}}{2} = \frac{8.136}{2} = 4.068 \text{ (man)}.$$

The results of the calculations of the $q_z(T)$ values given in **Table 2.9**.

Table 2.6 An example of the result of using the Porter stemmer for function names

Pre-processing stage	Title of replaced function	Title of replacement function
Original frame name	Filling in an electronic policy	Filling out an insurance contract
Results of using Porter's stemmer	Fill in an electron policy	Fill out an insure contract
Stopword removal results	Fill electron policy ($Name_d$)	Fill out an insure contract ($Name_2$)
Original frame name	Electronic policy registration	Insurance policy registration
Results of using Porter's stemmer	Electron policy register	Insure policy register
Stopword removal results	Electron policy register ($Name_d$)	Insure policy register ($Name_2$)
Original frame name	Selling an electronic policy	Selling an order
Results of using Porter's stemmer	Sell an electron policy	Sell an order
Stopword removal results	Sell electron policy ($Name_d$)	Sell order ($Name_2$)
Original frame name	Print-out an electronic policy	Print-out the original document
Results of using Porter's stemmer	Print-out an electron policy	Print-out the origin document
Stopword removal results	Print-out electron policy ($Name_d$)	Print-out origin document ($Name_2$)
Original frame name	Payment of an electronic policy	Payment of an insurance contract
Results of using Porter's stemmer	Payment of an electron policy	Payment of an insure contract
Stopword removal results	Payment electron policy ($Name_d$)	Payment insure contract ($Name_2$)
Original frame name	Authorization via SMS	Authorization via SMS
Results of using Porter's stemmer	Author via SM	Author via SM
Stopword removal results	Author SM ($Name_d$)	Author SM ($Name_2$)

Table 2.7 Results of calculations of $FP_{Add,z}$ values for all functions, the reuse of which is proposed in the IT project

Parameter		Parameter value				
z	1	2	3	4	5	6
$FP_{Add,z}$	64.67	28	157	32	94.67	0

Table 2.8 Values of the elements of formula (2.47), selected for the quantitative assessment of possible losses

Element name	Element value	Explanation
k_z	46	Maximum standard number of Java commands per functional point
B_z	1.12	Locked mode of operation (application system development)
T	2 months/4 iterations	Iteration duration - 12 working days

Table 2.9 Results of the calculations of the $q_z(T)$ values for all functions which reuse is proposed in the IT project

Parameter		Parameter value				
z	1	2	3	4	5	6
$q_z(T)$	4.068	1.5936	10.992	1.848	6.24	0

Then it was decided to calculate by formula (2.24) the value of losses P_1 from the absence of information in case of need for additional processing with information that replaces. The values of the elements of formula (2.24), which were used to experimentally verify the obtained results when estimating losses in the situation under consideration, are given in **Table 2.10**.

The standard salary of a specialist at Profitsoft was determined based on the labor costs of the KSASK technical support team given in [25].

Based on these values, the losses P_1 are equal to

$$P_1 = 1 \times \left(\sum_{z=1}^6 \sum_{i=1}^m k_{zi} q_{zi}(T) St_{zi}(T) + \sum_{z=1}^1 C_{Cl}(z) \right) = 1 \times \left(k_z \times T \times \sum_{z=1}^6 q_z(T) St_z(T) + \sum_{z=1}^6 C_{Cl}(z) \right) =$$

$$= 1 \times (500 \times 4 \times 24.7416 + 240) = \$49723.2.$$

Next, it was decided to calculate the value of the losses P_2 from the absence of information using formula (2.25) if there is a difference in the cost of the

information being replaced and the information being replaced. Since the z -th functions, the descriptions of which are proposed to be used instead of the descriptions of the d -th functions, were developed in previous IT projects of ProfITsoft, it was decided to use the calculation rule established for case I.

Table 2.10 Values of the elements of formula (2.24), selected for quantitative assessment of possible losses

Element name	Element value	Explanation
$C_{C_i}(z)$	\$40	Electricity costs and additional costs
$St_{z_i}(T)$	1	The fact that the developer is in the process of creating the z -th element of the IS during the time period T at the i -th position
k_z	\$500	The standard salary of a company specialist during the iteration
$R(H)$	6	The number of cases of incompleteness or lack of information
k	1	The value of the quality coefficient of management decision-making

First, an estimate of the personnel requirement was determined. For the example under consideration, formula (2.16) took the form:

$$q_z(T) = \frac{2.4(KLOC_z)^{B_z}}{T} = \frac{2.4(k_z FP_z / 1000)^{B_z}}{T}, \quad (2.49)$$

$$q_d(T) = \frac{2.4(KLOC_d)^{B_d}}{T} = \frac{2.4(k_d FP_d / 1000)^{B_d}}{T}, \quad (2.50)$$

where $KLOC_d$ – the number of thousands of lines of source code that must be created to implement the d -th function; k_d – rollback coefficient, which determines the normative number of lines of derived code for the implementation of one functional point of the d -th IS function; FP_d – the number of functional points as an estimate of the efforts to develop the d -th IS function; B_d – an indicator, the value of which is determined by the selected mode of development of the d -th IS function.

The FP_d values for the service functions that are proposed to be replaced are given in **Table 2.5**. The $q_z(T)$ values are given in **Table 2.9**.

The values of the elements of formula (2.50), which were used to experimentally verify the obtained results when assessing losses in the situation under consideration, are given in **Table 2.11**.

As a result of the calculation by formula (2.50) for each of the d -th functions, the descriptions of which are incomplete

$$q_d(T) = \frac{2.4(k_d FP_d / 1000)^{B_d}}{T} = \frac{2.4(46 \times 102.85 / 1000)^{1.12}}{2} =$$

$$= \frac{2.4(4.7311)^{1.12}}{2} \approx \frac{13.683}{2} \approx 6.84 \text{ (man).}$$

The values of the estimates of the value of the information S_{repm} being replaced were established based on the results of the analysis of previously completed IT projects. From these results it turned out that on average for each individual function the value S_{repm} is equal to \$12000. The value of the estimate of the replaced information was calculated by formula (2.20). The values of the elements of formula (2.20) used for the calculation are given in **Table 2.12**.

Table 2.11 The values of the elements of formula (2.50), selected for quantitative assessment of possible losses

Element name	Element value	Explanation
k_d	56	Maximum regulatory number of Java commands per functional point
B_d	1.12	Locked mode of operation (application system development)

Table 2.12 The values of the elements of formula (2.20) selected for the calculation of the estimates of the values of the replaced information

Element name	Element value	Explanation
$C_{Cl}(z); C_{Cl}(d)$	\$40	Electricity costs and additional costs
$St_i(T); St_{di}(T)$	1	The fact that the developer is in the process of creating the z-th element of the IS during the time period T at the i -th position
$k_r; k_d$	\$500	The standard salary of a company specialist during the iteration
k	1	The value of the quality coefficient of management decision-making

Then, according to formula (2.25), the value of losses P_2 is equal to

$$P_2 = 1 \times \left(\sum_{r=1}^6 \hat{S}_{repm,r} - \sum_{d=1}^6 \hat{S}_{repd,d} \right) = 1 \times (6 \times (12000 + 40) - 6 \times (13680 + 40)) =$$

$$= 1 \times (6 \times 12040 - 6 \times 13720) = -\$10080.$$

As a result of checking the fulfillment of condition (2.26) of the feasibility of choosing the consequence Ex_1 , it was established that

$$P_{sum} = P_1 + P_2 = 49723.2 - 10080 = \$39643.2;$$

$$\sum_{d=1}^6 \hat{S}_{repd,d} = \$81080; \$39643.2 \leq \$81080,$$

that is, the reuse of software descriptions of the functions of the "Online store" module to eliminate the incompleteness of the software descriptions of the functions of the "Electronic CMTPL Policy" service is feasible subject to the following conditions for the implementation of the IT project for the development of the "Electronic CMTPL Policy" service:

- the total duration of the IT project, according to the estimate, should be 2 months;
- the total number of personnel for the implementation of the IT project, according to the estimate, should be 25 people (5 teams of 5 people each [25]);
- during the implementation of the IT project, the same software development tools will be used as in previous IT projects of the ProfITsoft company.

2.7 Discussion of the results of the loss assessment for the considered solution options

To eliminate the situation that arose during the initiation of the IT project for the development of the "Electronic CMTPL Policy" service due to incomplete descriptions of the functions of this service, two management solution options were proposed. The first option, proposed by the staff of the Department of Information Control Systems (ICS) of the Kharkiv National University of Radio Electronics (NURE), was based on the results of evaluating alternatives according to the ISO/IEC 1926:2001 software product quality criteria system. According to this option, it was proposed to abandon the reuse of previously developed KSASK functions and develop the service "from scratch". The alternative proposed by the authors of this study was to use descriptions of previously developed functions of the KSASK module "Internet Store" instead of incomplete descriptions of the service functions.

The results of the comparison of these two solution options are given in **Table 2.13**.

Based on the estimates given in **Table 2.13**, in order to minimize the total financial costs for wages and reduce the total time of implementation of the IT project, it is necessary to choose an alternative management solution.

But such a solution causes a certain increase in the risk of deterioration in the quality of the service being developed. In particular, after the implementation of the IT project for the development of the service "Electronic CMTPL Policy"

according to the alternative option, the following sets of errors may exist in the service code:

- errors remaining in the software of reused KSASK functions after its adaptation to the requirements of the service;
- errors that arose in the software of the service functions during its adaptation to the requirements of the service.

Table 2.13 Results of the comparison of the characteristics of the proposed solution options for the IT project

Characteristics	Characteristic value	
	First option	Alternative
Total duration of the IT project	15.58 months	2 months
Requirement for personnel of the IT project	7.914 people	25 people
Nominal value of the salary of a company specialist during the iteration	\$500	\$500
Average number of iterations per month	2	2
Estimation of the total financial costs for salaries in the IT project	\$123300.12	\$50000

Of course, some subsets of these sets of errors will be detected during testing of the service software. But testing, as a rule, does not guarantee the detection of the full set of errors (especially in IT projects with a limited duration). However, the alternative solution involves the possibility of increasing the time spent on testing the service software (which will lead to an increase in the total time spent on the implementation of the IT project). Even if the total duration of the IT project is increased by 2 months, the total salary costs will not exceed the similar costs for the first option.

It should also be noted that the correctness of the alternative solution is emphasized by the fact that in the current (2024–2025) KSASK version, the "Electronic CMTPL Policy" service is implemented as a widget of the "Online Store" service [26].

Thus, in the course of this study, detailed models of production losses due to the lack of information were developed, adapted to the specifics of the IT company and its IT projects. The developed models allow to quantitatively assess the costs that arise during the initiation, planning and implementation of the IT project of creating an IS during the elimination of incomplete descriptions of individual system elements. These costs are the result of one of the following possible consequences:

- consequence Ex_1 : "Replacing missing information with information about a similar IS element that is reused";

- consequence Ex_2 : "Concentration of IT company personnel and IT infrastructure elements on the design or development of other IS elements";
- consequence Ex_3 : "Transferring IT company personnel and IT infrastructure elements to the design or development of other elements of the same IS";
- consequence Ex_4 : "Refusal to use IT company personnel and IT infrastructure elements due to the lack of information about IS elements".

The results obtained take into account the iterative nature of planning the activities of IT project teams and can be used for the majority of modern Agile and hybrid IT project management methodologies.

Unlike modern models and methods of IT project financial management [2], the implementation of the developed detailed models allows to estimate losses and make decisions on IT project cost management in the absence of a priori defined descriptions of individual IS elements (functions), which is the expected result of the project. The use of these models allows to choose solutions during the implementation of the IT project that minimize additional losses that arise in the following situations:

- a priori incompleteness of the results of the collection and analysis of functional requirements for the created system;
- the occurrence of a significant number of changes in the descriptions of functional requirements and IS architecture and its elements due to changes in the business processes of the automation object;
- the occurrence of changes in the descriptions of functional requirements and IS architecture and its elements due to the elimination of errors made during the design and implementation of individual elements of the created IS.

However, the use of the developed detailed loss models is not free from some limitations. The main limitation of these is the orientation of the developed detailed loss models to the assessment of direct losses exclusively during the period of time considered at the current stage of the project life cycle (the total duration of the IT project during the initiation and start of its planning; the total duration of the current iteration during the planning of the activities of the team of performers and the implementation of the IT project). The developed models practically do not take into account indirect losses that arise after making a decision on the selected consequence. Such a limitation during the practical application of the developed models may lead to the selection of a solution option that will be a local, rather than a global minimum of losses for the entire IT project or its individual iterations. Another limitation of the application of the obtained models is the implicit assumption of the constant availability of the relevant IT project performers and other resources. It is believed that these resources can be used at any time to eliminate the specified consequences of detecting incomplete information. In fact, the results of accounting for

the workload of personnel and other material resources of an IT project can significantly change the estimates of possible losses that arise during the elimination of the identified incompleteness of information about the elements of the created IS.

The main drawback of the results of the study is the lack of technological sophistication of the developed detailed models. It should be remembered that the use of the obtained mathematical models in the IT company management system is impossible without conducting scientific and applied research in the field of developing relevant information technologies. It should also be recognized that the existing information technologies for managing an IT company and its IT projects practically do not allow using the obtained detailed models for their own improvement. There is a need to develop fundamentally new information technologies for managing an IT company, which would combine the use of the obtained detailed loss models with the capabilities of managing the work of an IT company in a time regime close to real.

Therefore, the following main areas of further prospective research in the field of IT project cost management were proposed to be identified:

- research on the improvement and development of the obtained detailed loss models and the general concept of cause-and-effect management of IT project costs, taking into account the characteristics of IT projects and programs;
- research on the possibility of integrating detailed loss models into existing IS and information technologies for managing an IT company and its IT projects;
- research on the development of fundamentally new IS and information technologies for managing an IT company, including using modern methods and tools of artificial intelligence.

2.8 Conclusions

As a result of the study, the features of the design processes of IS elements in modern IT companies were identified. The situational-consequential scheme of the formation of losses in production as a result of the lack of information was adapted to these features. The adaptation results show that the use of the adapted scheme for estimating costs and making decisions on managing iterations of the implementation of the IS creation IT project in conditions of incompleteness or absence of descriptions of individual IS elements is possible.

Detailed models of losses from the lack of information have been developed if additional processing of the information that replaces is necessary, as well as losses from the lack of information if there is a difference in the cost of the information that is replaced and the information that is replaced. The obtained models can be

used with various models and methods of assessing efforts to implement the IS creation IT project.

An experimental verification of the obtained results has been carried out. The course and results of the calculations confirm the possibility of using the adapted scheme and the developed detailed models for estimating losses during the initiation, planning and implementation of the IS creation IT project.

Conflict of interest statement

The authors declare that there is no conflict of interest in relation to this paper, as well as the published research results, including the financial aspects of conducting the research, obtaining and using its results, as well as any non-financial personal relationships.

Use of artificial intelligence statement

The authors declare that they did not use artificial intelligence tools in preparing this manuscript.

References

1. ISO/IEC/IEEE Standard No 15288:2015 (2015). Systems and software engineering – System life cycle processes. ISO/IEC/IEEE International Standard. <https://doi.org/10.1109/IEEESTD.2015.7106435>
2. Nastanova do zvodu znan z upravlinnia proiektamy. Nastanova PMBOK (2021). Project Management Institute, Inc. Available at: https://learn.ztu.edu.ua/pluginfile.php/274061/mod_resource/content/1/PMBOK7_Ukr_ForPersonalUseOnly.pdf
3. Valadares, F. S., Moura, N. C. S., Pereira, T. N. F., Arantes, M. D. O. (2024). Identification of the Main Traditional Project Management Methods Through a Systematic Literature Review. *International Journal of Advanced Computer Science and Applications*, 15 (6). <https://doi.org/10.14569/ijacsa.2024.0150697>
4. Coali, A., Gambardella, A., Novelli, E. (2024). Scientific decision-making, project selection and longer-term outcomes. *Research Policy*, 53 (6), 105022. <https://doi.org/10.1016/j.respol.2024.105022>

5. Khosravizadeh, O., Maleki, A., Ahadinezhad, B., Shahsavari, S., Amerzadeh, M., Tazekand, N. M. (2022). Developing decision model for the outsourcing of medical service delivery in the public hospitals. *BMC Health Services Research*, 22 (1). <https://doi.org/10.1186/s12913-022-07509-1>
6. Zhou, M. (2024). Optimization of University Scientific Research Project Management Resources Based on Genetic Algorithm. *Disruptive Human Resource Management*, 128–138. <https://doi.org/10.3233/atde240422>
7. Chen, K., Ou, W., Zeng, C., Wu, H., Ye, M. (2024). Research on Cost Management of Power Transmission and Transformation Project Based on BIM Technology. *2024 International Conference on Machine Intelligence and Digital Applications*, 141–147. <https://doi.org/10.1145/3662739.3670223>
8. Shorikov, A. F., Butsenko, E. V.; Kahraman, C., Cevik Onar, S., Cebi, S., Oztayasi, B., Tolga, A. C., Ucal Sari, I. (Eds.) (2024). Description of the Structure and Functions of an Intelligent Software System for Optimizing Adaptive Project Control with Fuzzy Data. *Intelligent and Fuzzy Systems*, 488–495. https://doi.org/10.1007/978-3-031-67192-0_55
9. Rahman, H. U., da Silva, A. R., Alzayed, A., Raza, M. (2024). A Systematic Literature Review on Software Maintenance Offshoring Decisions. *Information and Software Technology*, 172, 107475. <https://doi.org/10.1016/j.infsof.2024.107475>
10. Sakka, A., Kourjeh, M., Kraiem, I. B. (2023). An IT projects' conceptual model to facilitate upstream decision-making: project management method selection. *International Transactions in Operational Research*, 30 (6), 3687–3718. <https://doi.org/10.1111/itor.13231>
11. Butt, S., Khan, S. U. R., Hussain, S., Wang, W.-L. (2023). A conceptual model supporting decision-making for test automation in Agile-based Software Development. *Data & Knowledge Engineering*, 144, 102111. <https://doi.org/10.1016/j.datak.2022.102111>
12. Svyrydov, V. V., Moroz, B. I. (1992). *Orhanyzatsiia protsessov obrabotkyinformatsii po kryteriyam tsennosti y starenia v ASU*. Kharkiv: Vydavnytstvo "Osnova" pry Kharkivskomu derzhavnomu universyteti, 112.
13. Calderon-Tellez, J. A., Bell, G., Herrera, M. M., Sato, C. (2023). Project management and system dynamics modelling: Time to connect with innovation and sustainability. *Systems Research and Behavioral Science*, 41 (1), 3–29. <https://doi.org/10.1002/sres.2926>
14. Levykin, V. M., Ievlanov, M. V., Kernosov, M. A. (2014). *Patterny proektirovaniia trebovaniia k informatsionnym sistemam: modelirovanie i primenenie*. Kharkiv: KhNURE, 320.

15. COCOMO II Model Definition Manual. Available at: https://athena.ecs.csus.edu/~buckley/CSc231_files/Cocomo_II_Manual.pdf?ref=thetack.technology
16. Profitsoft. Available at: <https://profitsoft.dev/ua/index.php>
17. Product history. Profitsoft. Available at: <https://ksask.profitsoft.dev/en/#history>
18. Key features. Profitsoft. Available at: <https://ksask.profitsoft.dev/en/#characteristics>
19. Luriev, I. O. (2021). Developing IT Service Management Information Technology. *Management Information System and Devises*, 177, 57–63. <https://doi.org/10.30837/0135-1710.2021.177.057>
20. Luriev, I. O. (2019). Methods, models and information technology of control of IT-services provision system [Extended abstract of PhD thesis; Kharkiv National University of Radio Electronics]. Available at: <https://openarchive.nure.ua/entities/publication/117efb1f-4d38-40f9-b3fa-dac4cb783736>
21. Polozhennia pro fond obminu funktsionalom. Profitsoft. Available at: <https://ksask.profitsoft.dev/polozhennya-pro-fond-obminu-funkczionalom/>
22. ISO/IEC/IEEE Standard No 9126-1 (2001). Software engineering – Product quality – Part 1: Quality model. ISO/IEC/IEEE International Standard.
23. Na start! Vnimanie! I? (2005). ITCua. Available at: http://itc.ua/articles/na_start_vnimanie_i_21814/
24. Levykin, V., Ievlanov, M., Neumyvakina, O., Levykin, I., Nakonechnyi, A. (2024). Estimation of IT-project efforts for information system creation in the conditions of re-use of its functions. *Eastern-European Journal of Enterprise Technologies*, 2 (2 (128)), 6–19. <https://doi.org/10.15587/1729-4061.2024.301227>
25. Technical support. Profitsoft. Integrated Insurance Company Automation System. Available at: <https://ksask.profitsoft.dev/en/#prices>
26. Online store and customer access (IMCD). Profitsoft. Available at: <https://ksask.profitsoft.dev/client-access/>

CHAPTER 3

Method of automatic assignment of tasks to IT project performers

Tetiana Borysenko
Maksym Yevlanov
Konstantin Petrov
Viktor Borysenko

Abstract

The aim of the study is to develop a method for solving the problem of assigning IT project tasks to its performers.

The object of the study is the process of planning an IT project.

During the study, the problem of assigning IT project tasks to its performers was solved. It is shown that in recent years heuristic solutions to this problem have gained popularity. The main requirements for such a solution are determined. The results of the analysis of modern research confirm the relevance of scientific and applied works devoted to solving the problem of assigning IT project tasks to its performers.

According to the results of the study, it is proposed to represent the problem of assigning IT project tasks to its performers as a type of classification problem. This representation allowed using a polynomial Bayesian classifier to solve this problem. In the process of the study, the classification rule with a minimum error and calculations of the main elements of this rule were adapted to the specifics of the problem. An additional classification condition was established that prevents the maximum load of the IT project performer from being exceeded. Based on the results of the adaptation, the algorithm for solving the classification problem, which uses this adapted classifier, was modified.

A general description of the method for solving the problem of assigning IT project tasks to its performers was developed. An algorithm for implementing this method was developed for a detailed description of the content of individual stages. The scheme of this algorithm and the proposed descriptions of the main steps of the algorithm determine the features of its implementation both as a methodology

for applying the obtained solutions in the current management of an IT project and as a specialized information technology.

To verify the operability of the obtained results, an experimental test of the method and its implementation algorithm was conducted during the management of one of the IT projects of an outsourcing IT company. The test results indicate the feasibility of using the developed method to solve the problem of assigning IT project tasks to its performers. The developed method contributes to better project planning, minimizes the administrative burden and helps to avoid delays and errors in project implementation.

Keywords

IT project, task, performer, polynomial Bayesian classifier, effort, team, sprint.

3.1 Introduction

The main difference between IT project management and project management in any other field of human activity is the recognition of personnel as the single most used resource of the project. The emergence and widespread implementation of Agile project management methodologies in the IT industry have not led to significant changes in the existing attitude towards IT project personnel. As early as 2007, it was established that direct costs for an IT project are almost exclusively determined by labor costs [1]. Therefore, personnel management problems remain one of the main problems facing IT project management.

Among these problems, it is worth highlighting the problems that arise during IT project planning. In general, IT project planning includes:

- definition of tasks;
- assessment of task duration;
- placement of tasks on a timeline that visually reflects their sequence, duration, start and end dates;
- according to the results of the placement - assigning resources to each task from available jobs, divided by skill set [2].

Assigning tasks to IT project performers is a complex and time-consuming activity, especially in large and complex projects. During the implementation of this activity, important issues arise regarding the correct distribution of tasks between performers, ensuring the optimization of human resources for task performance, and maximizing the productivity of performers.

It should be noted that modern IT companies use a significant number of various information technologies (IT) to support the IT project planning process. However,

the use of such IT is severely limited by the fact that the personnel of the majority of IT projects is a limited resource. Recognition of this fact led to the formulation of the Resource Constrained Project Scheduling Problem (RCPSP), which belongs to NP-hard problems [3]. Such problems for large and medium-sized IT projects are very often intractable due to the excessive growth of the volume of required calculations. Therefore, basic research suggests two main ways to solve this problem [4]:

- using models and methods that provide accurate solutions to the project planning problem taking into account scarce constraints;
- using heuristic solutions.

As a result, the most common IT planning of IT projects mostly leaves the implementation of the activity of distributing project tasks between performers to the discretion of project management specialists. But this approach leads to subjective errors, irrational use of resources and overloading of individual performers. This often happens in large projects, where the complexity of coordination increases in proportion to the number of tasks and performers. Therefore, the automation of the activity of assigning tasks to performers during the planning and implementation of an IT project remains an extremely important and relevant scientific and applied task. Solving this task will undoubtedly contribute to the optimization of planning processes, avoid excessive administrative burden, ensure more accurate and efficient distribution of work, and also reduce the risk of errors or delays in an IT project.

3.2 Overview of the current state of solving the problem of automated planning of IT projects

The existence of RCPSP as an NP-hard problem has been confirmed in recent years at the level of a number of recent versions of PMBOK [5, 6]. Therefore, a significant amount of modern research is focused on the search and development of models, methods and technologies that allow either to obtain an exact solution or to develop a heuristic solution to this problem.

An example of research aimed at finding an exact solution to RCPSP is the work [2]. The goal of this research is to develop software that creates an optimal schedule that takes into account both technical and resource constraints of the project in order to prevent resource reallocation. In this case, [2] identifies three main factors that can affect the allocation of resources during the project life cycle:

- an increase in the scope of the project, which will require greater efforts to complete it;

- possible unavailability of resources during periods in which they were initially considered available;
- an existence of competition for the same resources between several projects that can be performed simultaneously.

As a result of the research in [2], the creation of software is claimed, which should help the project manager in creating the optimal schedule of this project. This software provides an accurate RCPSP solution in a reasonable period of time in the conditions of performing within the IT company of several projects simultaneously. At the same time, the developed system takes into account the distribution of IT project performers according to their skill set for a better assessment of the duration of project tasks [2]. However, the research result given in [2] is not free from shortcomings that complicate the practical application of the developed software. The main of these shortcomings should be recognized as the need to recalculate all IT project schedules as a result of any change in the availability of IT company personnel. This need leads to an increase in the number of computational operations performed during planning or replanning of IT projects. In addition, this need contradicts the principle of the oncoming wave, according to which detailed planning of the IT project schedule until its completion is considered impractical [5].

Using heuristic solutions to solve the RCPSP requires dividing this problem into a sequence of separate IT project planning management tasks and solving each task separately using the appropriate heuristic. An example of such an approach is research [7]. In this work, it is proposed to develop task and developer profiles to solve the problem of assigning each project task to the most suitable developer. After that, the task is solved by finding the best match of these profiles for assigning tasks to developers. A comparative analysis showed that the Sokal and Snit method [7] should be considered the best for such a search. It should be noted that one of the authors of this study also tried to solve this problem in a similar way. In [8], he proposed a solution to create task and project developer profiles based on a data-logical description of the project work packages. The use of these profiles allowed [8] to propose a method for solving the problem of assigning enterprise employees to tasks of a new IT project, the basis of which is the apparatus of clustering and classification tasks. This method made it possible to assign to the task of the new project an employee whose value of the integral indicator of the quality of work performance was maximum.

But this method of solving the problem of assigning each project task to the most suitable developer is not free from some shortcomings. First, the solutions proposed in [7] and [8] require the creation and constant maintenance of data warehouses in which it is necessary to store and update historical data on the profiles of completed

IT project tasks and project performers. Second, the use of complex data logical descriptions to build profiles leads to significant expenditure of money and time on additional work on the creation, support and maintenance of such profiles during the planning and performing of an IT project. Third, this method leaves open the question of the optimal distribution of tasks among project personnel from the point of view of maximizing and equalizing the employment of project performers.

It must be recognized that the approach to profiling project tasks or work packages and storing the corresponding historical information is characteristic not only of IT projects. In [9], a similar approach is considered for creating a hierarchy of construction project task organization. This study is interesting in that it recognizes the need to create project task profiles not based on special complex datalogical descriptions, but on the results of pre-processing text descriptions of project work packages. Although such a solution complicates the work of processing information about project work packages, it significantly simplifies the work of forming, maintaining and maintaining similar profiles of these work packages.

An attempt to find an exact RCPSP solution using heuristics is considered in [10]. This study proposes to use a genetic algorithm to form a schedule that provides maximum profit from the implementation of an IT project. The following assumptions are taken into account [10]:

- capacity constraints can often be systematically changed by temporarily assigning expensive additional production resources or using overtime;
- project revenue decreases with increasing duration of its implementation.

But such a solution to RCPSP is not free from the shortcomings identified for the software for finding the exact solution to RCPSP proposed in [2]. In addition to these shortcomings, the accuracy of the obtained result is affected by the shortcomings caused by the use of the genetic algorithm, of which it is worth noting in particular [11]:

- the tendency to converge to a local optimum instead of searching for a global optimum;
- the difficulties that arise when formulating the stopping condition of the genetic algorithm;
- the poor scalability of genetic algorithms to the complexity of the problem being solved (which is of particular importance for creating schedules for large complex IT projects).

To eliminate the last of the listed shortcomings, in [12] it was proposed to use a genetic algorithm to solve the problem of assigning IT project tasks to members of the development team within one iteration or sprint of the project. At the same time, when solving this problem, the experience of each specific performer is taken into account. As descriptions of IT project tasks, user stories that the development

team must implement in a planned iteration or sprint are considered [12]. Such a solution to the problem of assigning tasks to members of the IT project development team partially eliminates the above-discussed disadvantages of the stopping complexity and poor scalability of genetic algorithms, but remains quite complex to implement and use.

In a later study [13], it was proposed to consider the RCPSP problem for many skills as a task of assigning tasks to employees, taking into account predecessor tasks and constraints for many skills to create an appropriate schedule with the shortest performing time. Differential evolution with multidimensional real-valued functions was proposed to solve this problem. In [13], for this variant of the evolutionary algorithm, its modification was proposed by improving the mutation method, which is performed twice per generation, in order to increase the diversity of the population and obtain better results. Thus, the authors of [13] proposed to eliminate the tendency of genetic and evolutionary algorithms to converge to a local optimum. However, the solution based on the use of evolutionary algorithms is also quite complicated to implement and use. It should be recognized that the task of comparative analysis of the economic and technological efficiency of the operation of various artificial intelligence tools in the conditions of IT project management requires additional research.

In general, for the first half of the 2020s, it was considered appropriate to consider the problem of assigning project tasks to its performers as a multi-criteria optimization problem. An example of this point of view is the conceptual model proposed in [14] for studying the problem of selecting partners for the implementation of research or educational projects. The main components of this model are determining the criteria for selecting potential partners, applying the multi-objective optimization method, expert assessment, forming the preferences of partners or project performers, and making a decision on assigning project tasks to partners that correspond to the corresponding work package [14]. However, the use of expert assessments does not give this conceptual model an advantage over existing methods for solving the problem under consideration for Agile IT projects. The use of experts when solving the problem of assigning project tasks to its performers significantly reduces the objectivity of the resulting solution.

In [15], it is stated that the criteria by which the problem of assigning IT project tasks to its performers should be solved should be:

- qualification of the team of performers;
- competences of the team of performers;
- seriousness of individual tasks;
- priority of individual tasks.

To solve this problem, a mathematical model of labor resource distribution was developed in [15]. This model is based on a probabilistic analysis of the importance and priorities of tasks, ensures more efficient use of personnel and allows for timely completion of the most important tasks of an IT project.

The solution proposed in [15] to the problem of assigning IT project tasks to its performers is quite interesting for automated planning of the activities of the IT project team within one iteration (or sprint). But it is not without some shortcomings, among which it should be noted the impossibility of taking into account the experience of individual performers in solving individual types of IT project tasks. Such experience, in particular, takes into account the performer's knowledge of the subject area and the performer's specialization in the successful implementation of specific types of functions (services, etc.) of the created IT product.

Unlike [15], in [16] it is proposed to use a machine learning-based decision support system that works with data in real time to solve the problem of assigning IT project tasks to its performers. This system analyzes the description of new requested tasks using text mining and machine learning approaches, and then predicts the optimal available personnel that meet the needs of the project task. Personnel qualifications are iteratively updated by the system after each completed task, which provides up-to-date information about personnel capabilities. The basis of this system is the mathematical apparatus of vector descriptions of unstructured texts and structured documents. As a measure of comparison of vector descriptions of new project tasks and performers in the system, cosine similarity is used. The system is built on the basis of microservice architecture, which facilitates its integration into existing IT project planning systems [16]. But the use of vector representations of texts and documents is a rather significant drawback of the proposed solution to the problem of assigning IT project tasks to its performers. Such representations require quite large expenditures of computing resources to create and maintain such descriptions.

The analysis of the considered scientific and applied research in the field of automated solution of RCPSP and its individual tasks for IT projects allowed to draw the following conclusions.

Firstly, none of the considered RCPSP solutions and, in particular, the problem of assigning IT project tasks to its performers is free from shortcomings that significantly limit their practical use in applied IT project planning activities.

Secondly, the considered research results indicate the principle possibility and practical feasibility of solving such problems under the following conditions:

- it is advisable to consider the RCPSP solution as a set of solutions to individual IT project planning problems, in particular - the problem of assigning IT project tasks to its performers;

- it is advisable to consider the problem of assigning IT project tasks to its performers as a multi-criteria optimization problem, the criteria of which can be both the characteristics of individual tasks (seriousness, priority, etc.) and the characteristics of individual IT project performers (competences, specialization, previous work experience, etc.);

- the expected result of solving the problem of assigning IT project tasks to its performers should form a solution to this problem with minimal or no human operator participation (operate in automatic mode);

- the expected result of solving the problem of assigning IT project tasks to its performers should not require significant time and computing resources even when planning large complex projects;

- the expected result of solving the problem of assigning IT project tasks to its performers should be easily integrated into existing application systems for automated planning of IT projects and their iterations.

These conditions can be met if the solution to the problem of assigning IT project tasks to its performers is based on the information that can be obtained with minimal costs from existing IT planning and management of IT projects. Such information, in particular, includes text names and descriptions of project tasks that should be distributed between its performers. Therefore, it will be advisable to consider in the future the task of assigning IT project tasks to its performers as a special case of the text information classification problem. However, unlike the solution proposed in [16], it is necessary to use not vector, but scalar characteristics of project tasks and individual performers to solve such a problem. This makes it possible to use methods that have long been known and proven in various industries to solve the text classification problem.

A comprehensive and detailed analysis of existing text classification methods is given in [17, 18]. In addition, in [18], methods of preprocessing text before classification and indicators of the quality assessment of classification algorithms are described. In [19], the authors use empirical studies to establish a relationship between the size of the training data set and the amount of data required for testing.

The naive Bayesian classifier (NB) [17, 18] has proven itself well in solving such classification problems. This method is based on Bayes' theorem and uses a probabilistic approach to object classification. Confirmation of the NB use can be found in [20, 21], where the performance of different text classifiers is compared in the context of specific classification problems. NB, together with several other classifiers, has demonstrated quite high accuracy.

However, there are certain problems associated with the NB use, as described in [22], and solutions have been proposed to overcome these problems and improve the performance of the algorithms.

Various algorithms have been proposed for the NB implementation, which are used to solve classification problems. These algorithms differ in the requirements for input data and mathematical models that they use to predict object classes [18].

Thus, the aim of this study is to develop a method for solving the problem of assigning IT project tasks to its performers, which must meet the above conditions. Using this method when distributing IT project tasks between members of the team of performers will reduce time costs and increase the objectivity of project iteration (sprint) planning.

To achieve this aim, the following research objectives were set:

- adaptation of polynomial NB to the features of the problem of assigning IT project tasks to its performers;
- development of an algorithm for implementing the developed method;
- experimental verification of the operability of the obtained results.

3.3 Materials and methods

The object of the study is the IT project planning process, in particular, the activity of planning project and technical management [23]. It is within the framework of this activity that the vast majority of IT projects for the creation, modernization or development of IT products should solve the problem of assigning IT project tasks to its performers.

The main hypothesis of this study is the hypothesis of the possibility of automated solution of the problem of assigning IT project tasks to its performers using the methods and models of NB text classification or its varieties.

The study of the features of solving the problem of assigning IT project tasks to its performers is considered in the context of teams working on an IT project using the Agile methodology and, in particular, the Scrum process framework, which provides for an iterative development process. The principles, values, processes, artifacts (Product Backlog, Sprint Backlog, Increment, etc.) and the roles of the Scrum process framework are described in [24, 25]. The practice and problems of prioritizing requirements (and, accordingly, project tasks) in industrial Agile projects, which plays an important role in sprint planning, are highlighted in [26].

As an object of automation in this study, it is proposed to consider an IT company specializing in the release of several products. The term "product" in this study denotes a named set of business opportunities that are valuable for a certain segment of the IT company's customers. Each product of such a company can be the result of the implementation of several projects.

To initiate a new project or implement a certain activity in an existing project, initiatives are created in the company with the participation of product managers and product owners. According to project management standards, the term "initiative" can have several definitions depending on the context. However, in this study, an initiative is an initial action or direction that is chosen to launch a new project or implement a certain activity. An initiative can arise for a variety of reasons, from ideas and strategic goals, to changes in the business environment or consumer needs.

In the process of forming an initiative, high-level tasks of the company are created with the participation of top management, which are agreed upon with the relevant stakeholders of the project. These tasks (another name is topics) are large groups of values for customers, reflected in a set of requirements or user stories related to common functionality, data source or level of security. The created initiative to achieve the results planned by the company involves the further involvement of one or another team or a set of teams of performers. Such involvement occurs with the participation of product managers and product owners. Product Owner is a role in the product development team, which is responsible for managing the product backlog to achieve the desired result that the product development team seeks to achieve [27]. Next, at the project planning stage, the decomposition of the specified initiative into more structured tasks – epics occurs. An Epic is a large, interconnected set of work designed to hierarchically organize a set of requirements and deliver specific business outcomes [6]. Epics divide the initiative's topics into separate functional elements – a set of interconnected functional requirements. Each requirement is described by its own user story. Epics can be created repeatedly in the future, based on the need to create new or additional epics to better perform the initiative. It is important to note that already during the formation of epics, the product owner receives information about the preliminary (possible) assignment of the project team. This information can be based on the results of previous IT projects in which these teams have already been involved in working on similar tasks, or be formed through preliminary conversations and agreements with the managers (leaders) of these teams.

For each of the formed epics, its performing priority is determined. As a result of this determination, the following are formed:

- product roadmap: a publicly available live document that describes the type and direction of product development throughout its life cycle [28];
- product backlog: a prioritized list of user stories for the development team, compiled on the basis of the product roadmap and its requirements [29].

Next, epics are decomposed into individual tasks. This allows to detail the requirements and estimate the scope of work on their implementation in order to

further determine the specific steps of the performers to achieve the desired results. This decomposition occurs at the following events:

- sprint planning (an event in Scrum that launches a sprint);
- Backlog Grooming (a process of regular analysis, refinement, and prioritization of tasks in the product backlog).

The result of this decomposition is the division of epics into separate prioritized user stories and the addition of epics with additional tasks of a technical nature (Tasks). These Tasks may arise from requests to change existing requirements, feedback from IT product users, technical needs that arise during the project, or as a result of an initiative by a team of performers. The results of the decomposition are elements of the product backlog that describe the content of each of the project epics in as much detail as possible. These elements are tasks that must then be distributed among individual performers of one or more teams participating in the implementation of a specific initiative or epic.

A diagram showing an example of possible relationships between initiatives, epics, and tasks of an IT project is shown in Fig. 3.1.

Thus, the solution of the problem of assigning IT project tasks to its performers in this study is considered as starting after the decomposition of initiatives into epics, and epics into individual IT project tasks. These decompositions form the product backlog as a set of epics and individual IT project tasks, which belong either to the corresponding epics or to individual initiatives.

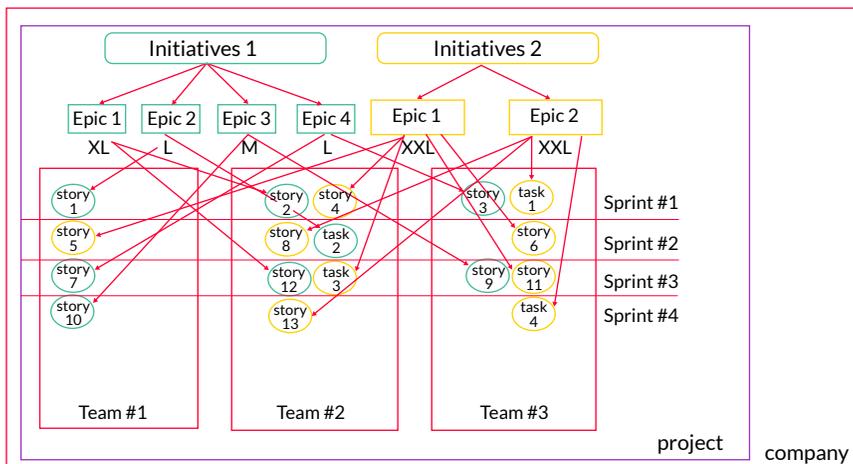


Fig. 3.1 Scheme of relationships between initiatives, epics and tasks of an IT project

To solve the problem of assigning IT project tasks to its performers, the following criteria are proposed to be used:

- the priority of performing that each task has;
- the possible belonging of the task to an epic that is assigned to one or several teams of performers;
- the total efforts for one sprint of each performer (depending on the position), which should not be exceeded when distributing tasks;
- the possible gradation of performers by positions (competences) in the team.

Of all the NB options, the multinomial naive Bayes classifier (MNB) was proposed to be used by the IT project task assignment task assigners. The choice of this classifier was due to the following considerations:

- attributes of IT project task descriptions as classification objects contain discrete features corresponding to a polynomial distribution;
- MNB is easy to implement and allows for fast processing of large amounts of data.

The MNB effectiveness has been confirmed by studies [30, 31]. In particular, [30] describes the results of a comparison of the Bernoulli-Naive Bayes algorithm and MNB and concludes that the polynomial model generally outperforms the Bernoulli model. [31] considers the application of seven different algorithms for the classification and analysis of tweets related to real natural disasters. MNB, in comparison with the Bernoulli-Naive Bayes algorithm and logistic regression, showed the best results in classifying tweets as concise texts with media elements.

Let the object of classification X be described by a set of features (X_1, X_2, \dots, X_n) . Then the posterior probability of belonging of the object X to the class y_k from the set of classes Y should be calculated by the formula

$$P(y_k | X_1, X_2, \dots, X_n) = \frac{P(y_k) \prod_{i=1}^n P(X_i | y_k)}{P(X_1, X_2, \dots, X_n)}, \quad (3.1)$$

where $P(y_k | X_1, X_2, \dots, X_n)$ – the posterior probability of belonging of a classification object with a set of features (X_1, X_2, \dots, X_n) to the class y_k ; $P(y_k)$ – the prior probability of the existence of the class y_k ; $P(X_i | y_k)$ – the probability that a classification object that has a feature X_i with some fixed set of values will be classified as belonging to the class y_k ; $P(X_1, X_2, \dots, X_n)$ – the probability that a classification object X will be described by a set of features (X_1, X_2, \dots, X_n) .

For the case when the prior probabilities $P(y_k)$ for each element y_k of the set of classes Y are known, the classification rule with minimum error [22] is used, which selects the class with the highest posterior probability

$$P(y_k | X_1, X_2, \dots, X_n) \propto \operatorname{argmax}_k P(y_k) \prod_{i=1}^n P(X_i | y_k), \quad (3.2)$$

where \propto – the proportionality sign.

In many cases, the classification rule (3.2) is translated into logarithmic space. Then it takes the form [22]:

$$\log P(y_k | X_1, X_2, \dots, X_n) \propto \operatorname{argmax}_k \log \left(P(y_k) \prod_{i=1}^n P(X_i | y_k) \right), \quad (3.3)$$

$$\log P(y_k | X_1, X_2, \dots, X_n) = \operatorname{argmax}_k \left(\log P(y_k) + \sum_{i=1}^n X_i \log P(X_i | y_k) \right), \quad (3.4)$$

where X_i in (3.4) is the frequency of occurrence of the feature values in the set of studied data (training set), on the basis of which the prior probabilities $P(y_k)$ are calculated.

To estimate the parameters of each class, in [22] it was proposed to use a smoothed version of the maximum likelihood estimation

$$P(X_i | y_k) = \frac{N_{y_k i} + \alpha}{N_{y_k} + \alpha n}, \quad (3.5)$$

where $N_{y_k i}$ – the number of times the feature X_i occurs in the objects of classification X belonging to the class y_k ; N_{y_k} – the total number of times all features X_i occur in the objects of classification X belonging to the class y_k ; n – the number of features describing the objects of classification in the training set; α – the smoothing parameter.

Although the value of the parameter α may be different for each feature, in [22] it is recommended to follow the generally accepted practice, setting $\alpha = 1$ for all features.

The estimate of the probability value $P(y_k)$ is proposed to be calculated as follows

$$P(y_k) = \frac{O_{y_k}}{O}, \quad (3.6)$$

where O_{y_k} – the number of objects of classification present in the training set and belonging to the class y_k ; O – the total number of objects of classification present in the training set.

The UML activity diagram describing the algorithm for solving the classification problem using MNB is shown in **Fig. 3.2**.

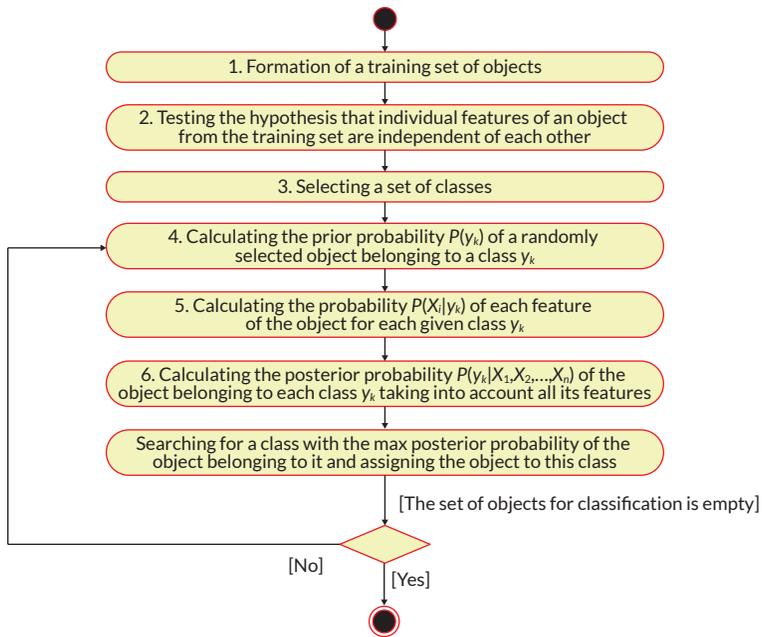


Fig. 3.2 UML activity diagram describing the algorithm for solving the classification problem using a polynomial Bayesian classifier

The use of MNB for solving applied classification problems is associated with the emergence of a number of problems [20]. Among these problems, it is necessary to highlight:

- the problem of reducing the MNB score for a class with a smaller number of training samples in the case when the training set has more training samples for one class than for another;
- the problem of the possible appearance of a class that lacks training sets and, as a result, cannot participate in the classification process;
- the problem of checking the independence of the features of the classification objects in the training set and those that need to be classified from each other.

These problems can be solved in the process of preparing the training set by appropriately selecting the training objects for classification and checking the training set for the independence of the features of these objects.

Another problem arises when the classification object, the class of which should be determined in the process of solving the problem, has a feature that has never

appeared in the training set [20]. This problem can be solved using the smoothing parameter α , $0 < \alpha \leq 1$. The choice of the value of this parameter can significantly affect the results of solving the classification problem.

For solving the problem of assigning IT project tasks to its performers, the description of the algorithm for solving the classification problem using MNB given in **Fig. 3.2** is not entirely suitable due to the following features:

- in this algorithm, the classification process continues until there are no objects left that have not yet been classified;
- the number of objects that can belong to a class is not limited in any way.

In the problem of assigning IT project tasks to its performers, the assignment process must continue until each performer accumulates the necessary total efforts for its position. Therefore, there is a need to modify the method for solving the classification problem using MNB, taking into account this feature.

3.4 Results of developing a method for solving the problem of assigning IT project tasks to its performers

3.4.1 Results of adapting the polynomial naive Bayes classifier to the specifics of the problem of assigning IT project tasks to its performers

The process of adapting MNB to the specifics of the task of assigning IT project tasks to its performers is proposed to be divided into two main stages:

- adaptation of the classification rule with minimal error (3.2) to the specifics of the task of assigning IT project tasks to its performers;
- modification of the algorithm for solving the classification problem using the adapted MNB.

During the adaptation of the classification rule with minimal error (3.2) to the specifics of the task of assigning IT project tasks to its performers, the main elements of this rule were considered and their definitions and calculation rules were clarified. As objects of the MNB classification X as a result of the adaptation, IT project tasks presented in the form of User Story and Task should be considered. The features (X_1, X_2, \dots, X_n) of the classification object X , which are taken into account in rule (3.2), should be the words from which the names of IT project tasks are composed. The names of tasks should reflect the characteristic features of the problem area or the performed part of the IT project. Classes y_k , by which the classification objects X are distributed, should describe individual performers who are part of one

or more teams of IT project performers. The training set should contain data on already completed tasks (in particular, their names) and performers of these tasks.

Then the classification rule with minimal error (3.2) should be formulated as a rule for selecting the performer e_k with maximum posterior probability

$$e_k \propto \operatorname{argmax}_k P(e_k) \prod_{i=1}^n P(\omega_i | e_k), \quad (3.7)$$

where $P(e_k)$ – the prior probability that the IT project task belongs to the performer e_k ; $P(\omega_i | e_k)$ – the probability that the word ω_i from the IT project task title will be classified as being distributed to the performer e_k ; n – the number of words in the IT project task title.

The prior $P(e_k)$ should be calculated by the formula

$$P(y_k) = \frac{T_{e_k}}{T}, \quad (3.8)$$

where T_{e_k} – the number of tasks in the training set of tasks performed by the performer e_k ; T – the total number of tasks in the training set.

The probability $P(\omega_i | e_k)$ should be calculated using formula (3.3)

$$P(\omega_i | e_k) = \frac{N_{e_k, \omega_i} + \alpha}{N_{e_k} + \alpha V}, \quad (3.9)$$

where N_{e_k, ω_i} – the number of occurrences of the word ω_i in the names of the tasks of the training set performed by the performer e_k ; N_{e_k} – the total number of words in the names of the tasks of the training set performed by the performer e_k ; V – the size of the vocabulary (the total number of words in the training set); α – the smoothing parameter (allows to solve the problem of the appearance in the names of tasks of such words that have never been encountered in the training set). In our case, it is possible to assume that $\alpha = 1$.

But the considered results of adapting the classification rule (3.2) to the specifics of the task of assigning IT project tasks to its performers do not take into account the specifics of the existence of individual performers within the IT project performed by the company. In particular, the adapted classification rule (3.7) does not take into account:

- the condition of no excess of the total efforts of each performer e_k for one sprint;
- the condition of the performer e_k being in the team in a specific position during the planning and performing of the sprint, which can be determined by the competencies of this performer.

To take these conditions into account, an assumption is introduced according to which each individual performer e_k can be characterized by the parameter "Maximum effort of the performer in one sprint". This parameter can be formally represented as the indicator "Maximum allowable weight of class e_k " $V_{\max k}$. The value of the indicator $V_{\max k}$ is measured in story points (Story Points) – conventional units of measurement for expressing the assessment of the total effort required for the full implementation of a product backlog element or any other part of the work [32]. For each performer e_k , the value of the indicator $V_{\max k}$ is determined based on the position held by the performer e_k in the team according to their own competencies. Similarly, to describe each individual task of an IT project, the parameter "Task completion effort" has been introduced, which can be formally represented by the indicator "Task weight" v_i . The value of the indicator v_i is also measured in Story Points.

Then the classification rule (3.7) when assigning each j -th task of an IT project will work only if the condition is met

$$S_{vk} = \sum_{i=1}^j v_i \leq V_{\max k}, \quad (3.10)$$

where S_{vk} – the total weight of the class e_k accumulated as a result of the successful assignment of the previous $j - 1$ tasks and the current j -th task of the IT project.

If this condition is violated, then the posterior probability of the IT project task belonging to the performer e_k should be considered equal to zero. The search for the performer e_k with the maximum posterior probability of the task belonging to it in this case should be continued.

The introduction of condition (3.10) led to the emergence of a new variant of solving the classification problem. In the derived variant, this problem was solved until all available classification objects were classified. In our case, an additional variant of solving the classification problem should be considered the variant in which the value of the total weight S_{vk} of each of the classes approached the value of the indicator $V_{\max k}$ as much as possible. This condition means that each performer received the maximum possible number of IT project tasks to perform, which it can perform during the planned sprint.

Based on the considered results of adapting MNB to the specifics of the task of assigning IT project tasks to its performers, the algorithm for solving the classification problem using MNB was modified and shown in **Fig. 3.2**. The UML activity diagram describing the modified algorithm for solving the classification problem using the adapted MNB is shown in **Fig. 3.3**.

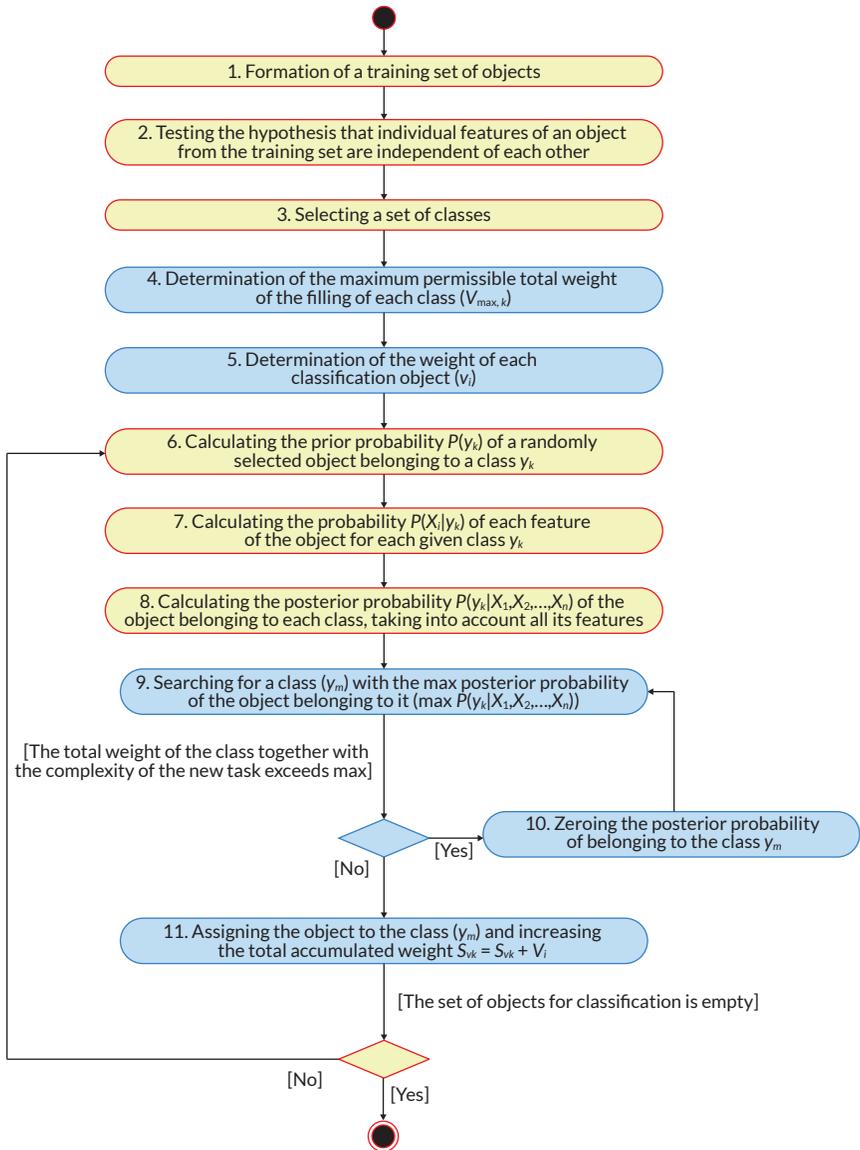


Fig. 3.3 UML activity diagram describing a modified algorithm for solving a classification problem using an adapted polynomial Bayesian classifier

3.4.2 Description of the results of developing an algorithm for implementing the developed method

The use of the modified algorithm allowed to develop a method for solving the problem of assigning IT project tasks to its performers as a sequence of such stages:

Stage 1 is the preparatory stage.

Stage 2 is the formation of the "Product Backlog" artifact as a set of IT project tasks.

Stage 3 is the formation of detailed descriptions of IT project tasks, the descriptions of which are included in the "Product Backlog" artifact.

Stage 4 is the assignment of IT project tasks from the "Product Backlog" artifact to performers. Formation of the "Sprint Backlog" artifact. Completion of the method.

But this representation of the developed method is too generalized. Therefore, for the successful implementation of this method, an appropriate algorithm was developed. The UML activity diagram, which describes the scheme of this algorithm, is shown in **Fig. 3.4**.

The use of the developed algorithm allowed to describe in detail the features of the implementation of each of the stages of the method as a sequence of steps of this algorithm. Let's consider these sequences of steps in more detail.

The implementation of Stage 1 of the developed method consists of a sequence of the following algorithm steps:

Step 1. Formation of the training set.

The implementation of Stage 2 of the developed method consists of a sequence of the following algorithm steps.

Step 2. Dividing the initiatives of the current IT project into epics.

Step 3. Distribution of epics between teams.

Step 4. Dividing epics into User Stories and Tasks.

Step 5. Adding Tasks to the task pool outside the epics.

The implementation of Stage 3 of the developed method consists of a sequence of the following algorithm steps.

Step 6. Determining the values of the attributes of User Stories and Tasks.

Step 7. Determining the maximum efforts of each performer.

Step 8. Sorting tasks by priority.

The implementation of Stage 4 of the developed method consists of a sequence of the following algorithm steps.

Step 9. If the "Product Backlog" artifact does not contain unassigned tasks, go to Step 17. Otherwise, select the next task from the "Product Backlog" artifact. If this task has the attribute of belonging to an epic, go to Step 10. Otherwise, go to Step 11.

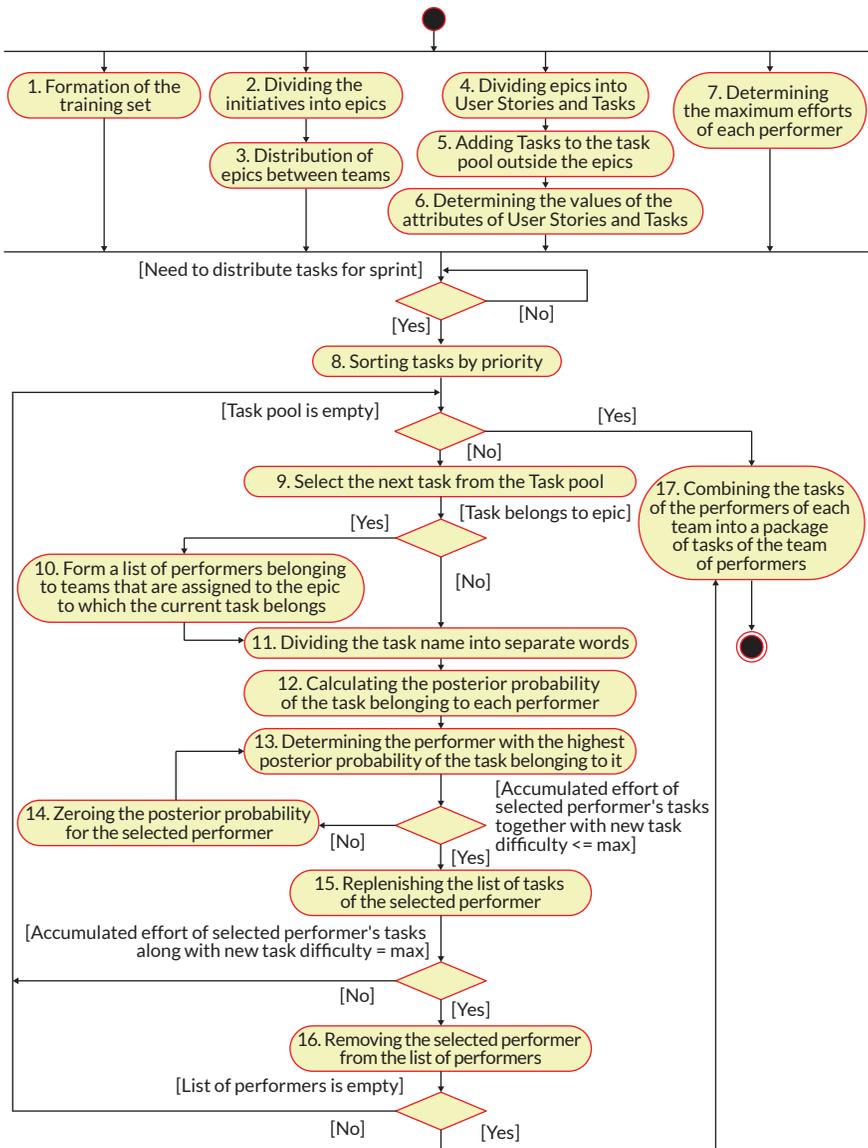


Fig. 3.4 UML activity diagram describing the algorithm for solving the problem of assigning IT project tasks to its performers

Step 10. Form a list of performers belonging to teams that are assigned to the epic to which the current task belongs.

Step 11. Dividing the task name into separate words.

Step 12. Calculating the posterior probability of the task belonging to each performer.

Step 13. Determining the performer with the highest posterior probability of the task belonging to it. Check the accumulated total weight of the performer S_{vk} . If this total weight does not exceed the value V_{maxk} , go to Step 15.

Step 14. Zeroing the posterior probability for the selected performer. Go to Step 13.

Step 15. Replenishing the list of tasks of the selected performer. Check the accumulated total weight of the performer S_{vk} . If this total weight is still less than V_{maxk} , go to Step 9.

Step 16. Removing the selected performer from the list of performers. If the list of performers of the planned sprint is not yet empty, go to Step 9. Otherwise, end the method.

Step 17. Combining the tasks of the performers of each team into a package of tasks of the team of performers. Form the artifact "Sprint Backlog".

End the method.

In Step 1 of the developed algorithm, data collection is first performed. It involves the accumulation of data on past tasks of those IT projects that are being performed, as well as on previously completed IT projects of the company. This data should include data on the performers to whom the tasks were assigned, and on the results of the work of these performers. This data must be prepared for further analysis and training of the adapted MNB before the first application of the developed method. When performing data collection, it is important to check the condition under which the number of tasks performed by different performers should be approximately the same. In the event that the use of the developed method is recognized as successful, this data can be updated after the completion of each individual IT project performed by the IT company.

After collecting historical data, it is necessary to test the hypothesis of the independence of individual words in the names of tasks. For this, it is proposed to use pairwise correlation analysis of each combination of two words of the name. Such an analysis can be done, for example, by counting how many times each pair of words occurs in the collected task names and comparing this number with the total number of word pairs in the collected task names. If the correlation percentage is lower than a priori set threshold, the correlation between these words can be considered absent.

For preprocessing of the collected data, it is necessary to remove stop words [18]. Stop words are words that do not add meaning to the phrase on their own, such as prepositions, articles, conjunctions, pronouns, etc.

After preprocessing, a training set is formed from the collected data. The size of the training set affects the accuracy of classification. To achieve the best results, the volume of test data should be 20–30%, and the remaining 70–80% should be training data [19]. However, it is advisable to determine the optimal size of the training set for a specific IT company and IT projects experimentally after obtaining several results using the developed method and algorithm. If among the performers of the IT project there is a new employee who has not yet worked in an IT company, then when forming the training set for this performer, it is advisable to add the names of tasks that correspond to its qualifications and specialization. These names can be obtained, for example, during the interview with such an employee or as a result of analyzing its resume.

The results of Step 1 of the algorithm are a repository of historical data and a training set prepared for preliminary training of the adapted MNB.

At Step 2, the following data must be collected:

- designation and/or name of the IT project that is planned for implementation;
- description of the set of initiatives of this IT project and each individual initiative from this set;
- description of the results of the decomposition of the IT project initiative into separate epics;
- description of each of the epics that are the result of this decomposition.

The division of initiatives into epics is usually carried out by project managers with the participation of product owners and the scrum master, who are responsible for planning and managing the IT project.

When forming epics, the project manager is guided by the principle of logical coherence, the size and complexity of the epic, as well as the relationships between the tasks of a single epic. Logical coherence aims to combine tasks that belong to the same functional area or have similar goals. This helps to avoid dispersion of efforts and ensures a more coherent implementation of the project. In terms of size and complexity, epics should be large enough to make their separation meaningful, but not so large that they become unmanageable.

The result of Step 2 of the algorithm is a data array that allows to form an information artifact "Product Backlog" as a description of the IT project initiative that is planned and/or being performed, and the epics to which this initiative is distributed.

In Step 3, it is necessary to collect the following data:

- a description of the epics planned for current performing;
- descriptions of the teams of potential IT project performers.

The distribution of epics is usually carried out by the product owner, with the participation of the scrum master, and in some cases (depending on the specifics of the IT project) and with the involvement of the teams of potential performers.

Epics are distributed in such a way that they correspond to the specialization and competencies of the team. When distributing epics, it is also necessary to take into account the dependencies between tasks in order to avoid situations where one team cannot complete its task due to dependence on the task of another team. An epic can be distributed to several teams at the same time if it contains tasks that require different competencies. In this case, the epic tasks are divided into separate subtasks, which are then distributed between the appropriate teams.

The result of Step 3 of the algorithm is a data array that describes the decision to involve individual teams of IT project performers in the performing of epics planned for current performing. The basis of this data array is the information artifact "Product Backlog", supplemented by descriptions of the teams of performers assigned to perform each epic and the individual performers that make up these teams.

At Step 4, it is necessary to collect the following data:

- a description of the epics planned for current performing;
- a set of descriptions of individual tasks that make up each planned epic;
- a description of the results of the decomposition of the IT project initiative into individual epics;
- an information artifact "Product Backlog" as a description of the set of IT project tasks planned for performing.

The breakdown of epics into individual IT project tasks (User Stories and Tasks) and a preliminary assessment of the efforts of these tasks are usually carried out by product owners with the participation of the scrum master, in close cooperation with the teams of performers. This step of the algorithm occurs during the planning of individual sprints or before the start of a new project phase.

The breakdown of epics into User Stories and Tasks is carried out according to the principle of logical sequence, effort and complexity. Epics are decomposed into individual tasks that are logically interconnected and can be performed independently of each other. This facilitates task management and their subsequent integration. When decomposing epics into individual tasks, it is necessary to constantly check whether the efforts of these tasks correspond to the planned maximum sprint effort. This allows to more accurately estimate the time and resources required to complete each task. The frequency of breaking epics into individual tasks of an IT project depends on the pace and specifics of this project. This breakdown can occur at regular planning meetings (for example, at the beginning of each sprint) or in cases where new epics appear that require distribution. This approach to breaking epics provides detailed work planning, facilitates the assessment of the efforts of individual tasks and allows to effectively manage the project.

The result of Step 4 of the algorithm is a data array that allows to form a "Product Backlog" information artifact as a detailed description of the IT project initiative, which contains:

- a description of the current IT project initiative that is planned and/or being performed;
- a description of the epics planned for current performing, to which this initiative is distributed;
- description of individual IT project tasks (User Stories and Tasks), into which epics planned for current performing are decomposed;
- descriptions of IT project teams involved in the performing of epics planned for current performing.

At Step 5, the following data must be collected:

- description of epics planned for current performing;
- a set of descriptions of individual tasks that make up each planned epic;
- a description of the results of the decomposition of the IT project initiative into individual epics;
- a set of descriptions of individual tasks that are not part of any epic;
- the "Product Backlog" information artifact as a description of the set of IT project tasks planned for performing.

Adding tasks that are not part of any epic to the "Product Backlog" artifact is usually done by the product owner, scrum masters or other responsible persons (the establishment of these persons depends on the team structure and IT project) mostly after discussing and agreeing on these tasks at the team level. Such tasks may appear, for example, as a result of detecting errors in already completed IT projects or if it is necessary to make changes to them at the request of the customer. This step is performed at planning meetings or planning of individual sprints in cases where new tasks appear that do not belong to existing epics.

Adding tasks that are not part of any epic allows not to lose important, but less global tasks that still need to be completed. Adding such tasks to the "Product Backlog" artifact ensures their inclusion in the general distribution and performing within the project.

The result of Step 5 of the algorithm is a data array that allows to specify the content of the "Product Backlog" information artifact as a detailed description of the IT project initiative, which includes:

- a description of the current IT project initiative that is planned and/or being implemented;
- a description of the epics planned for current performing, into which this initiative is distributed;

- a description of individual IT project tasks (User Stories and Tasks), into which the epics planned for current performing are decomposed;
- a description of individual IT project tasks (usually Tasks) that require current performing, but are not included in any of the epics planned for current performing;
- descriptions of the IT project teams involved in the performing of the epics planned for current performing.

At Step 6, the following data must be collected:

- a description of individual IT project tasks (User Stories and Tasks) planned for current performing;
- a set of tuples of attribute values that characterize each IT project task scheduled for current performing.

The determination of task attribute values can begin if there is a need for this, as well as on the eve of sprint planning. This step of the algorithm is performed by the product owner and scrum masters in interaction with the teams of performers assigned to perform the epics that include these tasks.

The attributes that characterize IT project tasks are the name, priority, evaluation of the performing effort, and the epic to which the task belongs. The values of these attributes significantly affect the performing of the modified classification task algorithm using the adapted MNB. Task priorities can be set based on their impact on the IT project, performing deadlines, and importance to the customer. These priorities are represented by numbers. The smaller the number, the higher the priority of the task.

Task effort is assessed in Story Points. Effort can range from minimum (1 Story Point) to maximum (13 Story Points). Typically, the following Story Points values are used for evaluation: 1, 3, 5, 8, and 13 [33].

The result of Step 6 of the algorithm is a data array that allows to specify the content of the "Product Backlog" information artifact as a detailed description of the IT project initiative, which includes:

- a description of individual IT project tasks (User Stories and Tasks) planned for current performing;
- a list of attribute values of each IT project task planned for current performing;
- descriptions of the teams of IT project performers involved in the performing of epics planned for current performing.

At Step 7, the following data must be collected for each performer:

- designation of the team to which this performer belongs;
- the level of the performer's competencies and the position that it occupies in the team according to these competencies;
- the standard of effort (in story points) for the planned sprint for the performer in the selected position.

This data can be collected by the scrum master for each performer of a subordinate team immediately before the start of a new iteration in the form of a sprint.

When setting this data, it is highly desirable to take into account such input data as the total number of sprint days, the number of performers, the number of holidays, the number of Story Points that were worked on during previous sprints. This additional data helps to plan the desired number of Story Points for each performer and thereby further plan the load of teams with tasks from the "Product Backlog" artifact in such a way as to maximally complete the tasks that have been taken into work.

The result of Step 7 of the algorithm is an array of normative data that allows to set the value of the V_{\max} indicator for each performer of the IT project that is planned and performed.

At Step 8, it is necessary to collect the following data:

- a description of individual IT project tasks (User Stories and Tasks) planned for current performing;

- a list of priority values of each IT project task planned for current performing.

Typically, the sorting of IT project tasks by priority is carried out by the product owner, placing tasks in the "Product Backlog" artifact by their priorities from the highest (minimum number) to the lowest (maximum number) value of the "Task Priority" attribute.

Sorting tasks in the "Product Backlog" artifact by priorities allows to arrange these tasks in such a way that the most important of them are planned for performing first. This helps to optimize resource costs and increase the efficiency of IT project performing.

The result of performing Step 8 of the algorithm is a data array that contains the "Product Backlog" information artifact, ordered by task priority values, as a list of individual IT project tasks scheduled for current performing.

In fact, Steps 9–17 of the algorithm for implementing the method for solving the problem of assigning IT project tasks to their performers repeat the steps of the modified algorithm for solving the classification problem using the adapted MNB shown in **Fig. 3.3**. Therefore, these features of the performing of these steps are not considered in detail here.

3.5 Experimental verification of the improved method

It was decided to conduct an experimental verification of the obtained results in one of the outsourcing IT companies of Ukraine. The selected IT company is characterized by a significant number of implemented IT projects and a variety of their

tasks (from software development to cybersecurity). The company has multidisciplinary teams consisting of specialists from different areas: developers (Frontend, Backend), DevOps engineers, QA engineers, BI engineers, etc. These teams can be either permanent or formed for the duration of specific projects. The company also has an accumulated database of previous IT projects, tasks and results of their implementation, which is important for training and tuning the modified algorithm for solving the classification problem.

For experimental verification of the obtained results, two teams of performers (symbols "A" and "B") were selected in the selected IT company. Team A consists of two people, team B consists of three people. The competencies of these individuals are characterized by one of three possible levels of gradation: junior, middle and senior. A description of these teams is given in **Table 3.1**.

Table 3.1 Characteristics of performers

Team	Performer	Gradation level	Max effort (SP)
A	Ivan Petrov	Junior	20
	Oleksiy Kovalek	Middle	35
B	Maria Ivanova	Senior	50
	Gana Sergienko	Middle	35
	Dmytro Orlov	Junior	20

Table 3.1 shows the level of gradation and maximum effort standard, measured in Story Points, for each performer.

During Stage 1 of the developed method, a training set was formed for training the adapted MNB based on historical data on successfully completed tasks.

Information on successfully completed tasks, on the basis of which the training set was built, is given in **Table 3.2**.

From **Table 3.2** it can be seen that the total number of tasks in the training set $T = 20$.

The names of the completed IT project tasks given in **Table 3.2** were tokenized – divided into separate words (tokens) with the exception of conjunctions and prepositions. The number of occurrences of individual tokens in the names of the tasks of each of the performers is given in **Table 3.3**.

As a result of the implementation of Stage 2 and Stage 3 of the developed method, the "Product Backlog" information artifact was formed as a detailed list of IT project tasks, ordered by priority values, from epics, for the implementation of which teams A and B were assigned. The description of this artifact is given in **Table 3.4**.

Table 3.2 List of completed tasks and their performers

No.	Task	Task performer (graduation level)
1	Frontend Design Update	Ivan Petrov (Junior)
2	UX/UI Improvement	Ivan Petrov (Junior)
3	Frontend Framework Update	Ivan Petrov (Junior)
4	Backend API Integration	Oleksiy Kovalek (Middle)
5	Web Page Load Optimization	Oleksiy Kovalek (Middle)
6	Business Intelligence Report Creation	Oleksiy Kovalek (Middle)
7	Backend Data Processing Optimization	Oleksiy Kovalek (Middle)
8	Real-time Data Analytics Dashboard	Oleksiy Kovalek (Middle)
9	Database Performance Tuning	Maria Ivanova (Senior)
10	User Authentication Implementation	Maria Ivanova (Senior)
11	Data Encryption Setup	Maria Ivanova (Senior)
12	Data Migration from Legacy Systems	Maria Ivanova (Senior)
13	DevOps Pipeline Automation	Maria Ivanova (Senior)
14	Security Vulnerability Assessment	Gana Sergienko (Middle)
15	Cloud Resource Allocation	Gana Sergienko (Middle)
16	Continuous Deployment Setup	Gana Sergienko (Middle)
17	Network Configuration Management	Gana Sergienko (Middle)
18	Mobile App Bug Fixing	Dmytro Orlov (Junior)
19	Automated Testing Script Development	Dmytro Orlov (Junior)
20	System Backup Configuration	Dmytro Orlov (Junior)

Table 3.3 Number of occurrences of tokens in the names of the tasks of each of the performers

No.	Word (token)	Ivan Petrov (Junior)	Oleksiy Kovalek (Middle)	Maria Ivanova (Senior)	Gana Sergienko (Middle)	Dmytro Orlov (Junior)
1	2	3	4	5	6	7
1	Allocation	0	0	0	1	0
2	Analytics	0	1	0	0	0
3	API	0	1	0	0	0
4	App	0	0	0	0	1
5	Assessment	0	0	0	1	0
6	Authentication	0	0	1	0	0
7	Automated	0	0	0	0	1
8	Automation	0	0	1	0	0
9	Backend	0	2	0	0	0

Continuation of Table 3.3

1	2	3	4	5	6	7
10	Backup	0	0	0	0	1
11	Bug	0	0	0	0	1
12	Business	0	1	0	0	0
13	Cloud	0	0	0	1	0
14	Configuration	0	0	0	1	1
15	Continuous	0	0	0	1	0
16	Creation	0	1	0	0	0
17	Dashboard	0	1	0	0	0
18	Data	0	2	2	0	0
19	Database	0	0	1	0	0
20	Deployment	0	0	0	1	0
21	Design	1	0	0	0	0
22	Development	0	0	0	0	1
23	DevOps	0	0	1	0	0
24	Encryption	0	0	1	0	0
25	Fixing	0	0	0	0	1
26	Framework	1	0	0	0	0
27	Frontend	2	0	0	0	0
28	Implementation	0	0	1	0	0
29	Improvement	1	0	0	0	0
30	Integration	0	1	0	0	0
31	Intelligence	0	1	0	0	0
32	Legacy	0	0	1	0	0
33	Load	0	1	0	0	0
34	Management	0	0	0	1	0
35	Migration	0	0	1	0	0
36	Mobile	0	0	0	0	1
37	Network	0	0	0	1	0
38	Optimization	0	2	0	0	0
39	Page	0	1	0	0	0
40	Performance	0	0	1	0	0
41	Pipeline	0	0	1	0	0
42	Processing	0	1	0	0	0
43	Real-time	0	1	0	0	0
44	Report	0	1	0	0	0

Continuation of Table 3.3

1	2	3	4	5	6	7
45	Resource	0	0	0	1	0
46	Script	0	0	0	0	1
47	Security	0	0	0	1	0
48	Setup	0	0	1	1	0
49	System	0	0	0	0	1
50	Systems	0	0	1	0	0
51	Testing	0	0	0	0	1
52	Tuning	0	0	1	0	0
53	Update	2	0	0	0	0
54	User	0	0	1	0	0
55	UX/UI	1	0	0	0	0
56	Vulnerability	0	0	0	1	0
57	Web	0	1	0	0	0

Table 3.4 List of IT project tasks, ordered by priority values, planned for current performing

No.	Task name	Priority	Effort (SP)	Epic ID
1	Backend API performance tuning	1	8	E2
2	Mobile app security audit	1	8	E3
3	Web page speed optimization	1	8	E5
4	Cloud infrastructure setup	1	13	E6
5	Network configuration automation	1	8	E9
6	Business intelligence dashboard creation	2	13	E8
7	Backend service integration	2	8	E2
8	Continuous integration pipeline setup	2	5	E7
9	System backup strategy implementation	2	5	E10
10	Automated testing suite development	2	8	-
11	Frontend Feature Extension	2	8	E1
12	Frontend design enhancement	2	5	E1
13	Database schema optimization	2	5	E4
14	Real-time data analysis implementation	2	5	-
15	Mobile app testing	2	5	E3
16	Data encryption security setup	3	3	E11
17	User interface design update	3	3	E1
18	Cloud service migration	3	13	E6
19	Machine learning model training	3	13	E12

From **Table 3.4** it can be seen that the total number of IT project tasks planned for the current performing is 19, of which two tasks (No. 10 and No. 14) are tasks that do not belong to any of the epics and are assigned additionally.

The progress of Stage 4 of the developed method (the actual solution of the problem of assigning IT project tasks to their performers) was proposed to be considered using the example of its iteration for the task "Frontend design enhancement" and Ivan Petrov (Junior) as its possible performer. Before the start of this iteration, the names of the IT project tasks given in **Table 3.4** were also tokenized.

The parameters for calculating the value of the classification rule (3.7), which establishes the posterior probability of the performance of the task "Frontend design enhancement" by the performer Ivan Petrov, are given in **Table 3.5**.

Table 3.5 Parameters for calculating the value of the classification rule (3.7)

Parameter	Value
General parameters	
T_{e_k}	3
T	20
N_{e_k}	8
α	1
V	57
"Frontend" token	
$N_{e_k w_j}$	2
"Design" token	
$N_{e_k w_j}$	1
"Enhancement" token	
$N_{e_k w_j}$	0 (missing from the training set)

Then the value of the prior probability P_{e_k} according to formula (3.8) is

$$P(\text{"Ivan Petrov"}) = \frac{3}{20} = 0.15.$$

The value of the probability $P(\text{"Frontend"} | \text{"Ivan Petrov"})$ according to formula (3.9) is

$$P(\text{"Frontend"} | \text{"Ivan Petrov"}) = \frac{2+1}{8+1 \times 57} = \frac{3}{65} = 0.046154.$$

The value of the probability $P(\text{"Design"}|\text{"Ivan Petrov"})$ according to formula (3.9) is

$$P(\text{"Design"}|\text{"Ivan Petrov"}) = \frac{1+1}{8+1 \times 57} = \frac{2}{65} = 0.03077.$$

The value of the probability $P(\text{"Enhancement"}|\text{"Ivan Petrov"})$ according to formula (3.9) is

$$P(\text{"Enhancement"}|\text{"Ivan Petrov"}) = \frac{0+1}{8+1 \times 51} = \frac{3}{65} = 0.015385.$$

The value of the posterior probability according to formula (3.7) is

$$P(e_k) \prod_{i=1}^n P(\omega_i | e_k) = 0.15 \times 0.046154 \times 0.03077 \times 0.015385 = \\ = 0.000003277370962995 \approx 3.3 \times 10^{-6}.$$

The results of the calculations of the posterior probability for each task and performer are given in **Table 3.6**.

Table 3.6 Results of the calculations of the posterior probability for each task and performer

Task No.	Ivan Petrov (Junior)	Oleksiy Kovalek (Middle)	Maria Ivanova (Senior)	Gana Sergienko (Middle)	Dmytro Orlov (Junior)
1	2	3	4	5	6
1	8.4×10^{-9}	3.48×10^{-8}	3.52×10^{-8}	7.44×10^{-9}	6.62×10^{-9}
2	8.4×10^{-9}	5.81×10^{-9}	8.8×10^{-9}	1.49×10^{-8}	2.65×10^{-8}
3	8.4×10^{-9}	6.97×10^{-8}	8.8×10^{-9}	7.44×10^{-9}	6.62×10^{-9}
4	5.46×10^{-7}	4.7×10^{-7}	1.29×10^{-6}	2.14×10^{-6}	4.57×10^{-7}
5	5.46×10^{-7}	4.7×10^{-7}	1.29×10^{-6}	2.14×10^{-6}	9.13×10^{-7}
6	8.4×10^{-9}	9.29×10^{-8}	8.8×10^{-9}	7.44×10^{-9}	6.62×10^{-9}
7	5.46×10^{-7}	2.82×10^{-6}	6.43×10^{-7}	5.36×10^{-7}	4.57×10^{-7}
8	8.4×10^{-9}	1.16×10^{-8}	3.52×10^{-8}	2.98×10^{-8}	6.62×10^{-9}
9	8.4×10^{-9}	5.81×10^{-9}	1.76×10^{-8}	7.44×10^{-9}	2.65×10^{-8}
10	8.4×10^{-9}	5.81×10^{-9}	8.8×10^{-9}	7.44×10^{-9}	5.29×10^{-8}
11	1.64×10^{-6}	4.7×10^{-7}	6.43×10^{-7}	5.36×10^{-7}	4.57×10^{-7}
12	3.3×10^{-6}	4.7×10^{-7}	6.43×10^{-7}	5.36×10^{-7}	4.57×10^{-7}
13	5.46×10^{-7}	1.41×10^{-6}	1.29×10^{-6}	5.36×10^{-7}	4.57×10^{-7}

Continuation of Table 3.6

1	2	3	4	5	6
14	8.4×10^{-9}	3.48×10^{-8}	5.28×10^{-8}	7.44×10^{-9}	6.62×10^{-9}
15	5.46×10^{-7}	4.7×10^{-7}	6.43×10^{-7}	5.36×10^{-7}	3.65×10^{-6}
16	8.4×10^{-9}	1.74×10^{-8}	1.06×10^{-7}	2.98×10^{-8}	6.62×10^{-9}
17	5.04×10^{-8}	5.81×10^{-9}	1.76×10^{-8}	7.44×10^{-9}	6.62×10^{-9}
18	5.46×10^{-7}	4.7×10^{-7}	1.29×10^{-6}	1.07×10^{-6}	4.57×10^{-7}
19	8.4×10^{-9}	5.81×10^{-9}	8.8×10^{-9}	7.44×10^{-9}	6.62×10^{-9}

The results of applying logarithmic transformation (based on the decimal logarithm) for better perception of the classification results according to formula (3.3) are given in **Table 3.7**.

Table 3.7 Results of applying logarithmic transformation of posterior probability calculations for each task and performer

Task No.	Ivan Petrov (Junior)	Oleksiy Kovalek (Middle)	Maria Ivanova (Senior)	Gana Sergienko (Middle)	Dmytro Orlov (Junior)
1	-8.076	-7.46	-7.45	-8.13	-8.18
2	-8.076	-8.24	-8.06	-7.83	-7.58
3	-8.076	-7.16	-8.06	-8.13	-8.18
4	-6.26	-6.33	-5.89	-5.67	-6.34
5	-6.26	-6.33	-5.89	-5.67	-6.04
6	-8.076	-7.03	-8.06	-8.13	-8.18
7	-6.26	-5.55	-6.19	-6.27	-6.34
8	-8.076	-7.94	-7.45	-7.53	-8.18
9	-8.076	-8.24	-7.75	-8.13	-7.58
10	-8.076	-8.24	-8.06	-8.13	-7.28
11	-5.79	-6.33	-6.19	-6.27	-6.34
12	-5.48	-6.33	-6.19	-6.27	-6.34
13	-6.26	-5.85	-5.89	-6.27	-6.34
14	-8.076	-7.46	-7.28	-8.13	-8.18
15	-6.26	-6.33	-6.19	-6.27	-5.44
16	-8.076	-7.76	-6.98	-7.53	-8.18
17	-7.3	-8.24	-7.75	-8.13	-8.18
18	-6.26	-6.33	-5.89	-5.97	-6.34
19	-8.076	-8.24	-8.06	-8.13	-8.18

The results of the IT project task distribution for the performers from teams A and B are presented in **Table 3.8**.

They show that none of the performers reached their maximum workload. This is probably due to the lack of tasks with a complexity that would exactly match the remaining maximum workload capabilities of the performers.

Table 3.8 Task distribution results

User Stories & Task	The complexity (SP)	Performer	Max effort (SP)	Remain-der SP
11. Frontend Feature Extension	8	Ivan Petrov (Junior)	20	4
12. Frontend design enhancement	5			
17. User interface design update	3			
3. Web page speed optimization	8	Oleksiy Kovalek (Middle)	35	1
6. Business intelligence dashboard creation	13			
7. Backend service integration	8			
13. Database schema optimization	5	Maria Ivanova (Senior)	50	8
1. Backend API performance tuning	8			
8. Continuous integration pipeline setup	5			
10. Automated testing suite development	5			
14. Real-time data analysis implementation	5			
16. Data encryption security setup	3	Gana Sergienko (Middle)	35	1
18. Cloud service migration	13			
4. Cloud infrastructure setup	13			
5. Network configuration automation	8	Dmytro Orlov (Junior)	20	2
19. Machine learning model training	13			
2. Mobile app security audit	8			
9. System backup strategy implementation	5	Dmytro Orlov (Junior)	20	2
15. Mobile app testing	5			

In the case of assignment of task No. 10, the total weight of assigned tasks for the performer Dmytro Orlov exceeded his maximum allowable weight. Therefore, it was decided to transfer this task to the next performer with the highest posterior probability. This performer turned out to be Maria Ivanova.

In the case of assignment of task No. 19, the total weight of assigned tasks for the performer Maria Ivanova exceeded her maximum allowable weight. Therefore, it was decided to transfer this task to the next performer with the highest posterior

probability. This performer turned out to be Ivan Petrov. But in the case of assignment of task No. 19 to him, the total weight of assigned tasks also exceeded his maximum allowable weight. Therefore, it was decided to transfer task No. 19 to the next performer. This performer turned out to be Gana Sergienko.

After distributing tasks among the performers of the IT project, task packages were formed for each team of performers by combining tasks that were distributed to performers of the same team. The formed task packages of each team are given in **Table 3.9**.

Table 3.9 Results of using the developed method

Team	Task
A	3. Web page speed optimization
	6. Business intelligence dashboard creation
	7. Backend service integration
	11. Frontend Feature Extension
	12. Frontend design enhancement
	13. Database schema optimization
	17. User interface design update
B	1. Backend API performance tuning
	2. Mobile app security audit
	4. Cloud infrastructure setup
	5. Network configuration automation
	8. Continuous integration pipeline setup
	9. System backup strategy implementation
	10. Automated testing suite development
	14. Real-time data analysis implementation
	15. Mobile app testing
	16. Data encryption security setup
	18. Cloud service migration
	19. Machine learning model training

3.6 Discussion of the research results

As a result of the study, a method was developed for automatically solving the problem of assigning IT project tasks to its performers. The developed method is based on the presentation of the problem of assigning IT project tasks to its

performers as a type of classification problem. This presentation made it possible to propose using MNB to solve this problem. This version of the classifier is quite simple to implement and allows for good processing of sets of attribute data, which are descriptions of IT project tasks.

During the study, the existing version of MNB was adapted to the features of the problem of assigning IT project tasks to its performers. In particular, the classification rule with minimal error (3.2) as a result of adaptation to the features of this problem took the form (3.7). The results of adapting the methods for calculating the values of the elements of the classification rule (3.7) are given in the form of expressions (3.8) and (3.9). To take into account the limitations that arise as a result of the IT project performer being in the team in a specific position, it was proposed to add condition (3.10) to the classification rule (3.7) of the possibility of assigning a task to a specific performer.

The results of adapting MNB to the specifics of the task of assigning IT project tasks to its performers were used as the basis for modifying the algorithm for solving the classification problem. The result of this modification in the form of an activity diagram in the UML language is shown in **Fig. 3.3**.

Based on the results obtained, a method was developed for automatically solving the problem of assigning IT project tasks to its performers. Using this method makes it possible to exclude human participation in the process of directly solving the problem of assigning IT project tasks to its performers (in the process of performing Stage 4 of the method). This makes it possible to obtain a solution to this problem in a sufficiently short period of time and reduce the time spent on planning individual sprints (iterations) of the IT project.

Another significant advantage of the developed method is a fairly high level of objectivity of the results of task distribution. The use of a modified algorithm for solving the classification problem based on the adapted MNB allows making decisions based on statistical data and historical information, which minimizes the possibility of subjective errors and bias. This ensures transparency of the task distribution process, which in turn increases trust on the part of the performers and helps to increase their motivation.

Unlike the solution to this problem described in [15], the proposed method allows at least partially to take into account the experience and competence of the performers of the IT project. The experience of the performers when solving the problem is taken into account by selecting for a specific performer those tasks whose names largely coincide with the names of tasks successfully completed by this performer earlier. The competence of the performers when solving the problem is taken into account by assigning to each specific performer their own value of the

indicator "Maximum permissible weight of class e_k ," $V_{\max k}$. This value is set based on the position held by the e_k performer in the team based on their own competencies.

The use of MNB to solve the problem of assigning IT project tasks to its performers made it possible, in contrast to the solution described in [16], to abandon vector descriptions of IT project tasks and the calculation of the cosine similarity measure. The calculation of the scalar elements of the classification rule (3.7) and condition (3.10) requires less computational resources and requires less time to calculate the result of solving the problem.

The main limitation of the application of the developed method is the constant need to maintain the historical data repository, which is used to train the adapted MNB, in an up-to-date state. This data should be collected by analyzing previous IT projects, assessing the success of individual tasks, and providing feedback from the teams of performers. Regular updating and validation of historical data are critically important for ensuring the accuracy and relevance of the results of solving the problem of assigning IT project tasks to its performers.

Among the limitations and disadvantages of the application of the algorithm for implementing the developed method, it is also worth noting some difficulties with the classification of tasks that have similar names, but differ in content. This can lead to erroneous distribution of tasks between individual performers.

To improve the accuracy of classification, it is necessary to use a larger amount of historical data for training the model. Regular updating and expansion of the keyword dictionary will help the algorithm better understand the context of the tasks. Therefore, work on the creation and development of a specialized thesaurus, on the basis of which the training set for MNB should be formed, is one of the promising areas of further research on this issue.

Another area of further research into solving the problem is the use of modern artificial intelligence tools to increase the accuracy and objectivity of solving the problem of assigning IT project tasks to its performers. In particular, it is about the possibility and feasibility of using lemmatization for pre-processing text names of individual IT project tasks. Although one of the authors of this study has conducted work in this direction [34], it requires further development for successful use in the field of ongoing management of IT projects and their performers.

3.7 Conclusions

As a result of the research, a method for solving the problem of assigning IT project tasks to its performers was developed. The use of this method requires human

participation only to perform operations in preparation for the distribution of IT project tasks among the members of the team of performers. The actual solution to the problem of assigning IT project tasks to its performers using the developed method is carried out without the participation of stakeholders of the IT project that is planned and/or being performed.

During the development of the method, MNB was adapted to the specifics of the problem of assigning IT project tasks to its performers. As a result of this adaptation, the classification rule with a minimum error (3.7) was clarified by introducing an additional condition (3.10). This condition allows taking into account the level of competence of the IT project performer when solving the classification problem as the maximum permissible efforts of the tasks assigned to it for performing. The features of calculating the elements of the classification rule (expressions (3.8) and (3.9)) were also clarified. This made it possible to adapt the classification rule, proven in practice, to the task of assigning IT project tasks to its performers and to modify the algorithm for solving the classification problem accordingly.

Based on the results obtained, a general description of the method for solving the task of assigning IT project tasks to its performers was developed. For a detailed description of the content of individual stages, an algorithm for implementing this method was developed. The diagram of this algorithm shown in **Fig. 3.3** and the proposed descriptions of the main steps of this algorithm determine the features of its implementation both as a methodology for applying the obtained solutions in the current management of an IT project and as a specialized information technology.

To verify the operability of the obtained results, an experimental test of the method and its implementation algorithm was conducted during the management of one of the IT projects of an outsourcing IT company. The course of using the method and the main results of the implementation of its individual stages are presented. These results include:

- determination of teams of IT project performers and the composition of these teams, taking into account the level of gradation of individual performers;
- formed training set and results of its tokenization;
- "Product Backlog" information artifact as a detailed list of IT project tasks, ordered by the values of their performing priorities;
- example of iterations of Stage 4 of the developed method;
- results of calculating the values of the classification rule for all tasks and IT project performers;
- results of distributing IT project tasks to performers from selected teams;
- "Sprint Backlog" information artifacts as task packages for each team of performers.

The results of experimental verification indicate the feasibility of using the developed method to solve the problem of assigning IT project tasks to its performers. The developed method contributes to better project planning, minimizes administrative burden and helps to avoid delays and errors in project performing. The method is easily adapted to different projects and teams due to the ability to adjust the algorithm parameters in accordance with specific features.

The authors consider it advisable to continue the research to conduct a comparative analysis of the results of using different text classification algorithms to solve the problem of assigning IT project tasks to its performers.

Conflict of interest statement

The authors declare that there is no conflict of interest in relation to this paper, as well as the published research results, including the financial aspects of conducting the research, obtaining and using its results, as well as any non-financial personal relationships.

Use of artificial intelligence statement

The authors declare that they did not use artificial intelligence tools in preparing this manuscript.

References

1. Agrawal, M., Chari, K. (2007). Software Effort, Quality, and Cycle Time: A Study of CMM Level 5 Projects. *IEEE Transactions on Software Engineering*, 33 (3), 145–156. <https://doi.org/10.1109/tse.2007.29>
2. Chilton, M. A. (2022). Resource allocation in IT projects: using schedule optimization. *International Journal of Information Systems and Project Management*, 2 (3), 47–59. <https://doi.org/10.12821/ijispm020303>
3. Herroelen, W., Leus, R. (2005). Identification and illumination of popular misconceptions about project scheduling and time buffering in a resource-constrained environment. *Journal of the Operational Research Society*, 56 (1), 102–109. <https://doi.org/10.1057/palgrave.jors.2601813>

4. Kolisch, R. (1996). Serial and parallel resource-constrained project scheduling methods revisited: Theory and computation. *European Journal of Operational Research*, 90 (2), 320–333. [https://doi.org/10.1016/0377-2217\(95\)00357-6](https://doi.org/10.1016/0377-2217(95)00357-6)
5. A guide to the project management body of knowledge (PMBOK guide) (2017). Project Management Institute, Inc.
6. Nastanova do zvodu znan z upravlinnia proiektamy. Nastanova PMBOK (2021). Project Management Institute, Inc. Available at: https://learn.ztu.edu.ua/pluginfile.php/274061/mod_resource/content/1/PMBOK7_Ukr_ForPersonalUseOnly.pdf
7. Ruiz, S., Escudero, D., Cervantes, J., Trueba, A.; Figueroa-García, J., López-Santana, E., Villa-Ramírez, J., Ferro-Escobar, R. (Eds.) (2017). Assigning-Tasks Method for Developers in Software Projects Using up Similarity Coefficients. *Applied Computer Sciences in Engineering*. Springer, 119–128. https://doi.org/10.1007/978-3-319-66963-2_12
8. Ievlanov, M. V., Pogorelaya, N. I. (2012). Planning for the use of staff in the works of IT-project. *Eastern-European Journal of Enterprise Technologies*, 2 (4 (56)), 22–26. <https://doi.org/10.15587/1729-4061.2012.3706>
9. Martínez-Rojas, M., Soto-Hidalgo, J. M., Marín, N., Vila, M. A. (2018). Using Classification Techniques for Assigning Work Descriptions to Task Groups on the Basis of Construction Vocabulary. *Computer-Aided Civil and Infrastructure Engineering*, 33 (11), 966–981. <https://doi.org/10.1111/mice.12382>
10. Schnabel, A., Kellenbrink, C., Helber, S. (2018). Profit-oriented scheduling of resource-constrained projects with flexible capacity constraints. *Business Research*, 11 (2), 329–356. <https://doi.org/10.1007/s40685-018-0063-5>
11. Skiena, S. S. (2020). *The Algorithm Design Manual*. Texts in Computer Science. Springer International Publishing. <https://doi.org/10.1007/978-3-030-54256-6>
12. Vera-Rivera, F. H., Barbosa-Mora, J. L., Gaona-Cuevas, C. M. (2020). Generación automática de la planificación de la entrega en desarrollo de software ágil, asignación de historias de usuario a los desarrolladores usando algoritmos genéticos. *AiBi Revista de Investigación, Administración e Ingeniería*, 8 (2), 29–38. <https://doi.org/10.15649/2346030x.735>
13. Homwiseswongsa, A., Ratanavilisagul, C. (2023). Modified Differential Evolution Algorithm for Solving Multi-Skill Resource-Constrained Project Scheduling Problem. 2023 15th International Conference on Information Technology and Electrical Engineering (ICITEE). Chiang Mai, 1–6. <https://doi.org/10.1109/icitee59582.2023.10317769>

14. Xu, H., Kuchansky, A., Biloshchytska, S., Tsiutsiura, M. (2021). A Conceptual Research Model for the Partner Selection Problem. 2021 IEEE International Conference on Smart Information Systems and Technologies (SIST). Nur-Sultan, 1–6. <https://doi.org/10.1109/sist50301.2021.9465931>
15. Druzhyinin, V., Gladka, M., Borysenko, I., Hladkyi, Ya., Lisnevskiy, R. (2024). A Model for Allocating Labor Resources to Project Work Based on Task Prioritization. Information Technology and Implementation Workshop 2024: IT Infrastructure and Applied Solutions, IT and I-WS 2024: ITIAS. CEUR Workshop Proceedings, 3955, 42–54.
16. Arslan, H., Işık, Y., Görmez, Y., Temiz, M. (2024). Machine learning and text mining based real-time semi-autonomous staff assignment system. Computer Science and Information Systems, 21 (1), 75–94. <https://doi.org/10.2298/csis220922065a>
17. Mitchell, T. M. (1997). Machine Learning. McGraw-Hill, 414.
18. Manning, C. D., Raghavan, P., Schuetze, H. (2008). Introduction to Information Retrieval. Cambridge University Press. <https://doi.org/10.1017/cbo9780511809071>
19. Gholamy, A., Kreinovich, V., Kosheleva, O. (2018). Why 70/30 or 80/20 Relation between Training and Testing Sets: A Pedagogical Explanation. International Journal of Intelligent Technologies and Applied Statistics, 11 (2), 105–111. [https://doi.org/10.6148/IJITAS.201806_11\(2\).0003](https://doi.org/10.6148/IJITAS.201806_11(2).0003)
20. Tan, H. (2021). Machine Learning Algorithm for Classification. Journal of Physics: Conference Series, 1994 (1), 012016. <https://doi.org/10.1088/1742-6596/1994/1/012016>
21. Kulkarni, A., Brown III, L. L. (2019). Phishing Websites Detection using Machine Learning. International Journal of Advanced Computer Science and Applications, 10 (7), 8–13. <https://doi.org/10.14569/ijacsa.2019.0100702>
22. Rennie, J. D., Shih, L., Teevan, J., Karger, D. R. (2003). Tackling the poor assumptions of naive bayes text classifiers. ICML'03: Proceedings of the Twentieth International Conference on International Conference on Machine Learning, 3, 616–623. Available at: <https://dl.acm.org/doi/10.5555/3041838.3041916>
23. ISO/IEC/IEEE Standard No 15288:2015 (2015). Systems and software engineering – System life cycle processes. ISO/IEC/IEEE International Standard. <https://doi.org/10.1109/IEEESTD.2015.7106435>
24. Schwaber, K., Sutherland, J. (2020). The Scrum Guide. The Definitive Guide to Scrum: The Rules of the Game. Available at: <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf>

25. Sutherland, J. (2014). *Scrum: A revolutionary approach to building teams, beating deadlines and boosting productivity*. Random House, 248.
26. Jarzębowicz, A., Sitko, N. (2020). Agile Requirements Prioritization in Practice: Results of an Industrial Survey. *Procedia Computer Science*, 176, 3446–3455. <https://doi.org/10.1016/j.procs.2020.09.052>
27. What is Product Owner? Agile Alliance. Available at: <https://www.agilealliance.org/glossary/product-owner/>
28. Yang, A. (2023). Guide to building a product roadmap (with template and examples). LogRocket. Available at: <https://blog.logrocket.com/product-management/product-roadmap-template-examples/>
29. Radigan, D. Product backlog: tips for creation and prioritization. Atlassian. Available at: <https://www.atlassian.com/agile/scrum/backlogs>
30. McCallum, A., Nigam, K. (1998). A Comparison of Event Models for Naive Bayes Text Classification. *AAAI/ICML-98 Workshop on Learning for Text Categorization*, p Technical Report WS-98-05. AAAI Press, 41–48.
31. Iparraguirre-Villanueva, O., Melgarejo-Graciano, M., Castro-Leon, G., Olaya-Cotera, S., Ruiz-Alvarado, J., Epifanía-Huerta, A. et al. (2023). Classification of Tweets Related to Natural Disasters Using Machine Learning Algorithms. *International Journal of Interactive Mobile Technologies (IJIM)*, 17 (14), 144–162. <https://doi.org/10.3991/ijim.v17i14.39907>
32. Cohn, M. (2023). What Are Agile Story Points? Mountain Goat Software. Available at: <https://www.mountaingoatsoftware.com/blog/what-are-story-points>
33. Planning (Scrum) Poker (2023). QATestLab. Available at: <https://training.qatestlab.com/blog/technical-articles/planning-poker/>
34. Ievlanov, M. V., Moroz, B. I., Moroz, D. M., Luchytskyi, V. V. (2024). Information technology for identifying terms and project artifacts in the requirements for the information system. *Management Information System and Devises*, 182, 73–93. <https://doi.org/10.30837/0135-1710.2024.182.073>

CHAPTER 4

Method for assessing the complexity of an IT project for video game localization

Maksym Yevlanov
Nataliia Vasylytsova
Iryna Panforova
Dmytro Osypchuk

Abstract

The object of the study is the process of project assessment and control.

During the study, the task of assessing the complexity of an IT project for video game localization was solved. In this direction, there has been little research recently conducted in the IT industry, mainly aimed at solving organizational problems of managing relevant IT projects. Research aimed at solving problems of assessing time spent on implementing IT projects for video game localization is mainly of an applied nature. Therefore, there is a constant need to conduct scientific research and process the experience of individual specialists and IT companies acquired during the planning and implementation of IT projects for video game localization.

According to the results of the study, a set of main factors that influence the implementation of a third-level IT project for video game localization was identified, and a system for their classification was proposed. Based on the results of identifying and classifying these factors, a method for assessing the complexity of an IT project for video game localization was developed. This method consists of four stages, which, in turn, are divided into 12 activities. The features of each of the activities of the developed method are described in detail.

Experimental verification of the obtained results was carried out in two stages. At the first stage, the developed method was used to assess the duration of the Ukrainian localization of the video game "Don't Starve Together". The predicted assessment of the translation duration of this game was 32 hours 57 minutes. The actual translation duration was 26 hours 26 minutes. The predicted translation duration exceeded the actual one by 24.7%. At the second stage, the effectiveness of using the localization characteristics obtained during the assessment of previous

IT projects was tested in the project on the Ukrainian localization of the game "Sid Meier's Civilization® VI". The predicted translation duration was 11 hours 11 minutes. The actual translation duration was 9 hours 57 minutes (the difference with the predicted value was 12.1%).

Additionally, the assessment results obtained using the developed method were compared with the assessment results obtained using the common methodology for assessing time spent on software localization. The results showed that using the developed method allows for the most accurate estimates of time spent on third-level IT projects for video game localization.

Keywords

Localization, assessment, IT project, translation unit, video game.

4.1 Introduction

Developers of modern IT products are increasingly focused on consumers from the global, rather than local market. However, the adaptation of IT software products to local markets (localization) remains a necessity caused by the prevalence of languages, cultural diversity and the specifics of local legislation. One of the most complex types of such IT products, from the point of view of localization, are computer games (or video games). The term "computer game" should be understood as a separate class of application software that involves the interaction of the user (player) with other players or a computer in order to achieve a goal set by the rules of the game [1].

Video game localization is a relatively new area of activity in the IT industry. Researchers have recognized that since the 2000s. IT companies have begun to apply a new approach known as "deep localization" [2, 3]. This approach requires bringing the game closer to the consumer in each individual localization (everything that does not contradict the game world itself and can facilitate player immersion can be revised and adapted in accordance with what is considered a more successful local influence) [3]. Therefore, modern research suggests dividing the global problem of "deep localization" into two separate aspects [2, 4, 5]:

- "linguistic" localization (which mainly concerns the translation process);
- "product localization" (which consists in adapting a product or service to the culture and language of customers in a specific target market [4, 5], even deeply changing the relevant structural and artistic aspects of the product).

IT projects for video game localization also differ significantly from other IT projects for the creation, modification or development of IT products. Among the

differences of IT projects for video game localization, it is worth noting the special requirements that are imposed on the planning of such projects. In particular, during planning, an important task is to assess the complexity of video game localization. Such an assessment requires a deep understanding of the various factors that affect the successful implementation of relevant IT projects. An objective and accurate assessment of the complexity of video game localization allows for increased accuracy in planning time and personnel needs for an IT project, and also affects the accuracy of assessing financial costs for this IT project.

The current state of research in the field of software localization for IT products and video games, as a separate class of such products, demonstrates some progress in the development of methods and tools for assessing the complexity of IT projects. Separate works by such leading institutions and organizations as the Localization Industry Standards Association (1990–2011), the International Game Developers Association (IGDA), and some localization service providers are devoted to solving the issues of effective management and assessment of localization projects. However, the problem of increasing the accuracy of assessing the complexity of IT projects for video game localization has not been fully resolved, in particular, due to the growth of globalization processes and the complexity of video games. At the same time, given the high requirements for translation quality and technical adaptation, correct assessment of complexity becomes critically important for the successful management of modern IT projects. Accurate forecasting of localization operations is critically important, since errors in assessment can lead to underassessment or overassessment of resources (team, budget, etc.) or time required for project implementation. These errors, in turn, negatively affect project planning processes and end-user satisfaction. Therefore, research on the development of new and improvement of existing methods for assessing the complexity of IT projects for video game localization is relevant from both theoretical and applied points of view.

4.2 Analysis of current research in the field of software and video game localization

In general, the number of studies in the field of software and video game localization should be recognized as insufficient. Thus, in [2] it was recognized that for the period 1981–2021 in well-known journals, conferences and seminars indexed in the three main scientific search engines (Scopus, WoS and Google Scholar), only 45 studies were published on this topic. A similar conclusion was made in [6]: out of 468 researched scientific articles devoted to the processes of game development,

localization was considered in only a few of these articles. At the same time, attention was mainly paid to the following issues [6]:

- research on regional aspects of game localization;
- research on individual concepts related to game localization.

An important problem was and remains the problem of effective organization of localization work. An equally important issue is the possibility of connections between localization and other work of an IT project on video game development. Unfortunately, in the early 2010s, the relationship between localization and software development processes was largely ignored in research. In practice, localization was often separated from development [4]. In 2013, [7] proposed an extended video game value chain diagram that included a game localizer (Fig. 4.1).

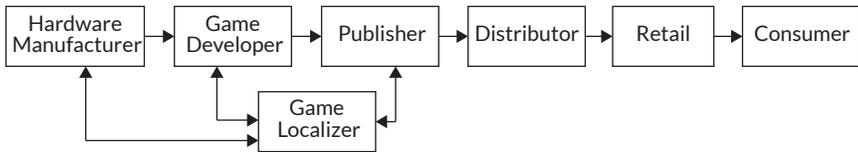


Fig. 4.1 Diagram of the extended video game value chain
Source: [7]

As can be seen from this diagram, the game localizer function is excluded from the main chain of the IT company. In the process of localizing a game, the localizer could interact with the publisher, the game developer, and the hardware manufacturer [7].

The reasons for this separation are the perception of localization as a labor-intensive and expensive activity, often performed by external partners of IT software development companies. Such external partners have limited access to developers and rely on documentation and formalized processes. This contrasts with the minimal documentation and on-site presence cultivated in Agile IT project management methods [8].

This situation has hardly improved in the first half of the 2020s. In [9] it is noted that as of 2023, there are almost no studies that analyze localization from the perspective of video game development. According to the results of a survey of developers of video games available in Spanish on Steam, in [9] it was found that the choice of translator is influenced by the size and experience of the development company. The earlier the localization is planned, the greater the likelihood that the game will be translated by a professional who cannot play the game before its translation. In general, from the results of the processing of the surveys presented in [9], it is

possible to conclude that it is desirable to consider the localization of a video game as a subproject or a separate IT project, which is carried out in parallel with the IT project for the development of the video game. However, the models and methods of managing an IT project for the localization of a video game are formed exclusively on the basis of the private experience of individual specialists and IT companies and have hardly been studied by scientists.

This conclusion is confirmed by the results of field research by a specialist in the field of video game localization M. Toftedahl. Based on the results of this research, published in [10], he proposed to present the organizational structure of an IT company for the development of games in the indie sector in the form of a diagram shown in Fig. 4.2.

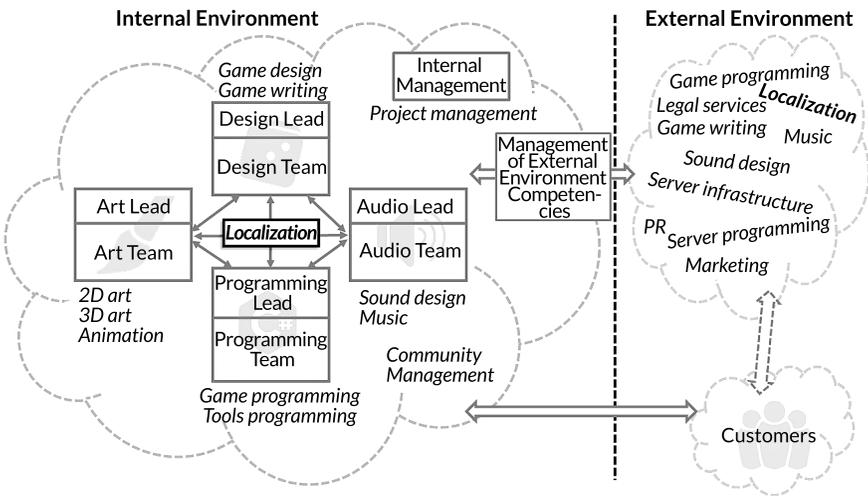


Fig. 4.2 Structure of an indie game development company based on field research in Sweden and China
Source: [10]

As can be seen from the diagram in Fig. 4.2, game localization is a work or separate IT project that is performed both by the IT development company and by the external (relative to the IT development company) environment. The interaction of the IT development company with such external projects in [10] is proposed to be organized through the management of the competencies of the performers of these external projects. However, in the studied cases of video game localization, the results showed that localization is a more demanding IT project than initially planned.

Even if the support of competencies from the external environment is used, management and development should also largely take place in the internal environment of the IT development company [10].

In general, based on the results of the research conducted in [10], the following general conclusions were formulated:

- game localization is one of the important development tasks that must be planned in a timely manner;
- game localization is a task which complexity is underestimated;
- localization is not the main (critical) element of the IT project for developing an indie video game, it is implemented as needed;
- in the context of indie game development, game localization is synonymous with translation, more complex aspects of localization related to game mechanics and game design (the so-called culturalization of the gameplay) require additional research.

The factors identified as contributing to success in game localization in [10] are:

- cultural sensitivity;
- effective communication and collaboration;
- use of localization tools and technologies.

The author of [10] believes that these factors are likely to be relevant to game localization efforts in different contexts and game genres.

Among the problems faced by game localization professionals, [10] specifically mentions:

- tight localization deadlines;
- the need for effective localization project management.

These problems, according to the author of [10], are likely common to the entire industry.

The main tools used in the process of video game localization will remain machine and automated (computer-aided translation, CAT) translation systems. However, the use of such systems has its own characteristics, which depend on both the translator and the language into which the text is translated. The experience of using machine translation systems in the process of software localization under the conditions of daily work of professional translators for languages with low resources and rich morphology (Czech, Estonian, Hungarian, Latvian, Lithuanian and Polish) is reviewed in [11]. The results of experimental studies showed that the use of machine translation suggestions in addition to the use of CAT increased the productivity of translators in all assessment experiments. However, significant differences were observed in the productivity of different translation tasks and in the results of different translators. In [11] it was proposed to explain these differences

by the work mode and skills of individual translators. It should also be noted that the error rate increased in all of the experiments (except one), while remaining at the "Good" quality level. This result was explained in [11] by the fact that translators tend to trust the suggestions coming from the machine translation tool and do not check them sufficiently, even if they are marked as machine translation suggestions. It should be recognized that with the development of the capabilities of artificial intelligence tools, interest in the use of such tools in the field of software and video game localization has increased. Thus, in [12] an example of tuning the original neural machine translation model RNN-encoder-decoder with the addition of categorical information related to the problem of rare words (words that relate to a certain subject area, which is a common phenomenon in the text of software) is considered. However, this model has not been tested for video game localization and, according to the authors, provides successful translation results mainly for English and other official languages used by the United Nations [12].

The use of GPT capabilities for video game localization was investigated in [13]. The results show that the GPT-4 model is capable of providing multilingual output in a structured form. The GPT-4 model seems to have the ability to support the resolution of transcreation-related problems such as brainstorming, comparison, and coherence, although the results are not always satisfactory, and there are errors including conceptual repetition, formatting issues, and inappropriate or unusual portmanteau. In terms of output quality, the English-Chinese combination appears to be less creative compared to adaptations from English into French or Spanish. However, to draw any conclusions on this issue, further comparative qualitative and quantitative studies are needed, as well as reception studies among users to find out their preferred translations, as creativity is a highly subjective matter [13].

However, the use of GPT-4 for video game localization is not free from many drawbacks, among which [13] particularly highlight the following:

- the strong dependence of localization performance on the specific arrangement of tokens in the GPT-4 model;
- the limitation of the amount of context that can be used to improve coherence can lead to transcreations that deviate from previous work or do not correspond to the setting of the video game;
- GPT models can produce hallucinatory or nonsensical results or contain grammatical or spelling errors;
- GPT models may not take into account culturally sensitive issues, which can lead to the perpetuation, reproduction, and even reinforcement of stereotypes and prejudices that were present in the training data of these models;

- the different amount of data available in different languages affects the quality of results for certain language pairs during translation and cultural adaptation;
- GPT models risk violating copyright and intellectual property laws, which can expose users to financial and legal consequences;
- GPT models are currently limited by paid access, and the number of monthly messages allowed by more powerful models is limited, which negatively affects the localization market and workforce;
- opacity, together with the randomness of the results, limits the interpretability and applicability of the GPT model, as well as the reproducibility of the results.

However, the authors of the study [13] note that the results obtained and the identified limitations of the application of GPT models for video game localization were established by them on the example of one video game. Therefore, additional research is required to clarify the conclusions regarding the possibility and feasibility of using GPT models for video game localization. In general, [13] recommends using the capabilities of GPT models to interact with such categories of localization specialists as interns and beginners. Thanks to the capabilities of GPT models, these categories of specialists can increase their productivity and creativity, while maintaining control over the result using spoken machine transcreation based on prompts [13].

Summing up the interim results of the analysis of research on game localization tools, it should be recognized that the main localization tools remain the human translator and the CAT and machine translation tools he/she uses. The main advantages of using such tools are their speed and cheapness compared to human translation. However, the use of machine translation has a number of disadvantages, among which it is especially worth noting:

- the impossibility of a complete and high-quality understanding of the context;
- the inability to interpret idioms and cultural features;
- the inability to convey the literary features of the text;
- difficulties with the translation of highly specialized texts;
- the inability to be creative.

Therefore, it is impossible to fully localize a video game using machine translation tools, especially while maintaining the consistency of terminology, ensuring its correctness, and taking into account textual and linguistic features. Editing such a text requires a large number of editors, and its result will contain more undetected errors. To combine the advantages of each method, machine translation tools can be used in some cases at the translator's discretion to speed up the translation of individual translation units, rather than the text as a whole. The term "translation unit" (TU) should be understood as the smallest indivisible text that can be represented by a word, phrase, sentence, or set of sentences [14].

Thus, based on the results of the analysis, it is possible to conclude that there is a need for a modern perception of video game localization as a separate type of IT project. The main resources of these IT projects are game text translation specialists and CAT and machine translation tools. Modern Agile and hybrid methodologies can be used for the general management of such IT projects. However, specific methods and models for assessing, planning, and managing the work of such projects remain almost unexplored.

Therefore, the aim of this study is to develop a method for assessing the complexity of video game localization work with increased accuracy. This increase is planned to be achieved by taking into account a larger number of factors affecting the complexity of localization work than in currently existing methods.

To achieve the aim, it is proposed to solve the following objectives:

- develop a method for assessing the complexity of an IT project for video game localization;
- carry out experimental verification of the developed method.

4.3 Basic research method

The object of the study is the process of project assessment and control. The purpose of this process is to ensure the balance and feasibility of plans, determine the status of the project, its technical implementation and implementation of processes, ensure implementation according to plans and schedules within the projected budgets for technical tasks [15].

The main hypothesis of the study is the hypothesis of the possibility of increasing the accuracy of forecasting the duration of the IT project for video game localization by means of:

- research of factors that affect the complexity of localization work;
- development of a method for assessing the complexity of the IT project for video game localization.

The main assumption of the study is the assumption about the method of implementing the IT project for video game localization. According to the results of the analysis of modern scientific research presented in **Section 4.1**, this method is considered as the translation of game texts by a human translator using machine translation tools.

A study of current experience in quantitatively assessing time costs as an indicator of the complexity of an IT project for video game localization [16–19] showed that the main method of such assessment remains the method based on

the number of words. This method is based on calculating the indicator of translation time costs t_{est} according to the formula

$$t_{est} = w/s, \quad (4.1)$$

where w – the number of words in the text being translated; s – the recommended translation speed.

Different variations of this methodology can use different analogues of the indicators of the number of words in the text being translated and the recommended translation speed. Thus, instead of the indicator "Number of words in the text being translated", the following indicators of text volume can be used:

- "Total number of words in the localization file" (indicator $W1$);
- "Number of unique words in the localization file" (indicator $W2$);
- "Number of words in unique TUs of the localization file" (indicator $W3$).

The software translation speeds recommended in [16–19] can be characterized by the following indicators:

- "Interface translation speed" (indicator $S1$) [16, 17];
- "Speed taking into account the complexity of the text" (indicator $S2$) [18];
- "Baseline speed" (indicator $S3$) [19].

A comparison of the above speed indicators is presented in **Table 4.1**.

The considered indicators $W1$ – $W3$ and $S1$ – $S3$ can be compared depending on whether they take into account certain factors of localization complexity. The results of the comparison of the complexity factors considered in [16–19], which are taken into account by variations of the method based on the number of words, are given in **Table 4.2**.

The variants of the translation duration assessment methodology defined in [16–19] offer slightly different indicators for assessing the translation duration. However, all of these variants state that the duration of editing is equal to half the duration of the text translation work.

Table 4.1 Comparison of translation speed indicators proposed in [16–19]

Indicators	Recommended speed, words/day	Recommended speed, words/hour	Availability of variability
$S1$	2000	250	Present (depends on the type of content)
$S2$	1200–4000	200–500	Present (depends on difficulty)
$S3$	2500	300	Absent

Source: [16–19]

Table 4.2 Results of the comparison of the complexity factors, taken into account by variations of the method based on the number of words

Factor	Indicator		
	W1	W2	W3
Text volume	Takes into account	Takes into account	Takes into account
Text complexity	Doesn't count	Doesn't count	Doesn't count
Speed of the method application	Very high	High	High
Need for preliminary data processing	Absent	Necessary	Necessary
Taking into account the linguistic features of the text	Doesn't count	Doesn't count	Doesn't count
Uniqueness of the text	Doesn't count	Takes into account	Partially takes into account

Factor	Indicator		
	S1	S2	S3
Text volume	Takes into account	Takes into account	Takes into account
Text complexity	Partially takes into account	Takes into account	Doesn't count
Speed of the method application	High	Low	Very high
Need for preliminary data processing	Absent	Necessary	Absent
Taking into account the linguistic features of the text	Doesn't count	Partially takes into account	Doesn't count
Uniqueness of the text	Doesn't count	Partially takes into account	Doesn't count

The main drawback of the considered indicators of text volume and translation speed is that they do not take into account the complexity of the text (with the exception of S2). The factors considered in **Table 4.2** are sufficient for comparing the indicators with each other. However, the list given in **Table 4.2** is not an exhaustive list of factors that determine the duration and complexity of translation work.

According to the data in **Table 4.1**, it can be concluded that in [16–19] the recommended translation speed is set at or close to 300 words per hour. In cases where the translation work is already in progress, the average translation speed can be taken as the recommended speed. However, this indicator will be individual for each individual translator.

The review of the methodology based on the number of words allowed to draw the following conclusions:

- this methodology was and remains the most common means of calculating the duration of an IT project for the localization of an IT product;
- this methodology does not take into account a number of indicators that directly affect the duration of translation;
- this methodology is a quick, but not accurate way of calculating the predicted duration of translation in an IT project for the localization of an IT product;
- among the studied variations of this methodology, only variations based on the S2 indicator take into account the complexity of the text when calculating the duration of the localization of an IT product.

Therefore, when developing a method for assessing the complexity of an IT project for the localization of a video game, the considered features of the methodology based on the number of words should be taken into account in order to eliminate the identified shortcomings of this methodology.

It is proposed to divide the experimental verification of the research results into two stages:

- the main stage;
- the additional stage.

The purpose of the main stage is to verify the effectiveness of the developed method for assessing the complexity of an IT project for video game localization in comparison with the existing method based on the number of words. To verify this effectiveness, it is proposed to use the following indicators: absolute localization duration, relative error in the prediction of localization duration, relative average error in the translation duration of localization files.

The purpose of the additional stage is to verify whether the values of the proposed individual translation speed coefficient (ITSC), which were obtained within the IT project for the localization of one video game, can be used in the IT project for the localization of another video game. This will allow to investigate the possibility of applying the proposed ITSC in games that differ in genres and text styles.

Crowdin [20] was chosen as the CAT tool for video game localization used during the experimental verification of the research results. It is a cloud platform for managing translations and localization. It is suitable for localizing desktop and mobile applications, websites, and software documentation. Crowdin is a Ukrainian development and performs the same functions as Phrase Strings [21], but has both a paid subscription and a free version. The free version allows to create public and private projects up to 60,000 words. Crowdin contains information on over 176,000 localization projects and over two million registered users.

4.4 Results of the development of a method for assessing the complexity of an IT project for video game localization

4.4.1 Identification of factors affecting the complexity of video game localization work

As noted above, the list given in **Table 4.2** is not an exhaustive list of factors that determine the duration and complexity of translation work. Therefore, to develop a method for assessing the complexity of an IT project for video game localization, a special study was conducted to identify the main classes and groups of factors that can affect the progress and results of such an assessment [22].

According to the results of this study, it is proposed to divide the entire set of main factors that affect the progress and results of assessing the complexity of an IT project for video game localization into three main classes (**Fig. 4.3**):

- class of technical factors;
- class of factors set by the customer;
- class of factors determined by volume and content limitations.

It is proposed to include a set of limitations in the class of technical factors, which are primarily determined by the architecture of the IT product being localized, as well as the capabilities of the software used for localization. It is proposed to divide this set into the following groups of factors:

- "Selected CAT tool";
- "Localization file structure";
- "Variable organization";
- "Limitations on the number of characters".

The group of factors that describe the selected CAT tool characterize the environment in which the members of the localization team will work. A common feature for all CAT tools is that they allow to upload localization files in the original language, translate and edit text in a convenient interface, and at the output receive already formatted localization files in the target language. Most of these tools have the ability to connect translation memory and machine translation, distribute tasks between employees, track project progress, and allow to create and use dictionaries and glossaries.

The main differences between CAT tools are in tariff plans and functional features, such as: version control systems, automatic payment generation, project and participant statistics, etc.

The group of factors "Localization file structure" is determined by the developer. It is proposed to divide this group into the following subgroups of factors: distribution of TUs by files, quality of description of individual TUs.

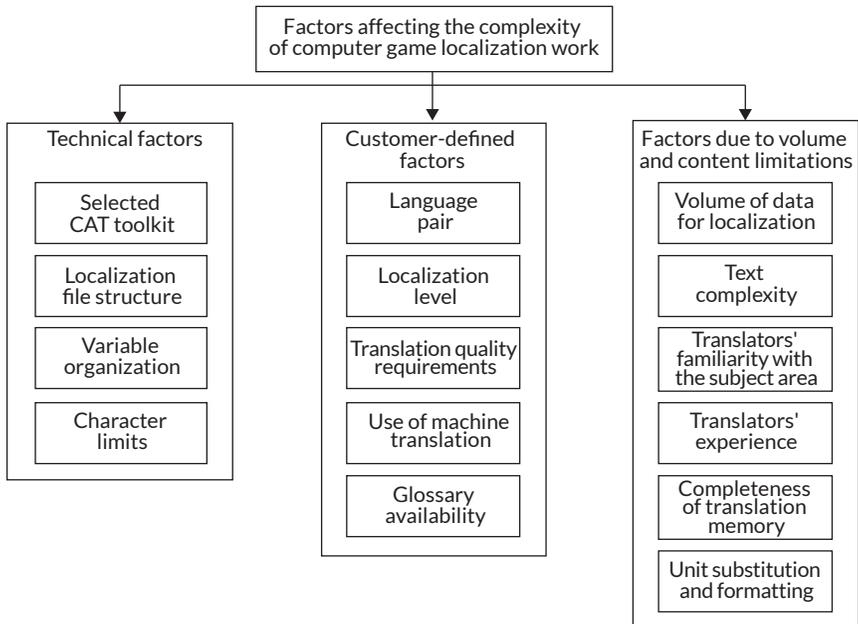


Fig. 4.3 Classification scheme of the main factors that influence the progress and results of the assessment of the complexity of an IT project for video game localization

The subgroup of factors "Distribution of TUs by files" characterizes how exactly TUs are distributed by developers across the file system. This distribution can occur in one of three ways:

- all TUs are contained in a single localization file;
- TUs are distributed across a set of several files;
- TUs are stored in a folder tree, where each folder is responsible for a separate group of localization files, united by common features.

The quality of the description of individual TUs involves their meaningful naming. Such naming simplifies the translator's understanding of the context and simplifies finding the TU's place in an IT product (in particular, in a video game).

If the localization files are organized as a tree, it is possible to determine the purpose of each TU group. Each TU has a key by which it is stored and inserted into the IT product. If the keys have meaningful names, the localization team can automatically distribute TUs to localization files. This will allow each employee to be responsible for translating files containing related TU groups. In addition, such a structure

allows to quickly understand the context of the TU being translated and, if necessary, find the desired TU in the IT product.

The subgroup of factors "Variable organization" characterizes the quality of variable naming and the number of these variables. This affects readability and understanding of the context. A large number of variables in a TU negatively affects its readability, as does their meaningless naming. High-quality organization of variables allows to better understand the context in which the TU is used, simplifies the coordination of words in a sentence, and increases its readability by translators and editors.

The subgroup of factors "Character Limits" characterizes the correspondence of the size of the translated content to the interface element in which it needs to be placed. First of all, the restriction is imposed on those TUs that are used to describe user interface elements. This achieves the correctness of the display of interface elements and texts. However, such coordination complicates the localization process by the need to fit the translated text into a limited set of characters. This may require the use of synonyms, alternative forms, abbreviations, but should not affect the content embedded in the TU.

The class of factors established by the customer proposes to include a set of restrictions that guide the localization process of an IT product and are determined by the customer. This set is proposed to be divided into the following groups of factors:

- "Language pair";
- "Localization level";
- "Localization quality requirements";
- "Permission to use machine translation";
- "Glossary availability".

The group of factors "Language pair" characterizes the pair of languages with which the translator works, namely: the original language and the target language. The analysis showed that today for most software products that are localized, the original language is English, however, in some cases the original language may be different. The localization process can be complicated depending on the language pair, namely the features of the target language and the original language, which differ. Such features may include:

- belonging of a pair of languages to different groups that determine the way words are formed in the language (agglutinative, inflectional or root languages);
- differences in the agreement of parts of a sentence depending on the gender and number of the noun;
- direction of writing the text, etc.

The group of factors "Localization level" characterizes the type of content that will be localized. In total, it is proposed to consider four levels of localization of video games, the choice of which depends on the volume of content to be localized [23].

The first level of localization involves the availability of the product in the local market in its original form. With the advent of digital distribution platforms such as Steam, Xbox, etc., the distribution of games has become much easier and does not require additional costs for delivering physical media to stores.

The second level of localization involves the localization of the game packaging (in the case of digital media – the store page) and documentation, but not the game content itself.

The third level of localization involves full text localization of the game, but not the sound. In this case, the interface and subtitles of the game must be fully localized.

The fourth level of localization involves full localization of the game, which includes text localization, localization of sound, graphical elements of the game, localization of documentation and other additional materials.

Thus, the first level of localization does not require translation, the second and third levels affect the amount of data for localization, which depends on the size of the project team. However, if the localization level is defined as the fourth, then the localization project team is joined by voiceover specialists (voiceover team), which will require additional interaction and cooperation with this team.

This study proposes to focus on the third localization level.

The group of factors "Localization quality requirements" is determined by the customer and depends on the budget and time allocated to the IT localization project. Depending on these requirements, such factors characterize the features of testing and elements of the software product that must be tested. Thus, testing of localization results can include both individual game elements and the game as a whole. In addition, if additional resources are available, additional stages of localization testing and text verification (so-called proofreading) can be carried out. Depending on the availability of human resources, testing of localization results can be carried out by the localization team, customer employees, or third-party contractors.

The group of factors "Permission to use machine translation" can be determined depending on the requirements set by the customer. Using machine translation can significantly speed up the translation process, but it increases the number of errors in it, because machine translation tools cannot yet understand the context, unlike humans. Thus, a text translated by machine translation will require additional attention during editing. Because of this, most often, machine translation is used by translators to speed up the translation in the following form: a quick translation of a separate TU is performed by machine translation, immediately after which

the TU is edited by the translator in order to correct errors, agree on proper names, take into account the character limit, etc.

The group of factors "Glossary availability" characterizes a document containing a list of terms and proper names that occur in the game, their transcription and explanations. The glossary should be translated before the localization files are translated. The glossary can be formed as follows:

- provided by the localization customer, since it is enough to form it only once in the original language, after which it can be provided to all localization teams;
- formed (developed) by the localization team "from scratch", which will require additional time.

The class of factors caused by volume and content restrictions is proposed to include a set of restrictions that are linguistic in nature and depend on the content that needs to be translated. This set is proposed to be divided into the following groups of factors:

- "Volume of data for localization";
- "Text complexity";
- "Translators' familiarity with the subject area";
- "Experience of translators";
- "Completeness of translation memory";
- "Unit substitution and formatting".

The group of factors "Volume of data for localization" consists of one indicator with the same name. This indicator is the main indicator that characterizes the required amount of work that needs to be done. On its basis, it is proposed to assess the time that will be spent on the implementation of a video game localization project. Most often, this indicator is measured in the number of words, but can also be measured in the number of TUs or characters.

The group of factors "Text complexity" shows how specialized the text being translated is. If the text contains a lot of technical terms or specialized vocabulary, localization may require finding or creating equivalents for these terms in the target language. For this, special industry dictionaries can be used or specialists working in a related field can be involved for additional consultations.

In video games, most often, the complexity of the text is associated with the literary features of the genre. Games set in the Middle Ages are characterized by the use of archaisms, historicisms, and archaic languages. Games set in fictional worlds are characterized by references to the original sources of the genre (if any) and the use of the appropriate terminology base.

The group of factors "Translators' familiarity with the subject area" in the case of video games characterizes:

- familiarity with the game itself or with the genres to which the game belongs;

- level of understanding of the context and plot of the game, which affects the understanding of the terminology and context of the terms used in the game;
- assessment of different translation options;
- course and result of choosing the translation option that best matches the original, helping to preserve the authenticity and atmosphere of the original.

Some games use special game systems (for example, such as *Dungeon & Dragons*). In such cases, it is necessary to coordinate the terminology with the localized original source (if the latter is available).

The group of factors "Translators' experience" characterizes the impact on the technical understanding of localization aspects and the speed of decision-making by the translator. Experienced translators who have knowledge of the gaming industry, its terminology, trends and features can better understand the context of the game and provide higher-quality localization. At the same time, the ability to work with CAT tools and effective management of their own resources can speed up the localization process. The group of factors "Completeness of translation memory" characterizes the impact on localization of previously acquired experience. Translation memory is a database containing a set of previously translated texts. It allows the user of the CAT tool to substitute a translation from memory when the current translation unit matches a previously translated one and, if necessary, refine it. The fullness of the translation memory significantly speeds up the localization process. Translation memory is filled by translators during the translation process of the game, but it can be transferred from previous localization projects that were carried out by the team. Such a transfer is useful for accelerating the translation of terms that are common to many games, but it is even more useful when transferring between games belonging to the same genre, since the terminological bases of such games overlap.

The group of factors "Replacement and formatting of units of measurement" characterizes the possibility or necessity of switching from one system of measurements to another (for example, from the metric system to systems of measurements based on the English system and vice versa). Such a transition includes formatting of measures of weight, length, as well as monetary units, date and time format, etc. Such formatting may include making changes to the text that is directly related to the units of measurement.

The proposed system for classifying factors that affect the progress and results of assessing the complexity of an IT project for video game localization allowed to establish the main factors that should be taken into account when developing a method for assessing the complexity of an IT project for video game localization.

4.4.2 Results of determining the content of the stages and activities of the method for assessing the complexity of an IT project for video game localization

The developed method for assessing the complexity of an IT project for video game localization is proposed to be presented as a sequence of the following stages:

- Stage 1: preliminary data preparation;
- Stage 2: assessment of the complexity of translating TUs contained in the localization files;
- Stage 3: obtaining the predicted duration of translating localization files;
- Stage 4: calculation of the total predicted duration of the project).

In accordance with the provisions of PMBOK [24], each stage of the developed method was divided into separate activities as follows:

Stage 1:

- Activity 1 "Preliminary preparation of data for further processing".

Stage 2:

- Activity 2 "Determination of TU similarity";
- Activity 3 "Assessing the complexity of the TU translation and calculating the complexity of the localization file".

Stage 3:

- Activity 4 "Calculating the baseline predicted translation speed of the localization file";
- Activity 5 "Calculating the baseline predicted translation duration of the localization file";
- Activity 6 "Determining the ITSC";
- Activity 7 "Calculating the individual predicted translation speed of the localization file";
- Activity 8 "Calculating the individual predicted translation duration of the localization files".

Stage 4: "Calculating the overall predicted duration of the project":

- Activity 9 "Calculating the overall predicted duration of the translation work";
- Activity 10 "Calculating the overall predicted duration of the editing";
- Activity 11 "Calculating the predicted duration of the IT project";
- Activity 12 "Calculating the average translation speed".

The activities are defined as follows:

1. Activity 1 "Preparation of data for further processing" consists in analyzing the input data and bringing it into a form convenient for further processing.

The input data of Activity 1 are provided in the form of localization files received by the technical specialist from the customer or from the program itself.

Pre-preparation of data involves structuring the TU (if this has not been done) and bringing the localization files to the sizes determined by the experts (members of the localization team) convenient for further processing.

The result of Activity 1 is structured localization files prepared for further processing.

2. Activity 2 "Determination of TU similarity" consists in determining the similarity of sets of words included in different TUs using the Jaccard measure.

The input data set for Activity 2 is the result of Activity 1.

The Jaccard measure (coefficient) $J(A,B)$ is calculated using the formula

$$J(A,B) = \frac{A \cap B}{A \cup B}, \quad (4.2)$$

where A – the set of words included in the first TU being compared; B – the set of words included in the second TU being compared.

The second TU is considered similar to the first if the Jaccard coefficient value exceeds 0.69. Then the same score will be applied to the second TU as for the first. This Jaccard coefficient value was obtained as a result of a study of the Crowdin CAT system settings [20].

The result of Activity 2 is a set of TUs that require further assessment.

3. Activity 3 "TU translation complexity assessment and localization file complexity calculation" consists in assessing the translation complexity of each unique TU included in the localization file. Based on the assessment results, the file complexity score and the predicted translation time are calculated.

The input data set for Activity 3 is the results of Activity 2.

The assessment of the complexity of the TU translation is proposed to be carried out by an expert on a nine-point scale. Possible assessments of complexity on this scale and their explanations are given in **Table 4.3**.

The localization file complexity assessment $grade_f$ is proposed to be presented as a weighted average assessment of all elements included in the file and determined by the formula

$$grade_f = \frac{\sum_{i=1}^m (grade_i \cdot n_i)}{\sum_{i=1}^m n_i}, \quad (4.3)$$

where $grade_i$ – the expert assessment of the complexity of the i -th TU; n_i – the size of the i -th TU (number of words); m – the number of TUs in the localization file.

The result of the execution of Activity 3 is a set of localization file complexity assessments.

Table 4.3 Developed scale for assessing the complexity of the translation of translation units

Difficulty group	Common features	Expert assessment	Excellent signs
TUs of low complexity	Translation of the TU does not require searching for additional information	1	Variables do not complicate the understanding of the TU, there is no need to adapt the text
		2	Variables make it somewhat difficult to understand the TU, there is no need to adapt the text
		3	Variables make it somewhat difficult to understand the TU, there is a need to adapt the text
TUs of medium complexity	In the TU, there are terms that require clarification, or non-standard lexical and grammatical constructions	4	Variables make it somewhat difficult to understand the TU, there is no need to adapt the text
		5	Variables to some extent complicate the understanding of the TU, there is no need to adapt the text
		6	Variables greatly complicate the understanding of the TU, there is a need to adapt the text
TUs of high complexity	In the TU, there are terms that need to be clarified, and non-standard lexical and grammatical constructions	7	Variables make it somewhat difficult to understand the TU, there is no need to adapt the text
		8	Variables to some extent complicate the understanding of the TU, there is no need to adapt the text
		9	Variables greatly complicate the understanding of the TU, there is a need to adapt the text

4. Activity 4 "Calculation of the basic predicted translation speed of the localization file" consists in calculating the basic predicted translation speed of the localization file v_f .

The input data set for Activity 4 is the results of Activity 3 and data on the size (number of words) of the assessed localization files.

According to [18], a translator can process 500 words per hour in the case of the simplest texts and 200 words per hour in the case of the most complex texts. For texts of average complexity, this indicator is 300 words per hour [16–19].

To establish the dependence of translation speed on the assessment of the complexity of the translation of the text, it was proposed to build a special model.

According to the developed scale for assessment (**Table 4.3**), it was established that the expert assessment of translation complexity 1 corresponds to a speed of 500 words per hour, the assessment of 5–300 words per hour, and the assessment of 9–200 words per hour. Thus, when the value of the function argument is 1, the function value is 500, when the value of the function argument is 5 – the function value is 300, and when the value of the function argument is 9 – the function value is 200.

Given the nonlinear nature of the relationship between the assessment of complexity and translation speed, the method of function approximation was used and it was established that this function is a branch of a hyperbola, which is located in the first quarter of the Cartesian coordinate system. Since the difficulty scores range from 1 to 9, the graph of the function is limited to the range $[1, \dots, 9]$ along the x -axis. Based on these data, a system of equations was developed to determine the parameters of the hyperbola passing through the specified points. This system of equations has the form:

$$\begin{cases} 500 = \frac{a}{1-h} + k, \\ 300 = \frac{a}{5-h} + k, \\ 200 = \frac{a}{9-h} + k. \end{cases} \quad (4.4)$$

The system of equations (4.4) has the following solution: $a = 4800$, $h = -7$, $k = -1000$. Therefore, the basic predicted translation speed of the localization file v_f is proposed to be determined by the formula

$$v_f = \left(\frac{4800}{\text{grade}_f + 7} \right) - 100. \quad (4.5)$$

The result of Activity 4 is the set of calculated basic predicted translation speeds of the localization files.

5. Activity 5 "Calculation of the basic predicted translation duration of the localization file" consists in calculating the basic predicted translation duration of the localization file t_f .

The input data for Activity 5 are the result of Activity 4 and the size (number of words) of the localization files.

The calculation of the basic predicted translation duration t_f is carried out by the formula

$$t_f = \frac{n_f}{V_f}, \quad (4.6)$$

where n_f – the number of words in the localization file.

The result of Activity 5 is the set of calculated basic predicted translation durations of the localization files.

6. Activity 6 "Determining the ITSC" consists of comparing the predicted translation duration value with the actual translation duration value of the data sample, which will allow to adjust further estimates of the translation duration of the localization files.

The input data for Activity 6 are sets of values of the actual and predicted translation duration of the test data sets.

The ITSC for a localization file $l_{f(x,y)}$ is defined as the ratio

$$l_{f(x,y)} = \frac{t_{f_{est}}}{t_{f_{real}}}, \quad (4.7)$$

where $t_{f_{est}}$ – the predicted translation duration of the localization file; $t_{f_{real}}$ – the actual translation duration of the localization file; x – the translator performing the translation; y – the expert performing the assessment.

The ITSC for pairs "translator-expert" $l_{(x,y)}$ is calculated by the formula

$$l_{(x,y)} = \frac{\sum_{i=1}^t l_{f_{(x,y)}}}{t}, \quad (4.8)$$

where $l_{f_{(x,y)}}$ – the ITSC value of the localization file obtained for the i -th pair of "translator-expert"; t – the number of localization files included in the test data set.

This calculation method is due to the fact that in different projects for the same pair of "translator-expert" who assess the TU complexity, knowledge in the subject area may differ. In addition, it reduces the amount of additional work performed, since the test sample is already part of the localization files of the project.

The result of Activity 6 is a set of calculated ITSC for each pair of "translator-expert".

7. Activity 7 "Calculation of individual predicted translation speed of a localization file" consists in calculating individual coefficients of predicted translation speeds of localization files.

The input data for Activity 7 are the results of Activity 4 and Activity 6.

The individual predicted translation speed v_f^* is determined by the formula

$$v_f^* = I_{test} \cdot V_f, \quad (4.9)$$

where I_{test} – the ITSC of the test data set.

The output of Activity 7 is a set of calculated individual predicted translation speeds for localization files.

8. Activity 8 "Calculation of individual predicted translation duration for localization files" consists in calculating the values of individual predicted translation durations for localization files, taking into account the difference in domain knowledge and experience of the translator and the expert who assesses the complexity of the translation.

The input data for Activity 8 are the sizes (number of words) of localization files and the results of Activity 7.

The individual predicted translation duration t_f^* is calculated by the formula

$$t_f^* = \frac{n_f}{v_f}, \quad (4.10)$$

where n_f – the number of words in the localization file.

The output of Activity 8 is a set of values for the calculated individual predicted translation times for localization files.

9. Activity 9 "Calculate the total predicted translation time" consists of calculating the sum of the translation times for the localization files into which the input data set was divided.

The input data for this activity are the glossary translation time and the results of Activity 8.

The total predicted translation time $t_{translation}$ is determined by the formula

$$t_{translation} = t_{glossary} + \sum_{j=1}^q \sum_{i=1}^b t_{f_{ij}}^*, \quad (4.11)$$

where $t_{glossary}$ – duration of glossary translation; $t_{f_{ij}}^*$ – duration of translation of one i -th localization file in the j -th category; b – number of localization files into which the j -th category is divided; q – number of categories into which the localization files are divided.

The result of Activity 9 is the calculated total predicted duration of work on the translation of the text provided for by the IT project for video game localization.

10. Activity 10 "Calculation of the total predicted duration of editing" consists in calculating the total predicted duration of editing in the IT project for video game localization.

The input data for Activity 10 is the result of Activity 9.

The total predicted duration of editing is determined by the formula [16–18]

$$t_{\text{editing}} = 0.5 \cdot t_{\text{translation}} \quad (4.12)$$

The result of Activity 10 is the total predicted duration of work on editing all the text provided for by the IT project for video game localization.

11. Activity 11 "Calculation of the projected duration of the IT project" consists of calculating the projected duration of the IT project for video game localization t_{total} .

The input data for Activity 11 are the results of Activities 9 and Activity 10, as well as the time allocated by the customer for testing the game localization.

The projected duration of the IT project for video game localization t_{total} is determined by the formula

$$t_{\text{total}} = t_{\text{translation}} + t_{\text{editing}} + t_{\text{testing}}, \quad (4.13)$$

where t_{testing} – the duration of localization testing.

The duration of localization testing is determined by the resources allocated by the customer and can be assessed by the localization team, development staff, or an external contractor.

The output of Activity 11 is the calculated projected duration of the entire IT project for video game localization.

12. Activity 12 "Calculation of the average translation speed" consists of calculating the average translation speed within the entire game localization project v_{avg} . The average translation speed in an IT project for video game localization can be used for further application in a word-based method to calculate the duration of other localization projects.

The input data for Activity 12 is the total volume of translated text and its translation duration.

The average translation speed v_{avg} is calculated using the formula

$$v_{\text{avg}} = \frac{n_{\text{total}}}{t_{\text{translation}}}, \quad (4.14)$$

where n_{total} – the number of words in the video game text for translation.

The result of Activity 12 is the value of the average translation speed within the IT project for video game localization.

The developed method takes into account the following factors that affect the complexity of video game localization work:

- "Localization file structure";
- "Variable organization method";
- "Language pair";
- "Translation quality requirements";
- "Volume of data for localization";
- "Text complexity";
- "Translators' familiarity with the subject area";
- "Translators' experience".

4.5 Experimental verification of the obtained results

Based on the proposals of **Section 4.3**, the experimental verification was divided into two stages: main and additional.

The IT project for video game localization "Don't Starve Together" [25] was selected for the main stage. At the same time, as indicated in **Section 4.3**, it is proposed to use the following indicators:

- absolute localization duration;
- relative error of localization duration prediction;
- relative average error of localization file translation duration.

The localization files for both stages were obtained directly from the game files. Thus, in the game "Don't Starve Together", each available language corresponds to a single localization file containing all TUs in this language.

An example of a TU from a localization file in the original language is shown in **Fig. 4.4**.

The TU shown in **Fig. 4.4** consists of several parts that have a certain meaning:

- keys (lines starting with "#."), which are unique identifiers needed to substitute values into the game interface;
- the TU context (lines starting with "msgctxt") indicates the context (application element) to which the string belongs;
- the original string (lines starting with "msgid") in English, which needs to be translated;
- the translation string (lines starting with "msgstr"), into which the text in the target language is placed.

```
#. STRINGS.ACTIONS.ABANDON  
msgctxt "STRINGS.ACTIONS.ABANDON"  
msgid "Abandon"  
msgstr ""
```

Fig. 4.4 Example of a Translation Unit in the original language from the "Don't Starve Together" game

In the case under consideration, the values of the keys coincide with the values of the context. This limits the context provided, but it is sufficient to understand which element of the game is being translated. The approach, in which the translation does not replace the original text, but duplicates it, increases the size of the localization file, but increases reliability (if the translation is missing, the value in the original language will be substituted).

During the execution of Activity 1, it was established that in the selected game, each available language corresponds to one localization file, which contains all TUs in this language. This file contains 72049 TUs with a total length of 515500 words. TUs contain meaningful key names, so they were automatically distributed among 113 localization files, each of which represents a certain subcategory. It was decided to reduce the amount of data for testing the developed method. 29 localization files were selected and it was decided to do only the translation without editing and testing. These files contain 2348 TUs with a total length of 18348 words (3.5% of the total number of words in the game). The localization files did not contain a separate glossary, so it was generated during the translation of the localization files using CAT tools.

During the execution of Activity 2, it was found that among the 2348 TUs there are 2200 unique TUs with a total length of 17815 words. The Jaccard coefficient (4.2) was used to determine the similarity of TUs.

Examples of the results of Activity 1 and Activity 2 are given in **Table 4.4**.

During Activity 3, an expert assessment of the translation complexity of TUs included in the localization files was conducted.

During the experimental test, the speed of expert assessment was measured. This speed ranges from 80 words per minute to 110 words per minute. This indicator may vary depending on the level of familiarity of the expert with the subject area. During the work on assessing the translation complexity of TUs, the expert's familiarity with the subject area increases. It is expected that the assessment speed will increase with the increase in the level of familiarity of the expert with the subject area.

As a result of Activity 3, according to formula (4.3), weighted average estimates of the complexity of the specified localization files were obtained.

Table 4.4 Characteristics of localization files

Name of file	Quantity of TUs	Quantity of words	Average size of TU
boarlord	65	377	5.8
carnival_crowkid	75	555	7.4
carnival_host	35	351	11.2
character_aboutme	19	317	16.68
character_descriptions	22	380	17.27
Name of file	Quantity of unique TUs	Quantity of words in unique TUs	
boarlord	64	375	
carnival_crowkid	74	551	
carnival_host	34	337	
character_aboutme	19	317	
character_descriptions	22	380	

During Activity 4, according to formula (4.4), the basic predicted translation speeds of localization files were determined.

During Activity 5, the baseline predicted translation time for each of the considered localization files was determined according to formula (4.5).

During Activity 6, the translation time for a test sample of 5 localization files with a total length of 1980 words was generated, translated, and measured. A comparison of the baseline predicted and actual translation times for these files is given in **Table 4.5**.

Table 4.5 Localization file characteristics

Name of file	Quantity of words	Basic predicted translation duration, min	Actual translation duration, min
boarlord	377	61	40
carnival_crowkid	555	92	47
carnival_host	351	55	36
character_aboutme	317	56	37
character_descriptions	380	72	49

After comparing the predicted and actual translation times of the localization files, the ITSC for long strings (2.5 words and more) was set to $I_{(xy)} = 1.5$. Given that files with a short average string length (less than 2.5 words) account for 806 words

out of a total of 17.815 words, it was decided not to calculate the ITSC for short strings using a separate test sample. Experimentally, during translation, the ITSC for short strings was found to be $I_{(x,y)} = 0.8$.

During Activities 7 and 8, individual predicted translation speeds and times were calculated and adjusted for a specific translator. In this study, one person acted as both the expert and the translator during the experimental validation. Therefore, the ITSC obtained as a result of Activity 6 is higher than that which can be obtained in a real project. The reason for this is that at the beginning of the translation work, the translator, who is also an expert, is already familiar with the text.

Examples of the results of Activities 4–8 are given in **Table 4.6**. In addition to the basic predicted translation speed and duration and the individual predicted translation speed and duration, which are calculated using the developed method, **Table 4.6** contains the actual indicators of translation speed and duration.

The individual predicted translation speed, according to [18], ranges from 200–500 words per hour. It was experimentally established that the real translation speed is higher than that suggested by the source [18]. The reasons for this may be:

- experience of translators;
- familiarity of translators with the subject area;
- peculiarities of video game translation as a type of software;
- peculiarities of the text being translated.

Table 4.6 Predicted and actual translation speed and duration

Name of file	Basic predicted speed of translation, words/min	Basic predicted translation duration, min	Individual predicted speed of translation, words/min
boarlord	367	61	550.5
carnival_crowkid	358	92	537
carnival_host	343	55	514.5
character_aboutme	319	56	478.5
character_descriptions	300	72	450

Name of file	Individual predicted translation duration, min	Real translation speed, words/min	Real translation duration, min
boarlord	41.09	565.5	40
carnival_crowkid	62.01	708.51	47
carnival_host	40.93	585	36
character_aboutme	39.75	514.05	37
character_descriptions	50.67	465.31	49

During the execution of Activity 9, the translation duration in the video game localization project was calculated. Since the glossary was generated and translated during the translation of the localization files, the translation time of the glossary was included in the translation time of the localization files. The predicted translation duration, which was calculated using formula (4.10), was 32 hours 57 minutes.

The video game localization IT project was limited to the translation of the localization files only. As a result, the editing time for Activity 10 is zero.

The testing time is determined by the customer, in this experiment it is also zero. Therefore, the predicted duration of the IT project for video game localization, calculated during the execution of Activity 11, is equal to the translation time, namely 32 hours 57 minutes.

The actual translation duration was 26 hours 26 minutes. Thus, the predicted translation duration exceeded the actual duration by 24.7%. The average predicted translation speed for the project, which was calculated during the execution of Activity 12, was 556.6 words per hour.

During the additional stage of the experimental verification, the effectiveness of using the ITSC obtained during the main stage in another video game localization project was tested.

The game "Sid Meier's Civilization® VI" was selected for localization and the ITSC obtained when using the developed method in the IT project for the localization of the game "Don't Starve Together" was taken. Games belonging to different genres were chosen for comparison in order to minimize the overlap of glossaries.

Unlike the video game "Don't Starve Together", the localization files of the game "Sid Meier's Civilization® VI" contain only a text string and a key, which is used to perform text substitution in the application.

When translating a localization file, the text in the original language is replaced with the text in the target language.

The localization files considered in the additional stage of experimental verification were obtained directly from the game in its original form. They were not changed during data preprocessing because the file organization proposed by the developer meets the requirements of the developed method. The number of words in the localization files was 6287.

The characteristics of the localization files are given in **Table 4.7**.

As in the main stage, the complexity of the selected localization files was assessed. As a result of the assessment, the basic predicted translation speed and duration were determined.

Table 4.7 Characteristics of the localization files of the video game "Sid Meier's Civilization® VI"

File name	Number of translation units	Number of words	Average size of translation units
Buildings_Text	59	1429	24.2
Combat_Text	75	703	9.4
CityStatePicker_FrontEndText	93	1646	17.7
DiplomacyPanel_Text	75	425	5.7
DiplomacyNotifications_Text	66	313	4.7
DiplomacyModifiers_Text	120	884	7.4
DiplomacyDeals_Text	136	887	6.5

ITSC $I_{(x,y)}$ was set to 1.5. This value was obtained at the main stage of experimental verification. Since the average TU lengths in the considered localization files are long, the corresponding ITSC value was selected.

Based on the obtained indicators of the basic predicted translation duration and ITSC, individual predicted translation speeds and durations were calculated.

The results of comparing the predicted values of translation speeds and durations with the real values for these localization files are given in **Table 4.8**.

Table 4.8 Results of comparing the predicted and real values of translation speeds and durations (additional stage)

File name	Basic predicted speed of translation, words/min	Basic predicted translation duration, min	Individual predicted speed of translation, words/min	Individual predicted translation duration, min	Real translation speed, words/min	Real translation duration, min
Buildings_Text	376	228	564	152.02	489.94	175
Combat_Text	341	123	511.5	82.46	602.57	70
CityStatePicker_Front-EndText	361	273	541.5	182.38	637.16	155
DiplomacyPanel_Text	338	75	507	50.30	520.41	49
DiplomacyNotifications_Text	415	45	622.5	30.17	722.31	26
DiplomacyModifiers_Text	408	130	612	86.67	964.36	55
DiplomacyDeals_Text	409	128	613.5	86.75	782.65	68

The predicted translation duration was 11 hours 11 minutes. The actual translation duration was 9 hours 57 minutes.

The predicted translation duration for the video game "Sid Meier's Civilization® VI", calculated using the ITSC obtained during the localization of the video game "Don't Starve Together", differs from the actual one by 12.1%. This deviation is less than that obtained for the game "Don't Starve Together" (24.7%).

Transferring ITSC from one IT localization project to another showed higher efficiency than in the original IT localization project for which the coefficient was calculated. This may be due to the following reasons:

- the peculiarities of working with the text of a particular game have become apparent;
- the sample on which the study was conducted is not large enough.

When loading localization files into the CAT tool and further processing them, it was found that the Crowdin CAT tool splits TUs stored in the ".xml" format (which contains the localization files for the game "Sid Meier's Civilization® VI") by sentences. If a translation unit consists of two sentences, then it is represented by the CAT tool as two separate TUs.

Splitting TUs by sentences speeds up the translation process of localization files. In the case when the level of similarity of TUs is low, the similarity of the sentences included in them may be greater. Such similarity expands the possibilities of using translation memory. However, such a division of TUs makes it difficult for the translator to understand the context, as he/she has to check the agreement of sentence parts not in a single TU containing several sentences, but in several TUs, each of which contains only one sentence.

An alternative reason for obtaining such results may be coincidence caused by an insufficient amount of test data.

In the experiments considered, the ITSC value is 1.5 for localization files with long lines.

In both experiments, this phenomenon requires further research with an increase in the number of games on which the study is conducted and the amount of text data.

4.6 Discussion of the results of the development of a method for assessing the complexity of an IT project for video game localization

The existing methodology for assessing the duration of an IT project for video game localization, based on the number of words, did not take into account the influence of a significant number of factors on this project. Therefore, a study was

conducted to identify and classify the main factors that affect the duration of such IT projects. The result of this study is a proposed classification system for the main factors that affect the IT project for video game localization, and the definition of the main groups of factors that make up the selected classes of factors.

Taking into account the proposed classification system for the main factors, a method for assessing the complexity of an IT project for video game localization was developed. The developed method consists of four stages, the implementation of which will allow calculating the predicted duration of an IT project for video game localization.

Unlike the existing methodology, the developed method takes into account the factors that affect the complexity of video game localization work, namely:

- "Localization file structure";
- "Method of organizing variables";
- "Language pair";
- "Translation quality requirements";
- "Volume of data for localization";
- "Text complexity";
- "Translators' familiarity with the subject area";
- "Translators' experience".

During the experimental testing, the developed method was compared with the existing method in the following options:

a) in the existing method, the total number of words was selected as the number of words, taking into account repeated TUs and variables, and as the translation speed, the speed recommended by sources [17–19] (300 words per hour);

b) in the existing method, the total number of words was selected as the number of words, taking into account repeated TUs and variables, and as the translation speed, the average translation speed obtained within the same project.

Comparison according to option (a) allows to verify or refute the statement that the developed method is more effective than the existing method.

Comparison according to option (b) allows to investigate the possibility of using the calculated average translation speed obtained as a result of using the method in further calculations of forecasting the duration of operations. In this case, the developed method can be used at the beginning of an IT project, when there is insufficient data to calculate the duration of translation. Then, after translating a certain number of localization files, the average translation speed obtained within the framework of a video game localization project can be used to predict the duration of translation of localization files. Further forecasting the duration of operations in an IT localization project based on the average translation speed will simplify the calculation of

the predicted duration of translation and will eliminate the need to use resources to assess the complexity of the TU translation.

For comparison, it is proposed to use the relative average error of the predicted translation duration [26]. Localization files have different sizes, so a relative, not an absolute value, was chosen for comparison.

For graphical visualization of the comparison results, it is proposed to use the relative error of the translation duration prediction and the square of the relative error of the translation duration prediction.

The absolute error of the translation duration of the localization file Δt is calculated by the formula

$$\Delta t = t_{est} - t_{real} \quad (4.15)$$

where t_{est} – the predicted file translation duration; t_{real} – the actual file translation duration.

The relative error of the localization file translation duration δt is calculated by the formula

$$\delta t = \frac{\Delta t}{t_{real}}. \quad (4.16)$$

The relative average error of the translation duration $\bar{\mu}$ is calculated by the formula

$$\bar{\mu} = \frac{1}{q} \sum_{i=1}^q \delta t_i, \quad (4.17)$$

where δt_i – the relative error of the translation duration of the i -th localization file; q – the number of localization files.

The results of the comparison of the errors of the developed method and existing methods for the localization files mentioned in **Tables 4.4–4.6** are given in **Table 4.9**.

The results of the graphical comparison of relative errors and their squares for both options are presented in **Fig. 4.5** and **Fig. 4.6**, respectively. These figures indicate: orange – data from a real IT project; blue – data calculated using the developed method; cyan – data calculated using the existing methodology according to option (a); green – data calculated using the existing methodology according to option (b).

The obtained data shows that the error in forecasting time costs calculated using the developed method is the smallest. In some cases, the results obtained using the developed method have a relative error of less than 10% (localization files actions_1, boarlord, character_aboutme, etc.). The accuracy of the developed

method is especially evident in the conditions of using localization files that contain short TUs.

Table 4.9 Results of the comparison of the errors of the developed method and existing methods for examples of localization files

File name	Word count method (basic)		
	Δt	δt	$\delta^2 t$
actions_1	-3.80	-0.1407	0.0198
boarlord	35.4	0.8850	0.7832
carnival_crowkid	64.00	1.3617	1.8542
carnival_host	34.20	0.9500	0.9025
character_aboutme	26.40	0.7135	0.5091
character_bios	293.80	1.2293	1.5112
character_descriptions	27.00	0.5510	0.3036

File name	Developed method		
	Δt	δt	$\delta^2 t$
1	5	6	7
actions_1	0.02	0.0007	0.0000
boarlord	1.09	0.0272	0.0007
carnival_crowkid	15.01	0.3194	0.1020
carnival_host	4.93	0.1370	0.0188
character_aboutme	2.75	0.0743	0.0055
character_bios	69.87	0.2923	0.0855
character_descriptions	1.67	0.0340	0.0012

File name	Methodology based on word count (average translation speed)		
	Δt	δt	$\delta^2 t$
actions_1	-13.8182	-0.5118	0.2619
boarlord	2.8409	0.0710	0.0050
carnival_crowkid	16.0682	0.3419	0.1169
carnival_host	3.8864	0.1080	0.0117
character_aboutme	-0.9773	-0.0264	0.0007
character_bios	63.7273	0.2666	0.0711
character_descriptions	-5.8182	-0.1187	0.0141

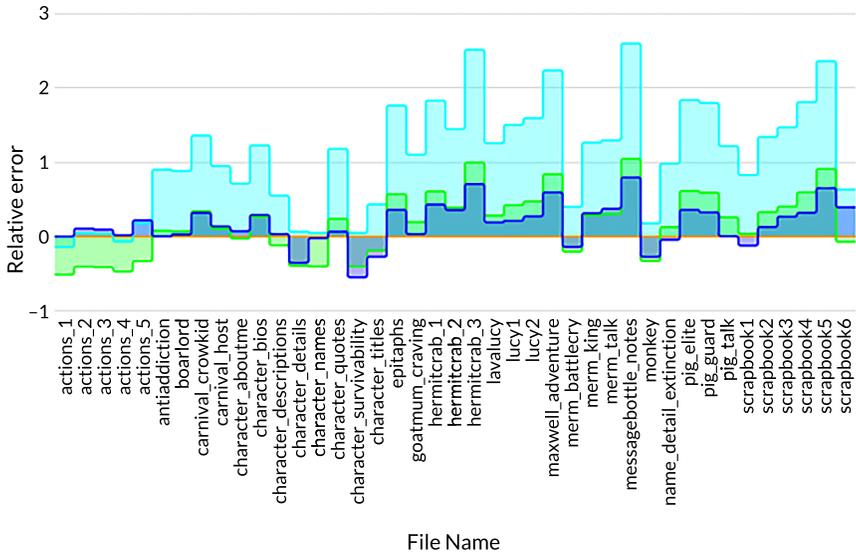


Fig. 4.5 Results of comparison of relative errors

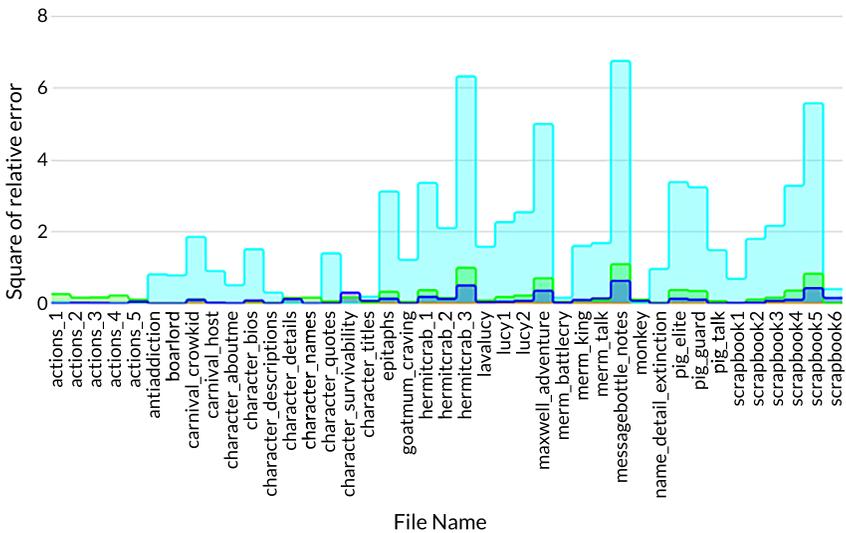


Fig. 4.6 Results of comparison of squares of relative errors

The results also showed that it is recommended to calculate the average translation speed for localization files with long and short TUs separately. This is due to the fact that the translation speed of short TUs is lower than the translation speed of long ones, and in video games, most TUs describe the game's scenario elements.

The total predicted translation duration was:

- using the developed method – 32 hours 57 minutes;
- using the existing methodology according to option (a) – 61 hours 9 minutes;
- using the existing methodology according to option (b) – 34 hours 44 minutes.

The calculated total relative average error was:

- using the developed method – 0.1639;
- using the existing methodology according to option (a) – 1.0661;
- using the existing methodology according to option (b) – 0.1739.

The actual translation duration was 26 hours 26 minutes.

The results of the predicted durations of the IT project for video game localization were compared with the actual duration of the project. The relative error of the forecast duration of the IT project for video game localization μ is calculated by the formula

$$\mu = \frac{t_{total_{est}} - t_{total_{real}}}{t_{total_{real}}}, \quad (4.18)$$

where $t_{total_{est}}$ – the predicted duration of the localization file translation; $t_{total_{real}}$ – the actual duration of the localization file translation.

The results of comparing the accuracy of the forecasts of the duration of the IT project for video game localization, obtained by both comparison options, with the actual results of the project duration are given in **Table 4.10**.

Table 4.10 Results of comparing the accuracy of the forecasts of the duration of the IT project for video game localization and the actual duration of this IT project

Calculation method	Accuracy metrics for duration forecasting for an IT video game localization project		
	Absolute duration	Relative error of duration forecast	Relative average error of localization duration
Real translation duration	26 h 26 min	-	-
Developed method	32 h 57 min	0.24	0.1639
Existing methodology (option a))	61 h 9 min	1.31	1.0661
Existing methodology (option b))	34 h 44 min	0.31	0.1739

Comparison of the obtained results showed that, in comparison with existing methods, the developed method for assessing the complexity of an IT project for video game localization showed the highest accuracy.

The main limitation of the developed method is the impossibility of its application in the case when the developer has not performed the distribution of TUs by localization files and the TUs themselves do not have meaningful names, which makes it impossible to automate the distribution of TUs by the localization team. In this case, manual implementation of such distribution is possible, but such an approach is not advisable for the following reasons:

- complication of input data analysis;
- the need to involve an excessively large number of resources;
- potential ambiguity of TUs;
- impossibility of searching for TUs in the application.

The complication of input data analysis lies in the fact that without a clear structure of localization files or meaningful names, the process of determining categories and subcategories by which TUs should be distributed becomes difficult.

The need to involve an excessively large number of resources is expressed in the fact that each TU must be read by an expert and assigned to one of the specified categories or subcategories. This operation is impractical because it consumes too much time that could be used to perform other processes by a team member.

The potential ambiguity of a TU is expressed in the fact that a video game localization expert cannot always determine which subcategory a TU belongs to based solely on its content. In this case, it must assign the TU to a subcategory based on some assumptions.

The impossibility of finding a TU in the application is that if there are named TU keys, localization team members can quickly and accurately determine the location of this TU in the application, which is necessary for verification and testing.

In the case where there are no named keys and the structuring is done by the developer, localization team members can only make assumptions about the location of the TU in the application based on its content.

Another limitation of this method is that it is intended for third-level localization. The amount of content localized at the first and second levels of localization is limited to several thousand words, so the use of the developed method is redundant. The fourth level of localization includes both text localization and the work of the voice-over team, the assessment of which is not provided for in the method.

The main disadvantage of the considered method is the need to assess the complexity of each TU. Such an assessment requires additional time and human resources, but allows to increase the accuracy of forecasts of the duration of operations in an IT project for video game localization.

The use of machine learning and natural language processing methods for assessing the complexity of TU translation is considered as the main vector for further development of the method.

The use of such tools will allow to get rid of the main disadvantage of the developed method – the time spent by experts on assessing the complexity of TU translation. In addition, it is expected to reduce ITSC fluctuations for different pairs of "expert-translator", since the assessment will be carried out by the same expert.

4.7 Conclusions

A new method for assessing the complexity of video game localization IT projects has been developed, which takes into account a larger number of complexity factors than the existing method based on the number of words. The developed method takes into account not only the number of words, but also other factors that affect the duration of localization. It is effective even in the absence of retrospective data on the speed of translation, which is an important advantage for the initial phase of projects.

An experimental verification of the obtained result has been carried out. The course and results of the calculations confirm the possibility of using the developed method to assess the characteristics of video game localization IT projects. It has been proven that using the developed method allows to obtain estimates of the time spent on the implementation of an IT project from the third level of video game localization, which are more accurate than the estimates obtained using the existing method. The total predicted duration of the translation was:

- using the developed method – 32 hours 57 minutes;
- using the existing method according to option (a) – 61 hours 9 minutes;
- using the existing methodology according to option (b) – 34 hours 44 minutes.

The calculated total relative average error was:

- using the developed method – 0.1639;
- using the existing methodology according to option (a) – 1.0661;
- using the existing methodology according to option (b) – 0.1739.

The actual translation duration was 26 hours 26 minutes.

Conflict of interest statement

The authors declare that there is no conflict of interest in relation to this paper, as well as the published research results, including the financial aspects of conducting

the research, obtaining and using its results, as well as any non-financial personal relationships.

Use of artificial intelligence statement

The authors declare that they did not use artificial intelligence tools in preparing this manuscript.

References

1. Wolf, M. J. P. (Ed.) (2008). *The video game explosion: A history from PONG to Playstation and beyond*. Bloomsbury Academic, 380.
2. Pirrone, M., D'Ulizia, A. (2024). The Localization of Software and Video Games: Current State and Future Perspectives. *Information*, 15 (10), 648. <https://doi.org/10.3390/info15100648>
3. Bernal-Merino, M. Á. (2011). Breve historia de la localización de videojuegos. *Trans. Revista de Traductología*, 15, 11–17. <https://doi.org/10.24310/trans.2011.v0i15.3191>
4. Esselink, B.; de Vries, A.-S., O'Brien, S. (Eds.) (2000). *A Practical Guide to Localization*. John Benjamins Publishing Company. <https://doi.org/10.1075/liwd.4>
5. Fry, D., Lommel, A. (2003). *The Localization Industry Primer*. Localization Industry Standards Association (LISA), 52. Available at: <https://www.scribd.com/document/581372706/The-Localization-Industry-Primer>
6. Toftedahl, M. (2020). Localization and regional aspects of game production – a research overview. *Proceedings of the 14th IADIS International Conference on Interfaces and Human Computer Interaction 2020 and 13th IADIS International Conference on Game and Entertainment Technologies 2020*, 135–143. https://doi.org/10.33965/ihci_get2020_2020101017
7. O'Hagan, M., Mangiron, C. (2013). *Game localization: translating for the global digital entertainment industry*. John Benjamins Publishing. <https://doi.org/10.1075/btl.106>
8. Ressin, M., Abdelnour-Nocera, J., Smith, A. (2011). Lost in Agility? Approaching Software Localization in Agile Software Development. *Agile Processes in Software Engineering and Extreme Programming*. Springer, 320–321. https://doi.org/10.1007/978-3-642-20677-1_25

9. Zorrakin-Goikoetxea, I. (2023). Survey on Localization From the Development Perspective. *Games and Culture*, 19 (3), 391–415. <https://doi.org/10.1177/15554120231168201>
10. Toftedahl, M. (2025). Towards a deeper understanding of localization as a game development process: Game studio studies in Sweden and China. *Entertainment Computing*, 54, 100944. <https://doi.org/10.1016/j.entcom.2025.100944>
11. Skadinš, R., Pinnis, M., Vasiljevs, A., Skadina, I., Hudik, T. (2014). Application of machine translation in localization into low-resourced languages. *Proceedings of the 17th Annual Conference of the European Association for Machine Translation, EAMT 2014*. <https://doi.org/10.13140/2.1.5047.6168>
12. Wang, X., Chen, C., Xing, Z. (2019). Domain-specific machine translation with recurrent neural network for software localization. *Empirical Software Engineering*, 24 (6), 3514–3545. <https://doi.org/10.1007/s10664-019-09702-z>
13. Moreno García, L. D., Mangiron, C. (2024). Exploring the potential of GPT-4 as an interactive transcreation assistant in game localisation: A case study on the translation of Pokémon names. *Perspectives*, 1–18. <https://doi.org/10.1080/0907676x.2024.2378346>
14. Bilous, O. M. (2013). *Teoriia i tekhnolohiia perekladu*. Kirovohrad: RVV KDPU im. V. Vynnychenka, 200.
15. ISO/IEC/IEEE 12207:2017. *Systems and software engineering – Software life cycle processes (2017)*. IEEE. <https://doi.org/10.1109/ieeestd.2017.8100771>
16. O'Donnell J. (2022). How long will it take to localize my game? – Localization scheduling. IGDA. Available at: <https://igda.org/news-archive/how-long-will-it-take-to-localize-my-game-localization-scheduling/>
17. Tarasiewicz, J. (2024). How much does the software localization cost? ATL. Available at: <https://www.atltranslate.com/articles/how-much-does-software-localization-cost>
18. Realistic translation times using human translators. PACTRANZ. Available at: <https://www.pactranz.com/translation-times/#speed>
19. Giulia, C. (2017). How do I calculate a translation time frame. Trusted Translations. Available at: <https://www.trustedtranslations.com/blog/calculate-translations-time-frame>
20. Crowdin. Available at: <https://crowdin.com>
21. Phrase Strings. Phrase. Available at: <https://phrase.com/platform/strings/>
22. Vasylytsova, N., Osypchuk, D. (2024). Analiz pokaznykiv, shcho vplyvaiut na skladnist robot z lokalizatsii informatsiinykh system. *Radioelektronika ta Molod u XXI Stolitti*. Vol. 6: Konferentsiia "Informatsiini Intelektualni Systemy". *KhNURE*, 231–233. <https://doi.org/10.30837/iyf.iis.2024.231>

23. Chandler, H. M., Deming, S. O. M. (2011). Game Localization Handbook. Jones & Bartlett Publishers.
24. Nastanova do zvodu znan z upravlinnia proiektamy. Nastanova PMBOK (2021). Project Management Institute, Inc. Available at: https://learn.ztu.edu.ua/pluginfile.php/274061/mod_resource/content/1/PMBOK7_Ukr_ForPersonal-UseOnly.pdf
25. Don't Starve Together on Steam (2016). STEAM. https://store.steampowered.com/app/322330/Dont_Starve_Together/
26. Taylor, J. R. (1997). An Introduction to Error Analysis: The Study of Uncertainties in Physical Measurements. University Science Books, 327.

CHAPTER 5

The task of quantitative assessment of changes in the long-term IT project management system

Maksym Yevlanov
Nataliia Vasylytsova
Iryna Panforova
Anastasiia Popova

Abstract

The object of the study is the process of change management in an IT project.

The study addressed the problem of quantitative assessment of changes that arise during the implementation of a long-term IT project. Existing common methods of change management do not allow for quantitative assessment of the main parameters of changes. Modern research is mainly aimed at solving the problem of quantitative assessment and change management within the entire IT project. The issue of quantitative assessment of changes that arise in the process of work of individual teams of IT project performers remains practically unexplored.

In the course of the study, it was proposed to improve the existing method for quantitative assessment of changes, based on the Beckhard and Harris's model. To improve this method, it was decided to use a descriptor approach. An improved method was developed, which, unlike the existing one, allows for quantitative assessment of changes based on the values of descriptors. These descriptors are formed in the process of performing sprints of a long-term IT project by its performers.

Based on the improved method, elements of information technology for automated solution of the problem of quantitative assessment of changes in the management system of a long-term IT project have been developed. A description of the architecture of information technology has been proposed, its technological stack has been defined, and the results of the development of software elements have been presented.

Experimental verification of the obtained research results was carried out within the framework of the long-term IT project "Web Constructor". The verification was carried out on the results of the work of one team of project developers.

The calculation was based on data obtained during the team's execution of seven project sprints. The verification results showed that the predicted values of the indicators quite accurately coincide with the values of the time actually spent on implementing changes (0.97 and 0.81 for the sixth and seventh sprints of the IT project, respectively).

Overall, the application of the improved method improved the change implementation rates during the IT project implementation and improved the team's attitude to the change management process by 14.5%. This allows to consider the improved method and implemented IT elements as a usable tool for change management of individual teams of IT project performers.

Keywords

IT project, change, Beckhard and Harris's model, least squares method, Huber method, descriptor, sprint.

5.1 Introduction

The last decades are characterized by the widespread use of long-term projects, the implementation period of which is three or more years. Such projects are typical for various branches of the IT sphere: development and implementation of information technologies (IT), information systems (IS), IT infrastructure, scientific research in this field, etc. [1, 2].

Analysis of the characteristics of modern long-term IT projects allows to highlight their main features [2]:

- complexity of description, caused by a large number of functions, processes, elements, data and connections between them;
- the presence of a set of closely interconnected subsystems, each of which has its own local goals and objectives;
- the need to integrate existing and re-developed applications;
- heterogeneity of individual groups of developers in terms of the level of classification and established traditions of using tools;
- a constant flow of changes in modern IS.

A significant number of changes in the implementation of such projects occur at the operational level of tasks, which makes high-quality change management at this level necessary. For example, poor quality of time-to-market time-variance assessment of individual project operations can lead to project overruns, resource overstrain, and unforeseen delays in product launch. On the contrary, systematic time-to-market time-variance assessment allows project management to make informed

decisions regarding resource planning, task allocation, and priorities in the event of changes in project requirements or conditions. Implementing such an assessment as a separate functional task of the long-term IT project management system helps reduce the risks of project delays and ensures its successful completion within the established deadlines.

The modern point of view considers change management as one of the tasks of change management. The term "change management" should be understood as a comprehensive, cyclical, and structured approach to changing individuals, groups, and organizations from their current state to a future state with expected business benefits [2, 3]. This approach is now described in the form of a framework – conventions, principles and methods of change management in the field of program management, project portfolios and individual projects. But, unfortunately, the existing change management framework [3] and standards and body of knowledge on IT project management [2, 4] do not provide specific recommendations for the selection of change management methods and their implementation in IT project management systems. Therefore, conducting research on solving change management problems as elements of the IT project management system is relevant from a theoretical and applied point of view.

5.2 Analysis of the approach and modern methods of project change management

The general features of the implementation of processes, subprocesses and individual works of the existing change management framework are determined by the change management life cycle adopted in [3]. According to this life cycle, each of the change management subprocesses is proposed to be considered as a set of individual works, each of which is performed in the order established by the framework (but not necessarily strictly sequentially one after the other). The entire change management process and its subprocesses in [4] are proposed to be described using an iterative life cycle model, taking into account the possibility of constant occurrence of adaptive changes in response to changing circumstances. Such a representation of the change management life cycle allows minimizing the connections between individual subprocesses and works of the existing change management framework and considering them as separate independent objects of scientific research. At the technological level, this means the possibility of creating and using for automation of the corresponding subprocesses and works of individual IT, which interact with each other according to the service-oriented paradigm.

Since the existing regulatory documents do not provide recommendations for choosing a specific change management method in an IT project, it was decided to consider the features of the change management methods that are considered the most common. These methods include [5]:

- Prosci ADKAR model [6, 7];
- the Accelerating Implementation Methodology (AIM) method [8, 9];
- Beckhard and Harris's model [10];
- Bridges Transition model [11];
- Kotter's 8-Step Change model [12];
- Kübler-Ross model [13];
- a method based on the Kurt Lewin model [14].

ADKAR is an abbreviation that represents five key stages necessary for the successful implementation of changes: "Awareness", "Desire", "Knowledge", "Ability", "Reinforcement" [7]. The Prosci ADKAR model was created by J. Hyatt in the late 1990s. This method offers a deep approach to change management, focused on personnel. It is assumed that in the process of implementing changes, it is necessary to clearly explain to personnel the reasons for the changes and their importance, which contributes to the involvement of personnel in the process. Next, it is important to train each employee in the methods of implementing changes, which allows to show the level of their professional knowledge and skills in the process of introducing changes. The final stage involves consolidating the changes, ensuring the sustainability of innovations in the organization's activities [6].

The Prosci ADKAR method is part of a broader approach to change management developed by PROSCI. It is based on the method using the ADKAR model, but includes additional tools, methods and practices aimed at effectively implementing changes in the organization [7].

AIM is a powerful and disciplined method for managing organizational change, including transformational change, until the full return on investment. AIM can be applied to any type of initiative or project, but most organizations direct the main resources and energy to the technical and business process components [8, 9]. One of the advantages of this method is the ability to systematically analyze and anticipate possible difficulties in the change process, which allows organizations to prepare and respond to them more effectively. However, AIM can require significant resources and time spent on using the method, and can also become difficult in the event of adverse circumstances or unforeseen events during the change implementation process [9]. The method, based on the Beckhard and Harris's model, provides five stages of change management aimed at identifying the need for change, developing a strategy for its implementation, forming an action plan and identifying responsible

executors. The main advantage of the method, based on the Beckhard and Harris's model, is a systematic approach to the change process, which allows to structure it and manage it effectively. Praxie has developed software applications for change management using a method based on the Beckhard and Harris's model and provides training [10, 11].

The method, based on the Bridges Transition model, enables organizations and individuals to better understand the human and organizational aspects of change and manage them effectively. The main advantage of this method is its focus on internal transition, which can provide a deeper and more sustainable change in the organizational environment. However, it may be less effective when it is necessary to respond quickly to external and unpredictable changes, as it focuses on the internal aspect of the transition [11].

The method, based on the Kotter's 8-Step Change model, is designed to increase staff involvement in change management and ensure its acceptance by all employees. The method includes the following eight stages [12]: "Creating the emotional need for change", "Building a coalition", "Building a vision", "Communicating the vision", "Implementing actions", "Formation of short-term achievements", "Consolidation of achievements", "Implementation of changes into the culture". Skipping one of these stages can lead to problems in making changes and complicate the change process.

The main advantage of the method is its systematic approach to change management and emphasis on involving personnel in the process.

The following can be noted as disadvantages of the method based on the Kotter's 8-Step Change model:

- implementation of all eight stages of the method may require significant effort and time, especially in large organizations or with a large scale of changes;
- the method focuses mainly on the organizational aspects of changes, leaving aside individual emotions and needs of employees;
- the method does not always take into account external influences, such as economic or political factors, which may affect the success of changes.

The method, based on the Kübler-Ross model, describes the stages of personnel behavior change, including: personnel resistance to change, lack of awareness of the consequences of change, personnel adaptation to new working conditions, positive attitude of employees to change and their acceptance. This method is important for understanding and predicting personnel reactions to changes in the organization. However, it is worth considering that the reaction to change can be individual and does not always correspond to this sequence of stages. From a practical point of view, the method can be used to identify any obstacles in the early stages of change projects and develop appropriate strategies. The main advantages of the method

include the fact that it is simple but effective for managing organizational changes. The disadvantage of the method is that since all employees react at different speeds, they are at different stages marked on the change curve. This often leads to problems when planning changes [13]. The method, based on the model of Kurt Lewin and developed in the middle of the twentieth century, remains one of the most popular at the current stage of change management development. This method is notable for its systematicity and practicality, which makes it effective for implementing changes in various types of organizations. However, it can require significant resources and time to implement, and may be less effective in the case of complex organizational structures or indiscriminate application of change management methods [14].

To compare the considered methods of change management in projects, the study proposed basic criteria (indicators). The results of the comparison of change management methods by criteria are given in **Table 5.1** [5].

Table 5.1 Comparison of existing methods of change management in projects

Indicator	Method No. 1	Method No. 2	Method No. 3	Method No. 4	Method No. 5	Method No. 6	Method No. 7
Ease of use	-	-	+	-	-	+	+
Complexity of implementation	-	-	+	-	+	+	+
Time required for use	+	+	-	+	+	+	-
Availability of quantitative assessment of changes	-	-	-	-	-	-	-
Flexibility of the method	-	-	+	+	-	+	-
Use in rapid changes	-	+	+	-	-	-	-
Possibility of certification	+	+	-	+	+	-	-
Possibility of training	+	+	+	+	+	+	+
Availability of an informative website	+	+	+	-	+	-	-
Availability of free training	-	-	+	-	-	+	+
Availability of use in complex projects	+	+	+	-	+	-	-
Cost of training, \$	2000	1790	-	-	6000 (850)	-	-

Source: [5]

Table 5.1 uses the following notations [5]:

- method No. 1 – Prosci ADKAR model;
- method No. 2 – AIM method;
- method No. 3 – Beckhard and Harris's model;
- method No. 4 – Bridges Transition model;
- method No. 5 – Kotter's 8-Step Change model;
- method No. 6 – Kübler-Ross model;
- method No. 7 – method based on the Kurt Lewin model.

The main drawback of the considered change management methods is their lack of opportunities for quantitative assessment of changes at the level of individual project tasks. Therefore, an analysis of modern research aimed at eliminating this drawback was conducted.

A significant amount of modern work in the field of change management in projects and, in particular, in IT projects is based on the idea of change management to the tasks of general (organizational) project management as a whole. For example, in [15] a new hybrid model of the IT project life cycle is proposed, which combines predictive planning with iterative implementation. This model emphasizes defined requirements, short sprints and early feedback, focusing on the interaction of people and customers, while limiting changes during the project life cycle [15]. But this model does not take into account the features of long-term IT projects. In addition, this model, like the change management methods discussed above, is focused on solving the problems of project personnel management, and not on quantitative assessment and change management. In [16] mathematical models and methods of change management in megaprojects caused by integrative actions of stakeholders under complex external conditions are investigated. The term "megaprojects" in [16] defines large-scale investment programs with complex organizational structures that unite many stakeholders, the interaction of which leads to the redistribution of power and the creation of temporary control centers. To describe the management of megaprojects, vector-matrix models of a dynamic system with feedback on the results of changes were used. To identify recurring patterns of negative events, the method of event-based analysis was used. As a result of the study, in [16] a prototype of IS was proposed, which is based on:

- a mathematical model of change management in megaprojects;
- a methodology of neural network analysis based on the use of the large language model Qwen 2.5-Plus for processing text information and forming quantitative estimates;
- a software interface for uploading documents, automated data processing and visualization of results.

This prototype provides users with the ability to analyze stakeholder interactions, assess the intensity of change, and predict potential risks based on historical data [16].

The disadvantage of this IS prototype is its focus on processing textual information, through which stakeholders can express their own attitude to the megaproject. This means that quantitative assessments of changes are formed in this prototype on the basis of individual statements, which can be influenced by the economic or political environment of stakeholders. In addition, this prototype is also focused on managing changes and risks of the entire megaproject as a whole. Solving the problem of project evaluation by using neural networks is one of the current research directions. Thus, in [17] it was proposed to use a neural network of radial basis functions to predict project efficiency, which illustrates changes in efficiency levels during the phases of failures and project recovery. To train this network in [17], data from 64 completed construction projects were used. The results show that the discrepancy between the predicted and actual values of the stability of the assessed project is less than 10%. But this approach has quite significant disadvantages. Among these shortcomings, it is necessary to highlight, in particular: the orientation of the proposed IT to the evaluation of the project as a whole; the need to train the neural network on a large volume of historical data; the impossibility of quantitatively assessing changes in individual project tasks.

A separate issue is the analysis of resistance to changes by project employees. Existing methods for assessing resistance are based on closed questionnaires and binary classifications. However, such methods, as indicated in [18], limit the expression of opinions and do not provide a nuanced segmentation of employees' positions on changes. Therefore, in [18] it is proposed to use an innovative automated methodology for analyzing resistance to changes by project employees, which combines specialized Large Language Models (LLM) with a zero result (in particular, DeBERTa-v3-large-zeroshot) and rapid engineering methods. However, the disadvantage of this methodology is that it, like existing methods, is based on employees' responses to pre-prepared questionnaires. Therefore, this issue requires additional scientific research that goes beyond the scope of this study.

In general, the limitations and shortcomings inherent in modern research in the field of change management in projects and, in particular, in IT projects, can be formulated as follows:

- the absence in the overwhelming majority of widespread applied change management methods of a tool for quantitatively assessing these changes;
- the orientation of the overwhelming majority of methods and IT change management precisely on managing project personnel and stakeholders, as well as their relationships with each other within the project life cycle;

- the use of neural networks for various purposes in the proposed modern IS and IT change management and assessment studies, which significantly complicates the design and implementation of these IS and IT;

- the use of large language models in IS and IT change management and assessment, which leads to the formation of change assessments based not on specific actions of project stakeholders, but on their statements on this issue.

Therefore, the main direction of work on automation of change management is the expansion of existing project management systems and, in particular, IT projects by developing separate analytical services. These services should be aimed at solving the problems of assessing changes in those phases, processes and activities of an IT project that are relevant for stakeholders in specific periods of time. A feature of these services should be their readiness for operation in the conditions of the so-called "cold start" (i.e., in the absence of an array of historical data or a small amount of such data for a specific project).

An example of research in this direction is the work [19], devoted to the analysis of the impact of changes together with differences in the code during the verification of the software code of an IT project. To carry out this analysis, in [19] it was proposed to combine methods of dependency analysis based on call graphs and methods of intelligent history analysis. Using this combination of methods made it possible to calculate a set of file metrics and an overall risk score for each change request. The obtained estimates were not very accurate, but their accuracy generally satisfied stakeholders and IT project personnel. In addition, the time of analytical calculations during the experimental verification of the obtained combination of methods ranged from 7.4 to 22.43 seconds [19]. This makes it possible to apply the corresponding IT to change management in the management systems of any IT projects based on almost any Agile or hybrid methodologies and frameworks.

Therefore, the purpose of this study is to develop a service for automated solution of the problem of quantitative assessment of changes for the long-term IT project management system. The operation of this service will reduce the costs of implementing long-term IT projects by reducing unplanned time costs for performing individual tasks of these projects.

To achieve this goal, the following tasks were solved in the study:

- to improve the method based on the Beckhard and Harris's model for quantitative assessment of changes in a long-term IT project;
- to develop IT elements of quantitative assessment of changes based on the improved method;
- to carry out experimental verification of the obtained research results.

5.3 Materials and methods

The object of the study is the change management process in an IT project. This process is not included in the standard processes of the IT product life cycle as a system [4, 20], but is recognized as a typical process of the current change management framework [3]. This process can be added to any process model of an IT project without significant changes in this model.

The main hypothesis of the study is the hypothesis of the possibility of improving the quality of assessment of changes in the execution time of individual tasks of long-term IT projects by developing and implementing a service that will allow automating the solution of the change assessment problem.

The developed service is based on a method based on the Beckhard and Harris's model [10].

This method consists of the following stages:

- "Internal organizational analysis";
- "Determination of the need for changes";
- "Analysis of differences between the current state and the desired one";
- "Action planning";
- "Transition management".

"Internal organizational analysis" stage. The purpose of this stage is to determine the general attitude towards change in the organization. At this stage, it is necessary to identify employees who may resist change. In particular, it is necessary to identify any external factors that may hinder the change process.

"Determination of the need for changes" stage. This stage is necessary in order to create a basis for implementing change. Key change participants must agree that change is necessary for the success of the organization. This agreement assumes that change participants can clearly articulate where they want to take the organization and why implementing change will help bring the organization closer to the desired state. It is also necessary to have an idea of the consequences associated with refusing to implement change [10].

"Analysis of the differences between the current state and the desired state" stage. Before implementing change, it is first necessary to determine what deviations exist between the current state of the organization and what it should be. Determining these deviations is important in order to clearly formulate an understanding of the future of the organization.

"Action planning" stage. A change plan is formed and submitted for implementation. It is necessary to identify the key participants in the change process and the responsibilities of everyone who makes the changes.

"Transition management" stage. After the change is implemented, the employees implementing the changes are responsible for continuously monitoring the progress of the changes and making adjustments.

The main disadvantages of this method are:

- the lack of the ability to quantitatively assess the changes that arise during the implementation of the IT project;
- the subjectivity of making decisions on change management, which depend on the individual competence of decision-makers during the implementation of the stages of the method under consideration.

To develop a service that will automate the solution of the problem of quantitatively assessing changes in long-term IT projects, it is proposed to apply a descriptive approach. This approach is based on the following concepts: SCRUM, sprint, task, descriptor, change indicator.

SCRUM is a project management methodology commonly used in software development. It is based on iterative and incremental approaches to product development, where the team works in short cycles (sprints) [21].

A sprint is a short period of time (usually two to four weeks) in which a project team works on a specific set of tasks. This practice is implemented when using the Agile software development methodology to increase the efficiency and transparency of the development process [21].

A task is a separate task with a unique identification number that is performed within a sprint by a specific project team to achieve a set goal or result. A task is most often evaluated by its complexity and execution time. It has the following characteristics: description, creation date, priority, estimated execution time, responsible executor, execution status, acceptance criteria, deadline, comments, and additional materials.

A descriptor is a quantitative characteristic of a task that provides descriptive information about the task in a numerical format. A descriptor can describe certain aspects of tasks, such as: information about the use of a certain technology, information about the team that will perform the task, etc.

Change indicator – a numerical indicator that can be used to assess changes when they are introduced. The change indicator can be the time required to complete a task, or the difference between the planned time and the time spent on completing the task.

The descriptor approach, which is proposed for assessing changes at the level of IT project tasks, includes the following processes:

- collecting and storing in a database information that will describe the characteristics of the tasks being performed in the form of descriptors;
- building statistical models to find the dependence of the change indicator on the descriptor in order to quantitatively assess the changes.

A condition for using the descriptor approach is the availability of sufficient information about the tasks being performed. For long-term IT projects (a distinctive feature of which is the characteristic accumulation of information on completed tasks in a significant amount, which allows this information to be analyzed using statistical methods), this condition is met.

5.4 Results of improving the method of quantitative assessment of changes in a long-term IT project

5.4.1 Classification of descriptors that describe tasks and changes in an IT project

To use the descriptor approach within the framework of the implementation of a long-term IT project, the following classification of descriptors is proposed:

- technical descriptors;
- task feature descriptors;
- team description descriptors;
- testing process descriptors;
- web page accessibility descriptors;
- other descriptors.

The number of types of descriptors by which the classification is carried out may vary depending on the requirements of the IT project.

A descriptor may have a binary value: "Yes" (or "1" or "True") or "No" (or "0" or "False"). For example, the descriptor of the change of the library for testing modules for a certain software component can have the value "1" (if the task requires changing the library for testing the component) or "0" (if not).

Examples of a descriptor with a simple numerical value can be a descriptor of the number of lines of code for which it is necessary to implement functional testing, or a descriptor of the number of errors in the code that must be fixed during the task.

When using the descriptor approach, it is recommended to provide the ability to enter descriptors into the database to all members of the project team.

It is proposed to carry out the process of updating the descriptor database within each sprint and add information about the presence of this process to the readiness requirements of tasks that exist in the project (definition of done, DoD) [22].

If there is a database of descriptors and change indicators, it is possible to build statistical models of the dependence of change indicators on descriptors.

It is proposed to build a unidimensional model, based on the following reasons:

- simplicity of calculations;
- the need for a smaller amount of data for model development;
- the absence of a problem associated with multicollinearity of the proposed descriptors [23].

A unidimensional model may be sufficient to identify the main trends and dependencies in the data, especially at the initial stages of the study.

5.4.2 Development of an improved method for quantitative assessment of changes

Using a descriptive approach, the basic method based on the Beckhard and Harris's model [10] was adapted to change management at the level of individual IT project tasks. As a result of this adaptation, an improved method for quantitative assessment of changes was obtained, which is presented as a sequence of stages and individual activities performed within specific stages. The scheme of implementation of the stages of the method and their most important activities, taking into account the cyclical transitions between activities and stages, is shown in **Fig. 5.1**.

The stage "Internal project analysis" (Stage 1) is intended to form a description of a long-term IT project as a basis for further quantitative assessment of changes. This stage is proposed to be considered as a set of the following activities:

- assessment of current processes and change management systems at the task level;
- identification of employees who may have significant resistance to change;
- identification of external factors that may hinder the change process;
- analysis of resources available for implementing changes;
- determination, description and classification of descriptors;
- determination of a place to store descriptors (database);
- determination and description of change indicators and their storage location;
- appointment of those responsible for forming the descriptor database and its updating.

These activities within Stage 1 can be performed sequentially or in parallel.

The determination, description and classification of descriptors for current tasks should be performed at each iteration of the project.

The stage "Collection of information in the form of descriptors at each iteration of the project" (Stage 2) is necessary for the formation of a descriptor database. This stage consists of operations for collecting and storing descriptor values

determined in the results of Stage 1. These operations can be implemented both by means of the existing IS for managing a long-term IT project and by means of the service being created.

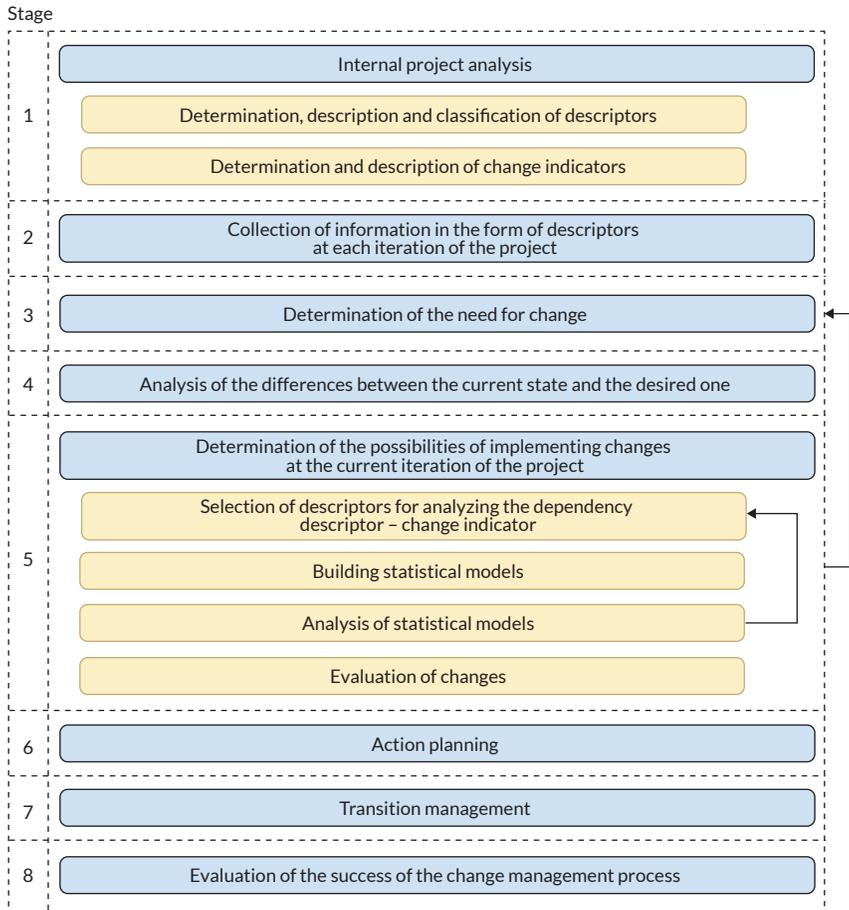


Fig. 5.1 Stages and key activities of the improved quantitative change assessment method

The stage "Determination of the need for changes" (Stage 3) is necessary to create a basis for implementing changes. At this stage, most of the work can be

performed by a business analyst. It is proposed to consider this stage as a set of the following activities:

- identifying specific areas where changes are needed at the task level;
- assessing the needs of users and customers in making changes to tasks;
- setting goals and expected results from implementing changes.

The stage "Analysis of differences between the current state and the desired one" (Stage 4) is necessary to identify the impact of changes on the IT project. This stage is proposed to be considered as a set of the following activities:

- identification of a set of significant deviations that exist between the current state of the analyzed task and its state after making changes;
- assessment of possible risks and the impact of changes on the project;
- comparison of technical characteristics of the current and desired state of the project.

The stage "Determination of the possibilities of implementing changes in the current iteration of project execution" (Stage 5) is necessary for quantitative assessment of the changes identified in Stage 3 and analyzed in Stage 4. At this stage, most of the work should be performed using the tools of the created service. This stage is proposed to be considered as a set of the following activities:

- selection of descriptors for analyzing the dependence of the change indicator on the descriptor;
- construction of statistical models;
- analysis of statistical models;
- assessment of changes at the task level.

A set of descriptors can be analyzed by the mean value of the descriptors, but this analysis can have significant drawbacks. Analyzing data using the mean (average) is a simple way to describe the central tendency of the data. However, this approach may not be predictive in cases where the data have high variability or when they are unevenly distributed. Especially in cases where the data have a large number of outliers or asymmetry, using the mean can lead to incorrect or incomplete understanding of the data.

Taking into account the features of the descriptor approach, within the framework of Stage 5, it is proposed to investigate the relationship between the indicators of change (the desired indicators) and the values of the descriptors. This relationship can be represented by the formula

$$y_i = \beta_0 + \beta_1 \cdot x_i, \quad (5.1)$$

where y_i - the dependent variable (change indicator); x_i - the independent variable (descriptor); β_0, β_1 - regression parameters.

After that, the task of finding the values of the regression parameters is solved.

There are many methods for building regression models. However, the most commonly used method is the Ordinary Least Squares (OLS).

Before using the OLS method, it is necessary to check the fulfillment of the prerequisites for using regression analysis. As a result of such analysis, outliers may be detected that will need to be processed.

In the case of outliers in the data collected during the implementation of the project tasks, the developed combined method proposes to use Huber regression, because it has properties related to robustness [24].

After building statistical models, it is proposed to check their performance using the following indicators [24]:

- coefficient of determination (R^2);
- forecasting coefficient (Q^2);
- forecasting coefficient calculated using the Leave-one-out cross-validation procedure (Q_{LOOCV}^2);
- standard deviation (σ);
- assessment of the significance of the statistical model (F-criterion).

If, during the verification of statistical models, these indicators acquire unsatisfactory values, another descriptor is selected and the statistical models are recalculated.

If the values of these indicators are satisfactory, the changes are assessed using the constructed model.

When using the obtained assessment, an assumption is made about the possibility of implementing changes at the task level in the current iteration of the IT project. If the assessment of the required task completion time corresponds to this possibility, the changes are approved. If the assessment does not correspond to the possibility of implementing changes, the results of Stage 3 are analyzed, after which, if necessary, the possibilities of either attracting additional human resources from other teams or overtime work of the current team members are analyzed. If it is determined that the changes cannot be implemented in the current project iteration, a new task is created. This task is included in the list of tasks when planning subsequent project iterations.

The stage "Action Planning" (Stage 6) is necessary for carrying out operations to re-plan the IT project taking into account the introduced and evaluated changes and implementing this plan. It is proposed to consider this stage as a set of the following activities:

- identification of key participants in the change process and responsibilities of each person making the changes;

- development of a specific change implementation plan, including resources and deadlines;
- determination of the sequence of steps for implementing the changes;
- preparation of a communications plan to inform customers about the changes and their impact on the project.

The stage "Transition Management" (Stage 7) is necessary for operational management of work on the direct implementation of the IT project iteration. It is proposed to consider this stage as a set of the following activities:

- implementation of the change plan for the task;
- monitoring the impact of changes on the project and timely identification of problems;
- providing support for users during the transition to a new state of the project.

The stage "Evaluation of the success of the change management process" (Stage 8) is necessary to analyze the progress of the IT project iterations and assess the success of the planned changes to the IT project. This stage is recommended to be carried out once every several (two or more) iterations of the project or at the request of the project stakeholders.

In order to assess the success of the change management process, it was proposed to use the following indicators at Stage 8 [5]:

- the share (in percent) Ch_{cp} of time changes canceled due to the impossibility of their implementation, excluding changes that became unnecessary for a certain period;
- the share (in percent) of time changes that were accepted by the customer and approved as successfully implemented for a certain period of time, Ch_{cp} ;
- the difference between the time spent on work and the estimated time for a certain period of time, δCh_T ;
- the share (in percent) of changes that were completed on time for a certain period of time, Ch_{Ttp} .

The share (in percent) of changes canceled over a certain period of time due to the impossibility of their implementation Ch_{cp} , excluding changes that have become irrelevant, can be calculated by the formula [5]

$$Ch_{cp} = \frac{Ch_c}{Ch} \cdot 100\%, \quad (5.2)$$

where Ch_c – the number of canceled changes for a certain period of time due to the impossibility of their implementation, excluding changes that have become irrelevant; Ch – the total number of changes for a certain period of time.

Large values Ch_{cp} indicate poorly planned changes.

The share (in percent) of changes that were accepted by the customer and approved as successfully implemented for a certain period of time Ch_{sp} can be calculated by the formula [5]

$$Ch_{sp} = \frac{Ch_s}{Ch} \cdot 100\%, \quad (5.3)$$

where Ch_s – the number of changes that were accepted by the customer and approved as successfully implemented for a certain period of time.

A large value Ch_{sp} indicates a better change management process.

The difference between the time spent on performing tasks of a long-term IT project and the estimated time for a certain period of time δCh_T can be calculated by the formula [5]

$$\delta Ch_T = \frac{\sum_{i=0} (t_i^{(p)} - t_i^{(a)})}{t \cdot i}, \quad (5.4)$$

where $t_i^{(p)}$ – the time planned for implementing the change, days; $t_i^{(a)}$ – the time spent on implementing the change, days; t – the time period for evaluation, days; i – the number of changes for the time period t .

The indicator δCh_T indicates whether changes are performed on time and in accordance with the change plan. The lower the indicator, the better organized the change management.

The share (in percent) of changes that were completed on time for a certain period of time Ch_{ITp} can be calculated by the formula [5]

$$Ch_{ITp} = \frac{Ch_{IT}}{Ch} \cdot 100\%, \quad (5.5)$$

where Ch_{IT} – the number of changes that were completed on time for a certain period of time.

A high value Ch_{ITp} indicates a better change management process and adherence to the planned schedule.

The use of the proposed improved method for solving the problem of quantitative assessment of changes made it possible to formulate the main requirements for the service that should provide an automated solution to this problem. To ensure the possibility of multiple use of the obtained solutions, this service during design and implementation (design & development) is proposed to be considered as a separate IT for automated solution of the problem of quantitative assessment of changes.

5.5 Elements of information technology for automated solution of the problem of quantitative assessment of changes

IT for automated solution of the problem of quantitative assessment of changes (hereinafter referred to as IT for quantitative assessment of changes) is proposed to be developed for automated implementation of the proposed improved method. Therefore, it was decided to use this IT to automate only those stages of the improved method that are directly related to the collection, processing and storage of data and information on changes that arise during the implementation of a long-term IT project.

These stages include [5]:

- Stage 1 "Internal project analysis";
- Stage 4 "Analysis of differences between the current state and the desired one";
- Stage 7 "Transition management".

Based on this decision, it was proposed to present the developed IT as a sequence of the following stages and steps [5].

Stage 1. Survey of the team of performers on the current perception of the change management process.

Step 1.1. Forming a questionnaire and conducting a survey of all employees who form the IT project teams regarding their current attitude to the change management process.

Step 1.2. Processing the survey results and forming current assessments of the level of employee satisfaction with the change management process.

Stage 2. Forming and storing descriptors of individual IT project work.

Step 2.1. Determining the set of descriptors of individual IT project work.

Step 2.2. Forming sets of values of the defined descriptors.

Step 2.3. Storing the formed sets of values of the defined descriptors.

Stage 3. Statistical analysis of IT project descriptors.

Step 3.1. Determining indicators of IT project changes.

Step 3.2. Selecting the defined descriptors to create a change model.

Step 3.3. Analysis of the sets of values of the selected descriptors for the presence of outliers.

Step 3.4. If the analysis results obtained as a result of Step 3.3 indicate the absence of outliers, then build a change model using OLS. Otherwise, build a change model using the Huber regression method.

Step 3.5. Calculate the performance indicators of the built change model.

Stage 4. Survey of the team of performers on the final perception of the change management process.

Step 4.1. Form a questionnaire and conduct a survey of all employees who form the teams of performers of the IT project on the final attitude to the change management process.

Step 4.2. Process the survey results and form final assessments of the level of employee satisfaction with the change management process.

A description of the IT architecture of quantitative change assessment in the form of a data flow diagram is given in Fig. 5.2 [5]. When creating this diagram, the Yordon-DeMarco notation was used. Fig. 5.2 does not indicate the names of the flows that are directly related to the data warehouses, because these names coincide with the names of the data warehouses themselves.

For the further implementation of IT, the features of its technology stack were determined. The technology stack will be understood here and now as a set of technologies that are used together to develop and support software [25].

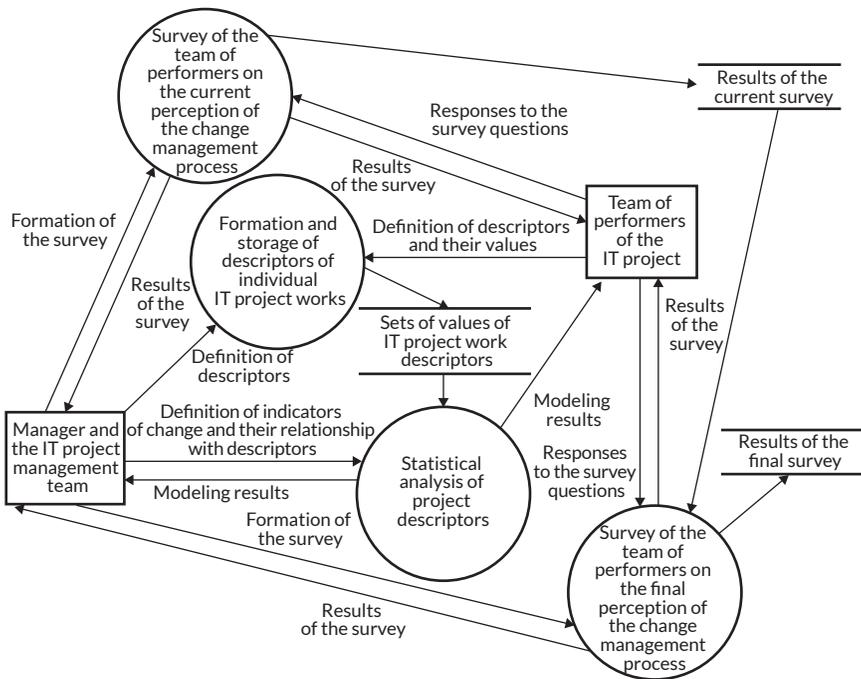


Fig. 5.2 Description of the architecture of the information technology of quantitative change assessment
Source: [5]

The technological stack of IT quantitative change assessment has the following components:

- for the development of the elements "Survey of the team of performers on the current perception of the change management process" and "Survey of the team of performers on the final perception of the change management process", it is proposed to use the Google Forms service, which is included in the free Google Docs editor package from Google;
- for the development of the element "Formation and storage of descriptors of individual IT project works", it is proposed to use existing IT project management systems (for example, Jira [26]) and tools for storing descriptor descriptions and their numerical values (for example, Microsoft Excel, or its analogues, or database management systems Firebase, MongoDB, PostgreSQL, etc.);
- for the development of the element "Statistical analysis of IT project descriptors", it was proposed to develop a specialized service using the Python programming language.

Microsoft Excel was used to implement data warehouses where it was planned to store descriptor descriptions and their numerical values.

A fragment of the program code for calculating the values of the Gaussian density function is shown in **Fig. 5.3**. A fragment of the program code for plotting the calculated Gaussian density function and checking the results obtained using the three-sigma method is shown in **Fig. 5.4** [5].

A fragment of the program code for creating a model using the OLS method is shown in **Fig. 5.5**. A fragment of the program code for creating a model using the Huber regression method is shown in **Fig. 5.6** [5].

A fragment of the program code for calculating the performance indicator of the model "Coefficient of determination" is shown in **Fig. 5.7**. A fragment of the program code for calculating the performance indicator of the model "Forecasting coefficient" is shown in **Fig. 5.8**. A fragment of the program code for calculating the performance indicator of the model "Forecasting coefficient Q_{LOOCV}^2 " is shown in **Fig. 5.9** [5].

A fragment of the program code for calculating the performance indicator of the model "Standard deviation (σ)" is shown in **Fig. 5.10**. A fragment of the program code for calculating the performance indicator of the model "F-criterion" is shown in **Fig. 5.11** [5].

```
// функція для обчислення значень гауссівської функції щільності (PDF)
function gaussianPDF(x, mean, variance) {
  const stdDev = Math.sqrt(variance);
  return (1 / (stdDev * Math.sqrt(2 * Math.PI))) * Math.exp(-((x - mean) ** 2) / (2 * variance));
}
```

Fig. 5.3 A fragment of the program code for calculating the values of the Gaussian density function
Source: [5]

```

// функція для побудови графіку розподілу Гауса у вигляді гистограми та перевірки методом трьох сигм
export function plotGaussianDistribution(x, chartRef2) {
  // Підготовка даних для побудови
  const mean = math.mean(x); // Середнє значення x
  const variance = math.variance(x); // Дисперсія x
  const stdDev = Math.sqrt(variance); // Стандартне відхилення

  // Обчислення значень гауссівської функції для кожного x
  const data = x.map(value => ({
    x: value,
    y: gaussianPDF(value, mean, variance)
  }));

  // Вивід корисної інформації в консоль
  console.log('Середнє значення (Mean): ${mean}');
  console.log('Дисперсія (Variance): ${variance}');
  console.log('Стандартне відхилення (Standard Deviation): ${stdDev}');

  // Перевірка за методом трьох сигм
  const lowerBound = mean - 3 * stdDev;
  const upperBound = mean + 3 * stdDev;
  const outliers = x.filter(value => value < lowerBound || value > upperBound);

  console.log('Аномальні значення (за межами 3σ): ${outliers}');

  new Chart(chartRef2.current.getContext('2d'), {
    type: 'bar', // Використання гистограми
    data: {
      labels: x,
      datasets: [{
        label: 'Гауссівський розподіл',
        data: data.map(point => point.y),
        borderColor: 'rgba(75, 192, 192, 1)',
        backgroundColor: 'rgba(75, 192, 192, 0.2)',
        borderWidth: 1
      }]
    },
    options: {
      scales: {
        x: {
          type: 'linear',
          position: 'bottom',
          title: {
            display: true,
            text: 'Значення X'
          },
        },
        y: {
          type: 'linear',
          position: 'left',
          title: {
            display: true,
            text: 'Густина ймовірності (Probability Density Function)'
          },
        },
      },
      plugins: {
        annotation: {
          annotations: {
            box1: {
              type: 'box',
              xMin: lowerBound,
              xMax: upperBound,
              yMin: 0,
              yMax: Math.max(...data.map(point => point.y)),
              backgroundColor: 'rgba(0, 255, 0, 0.1)',
              borderColor: 'rgba(0, 255, 0, 0.5)',
              borderWidth: 1,
              label: {
                content: '3σ Range',
                enabled: true,
                position: 'center'
              }
            }
          }
        }
      }
    }
  });
}

```

Fig. 5.4 A fragment of the program code for plotting the calculated Gaussian density function and checking the results obtained using the three-sigma method
Source: [5]

```
// Простий метод МНК (метод найменших квадратів)
export function leastSquaresRegression(x, y) {
  const n = x.length;
  const X = math.concat(math.reshape(x, [n, 1]), math.ones([n, 1]), 1);
  const theta = math.multiply(math.inv(math.multiply(math.transpose(X), X)), math.multiply(math.transpose(X), y));
  return theta;
}
```

Fig. 5.5 A fragment of the program code for creating a model using the least squares method
Source: [5]

```
while (iteration < maxIterations) {
  let updatedTheta = [0, 0];
  let totalLoss = 0;

  for (let i = 0; i < n; i++) {
    const r = y[i] - (thetaHR[0] * x[i] + thetaHR[1]);
    const loss = huberLoss(r, delta);
    const weight = Math.sqrt(delta / (r * r + delta));

    // Construct weighted matrices
    const weightedX = [X[i][0], X[i][1]]; // Select columns [0, 1] (corresponding to x and the intercept)
    const weightedY = yMatrix[i][0]; // Access element from yMatrix directly using array indexing

    // Update coefficients using weighted Least squares
    updatedTheta = math.add(updatedTheta, math.multiply(weight, weightedX, weightedY));

    totalLoss += loss;
  }

  // Перевірка на збіжність за зміною загальних втрат
  if (Math.abs(totalLoss - previousLoss) < tolerance) {
    break; // Зупиняємо ітерації, якщо втрати майже не змінюються
  }

  // Оновлення коефіцієнтів регресії на основі вагованих МНК
  thetaHR = updatedTheta;
  previousLoss = totalLoss; // Оновлюємо попередні втрати для порівняння
  iteration++;
}

return thetaHR;
}
```

Fig. 5.6 A fragment of the program code for creating a model using the Huber regression method
Source: [5]

```
function calculateR2(actual, predicted) {
  const meanActual = math.mean(actual);
  const totalSumOfSquares = math.sum(actual.map(val => Math.pow(val - meanActual, 2)));
  const residualSumOfSquares = math.sum(predicted.map((val, i) => Math.pow(actual[i] - val, 2)));
  const r2 = 1 - (residualSumOfSquares / totalSumOfSquares);

  return r2;
}
```

Fig. 5.7 A fragment of the program code for calculating the performance indicator of the model "Determination coefficient R^2 "
Source: [5]

```
// Розрахунок простого  $Q^2$  для регресії
function calculateSimpleQ2(y, yPred) {
  const meanY = math.mean(y);
  const totalSumOfSquares = math.sum(y.map(val => Math.pow(val - meanY, 2)));
  const sumSquaredErrors = math.sum(y.map((val, i) => Math.pow(val - yPred[i], 2)));
  const simpleQ2 = 1 - (sumSquaredErrors / totalSumOfSquares);

  return simpleQ2;
}
```

Fig. 5.8 A fragment of the program code for calculating the performance indicator of the model "Forecasting coefficient Q^2 "
Source: [5]

```

// Розрахунок Q2 за LOOCV для регресії
function calculateQ2LOOCV(x, y, regressionFunction, addParam) {
  const n = x.length;
  let sumSquaredErrors = 0;
  let totalSumOfSquares = 0;

  for (let i = 0; i < n; i++) {
    const xTrain = [...x.slice(0, i), ...x.slice(i + 1)];
    const yTrain = [...y.slice(0, i), ...y.slice(i + 1)];
    const xTest = x[i];
    const yTest = y[i];

    const theta = regressionFunction(xTrain, yTrain, addParam);

    // Перевірка, чи отримано коректні значення коефіцієнтів theta
    if (theta !== null && typeof theta === 'object' && theta.length >= 2) {
      const yPred = theta[0] * xTest + theta[1];
      const squaredError = Math.pow(yTest - yPred, 2);
      sumSquaredErrors += squaredError;

      const meanY = math.mean(yTrain);
      const totalSquare = Math.pow(yTest - meanY, 2);
      totalSumOfSquares += totalSquare;
    } else {
      console.error("Regression function did not return valid coefficients.");
      // Опціонально можна повернути null або інше значення у випадку помилки
      return null;
    }
  }

  const Q2 = 1 - (sumSquaredErrors / totalSumOfSquares);
  return Q2;
}

```

Fig. 5.9 A fragment of the program code for calculating the performance indicator of the model "Forecasting coefficient Q_{LOOCV}^2 "
Source: [5]

```

// StandardDeviation
function calculateCalculatedStandardDeviation(actual, calculated, numParams) {
  const n = actual.length;
  const residuals = actual.map((val, i) => val - calculated[i]);
  const squaredResiduals = residuals.map(residual => Math.pow(residual, 2));
  const sumSquaredResiduals = math.sum(squaredResiduals);

  const sigmaCalc = Math.sqrt(sumSquaredResiduals / (n - numParams - 1));
  return sigmaCalc;
}

```

Fig. 5.10 A fragment of the program code for calculating the performance indicator of the model "Standard deviation (σ)"
Source: [5]

```

function calculateFStatistic(R2, numParams, numDataPoints) {
  const F = (R2 / (1 - R2)) * ((numDataPoints - numParams - 1) / numParams);

  return F;
}

```

Fig. 5.11 A fragment of the program code for calculating the performance indicator of the model "F-criterion"
Source: [5]

5.6 Description of an example of solving the problem of quantitative assessment of changes during the implementation of a long-term IT project

5.6.1 Description of the features of the long-term IT project "Web Constructor"

Experimental verification of the obtained research results was proposed to be carried out during the implementation of the long-term "Web Constructor" IT project. The duration of the implementation of this IT project is 6 years.

The main result of this IT project is the "Construct" system. This system implements an administrative portal, where customers can use the library of components to create sites. The "Construct" system supports more than 20 languages and allows to create web pages for most countries of the world. The results obtained from the implementation of the long-term "Web Constructor" IT project are already in operation and continue to expand.

The main goal of the long-term "Web Constructor" IT project is the development and support of the created "Construct" system.

According to the classification of the main groups of projects by interests of the Project Management Institute (PMI), this project belongs to IS development projects [2, 3].

Table 5.2 provides an additional classification of the "Web Constructor" project. This classification was carried out according to classification criteria according to the materials of sources [27, 28].

Table 5.2 Additional classification of the "Web Constructor" project

Classification feature	Project type
By scale	Large
By complexity	Technically complex
By implementation time	Megaproject
By resource constraints of a set of projects	Program
By the nature of the project and the level of participants	International
By the nature of the project's target task	Marketing
By the main reason for the project	Need for structural and functional transformations
By the location of the customer	External
Degree of customer participation in the project	Average

The "Web Constructor" project is divided into subprojects depending on the requirements. For example, during the project implementation, subprojects took place to update designs, transition to more progressive and competitive technologies, add new functionality, and others. The project continues to scale, so new teams and subprojects are currently being created to perform individual business tasks.

During the implementation of the "Web Constructor" IT project and its individual subprojects, a lot of work is performed, the main of which are:

- creation of new system components;
- updating existing system components;
- creation of new web page templates;
- improving the performance of existing web pages;
- collection of analytical information on system usage;
- system support/updating;
- solving problems with content accessibility;
- transition to new technologies;
- implementation of the possibility of using the system in new countries;
- updating the system design.

To implement the "Web Constructor" IT project, the resources of the following IT company departments were used:

- Management department;
- Inclusion and accessibility department;
- Business analysis department (BA);
- Strategic architecture and consulting department (Solution Architects, SA);
- Back end (BE) development department;
- Front end (FE) development department;
- System efficiency and performance department;
- Software development and implementation automation and optimization department (DevOps);
- User interface design department;
- Software testing experts department (Quality Assurance, QA).

Several teams participate in the implementation of the IT project. Most teams are development teams, each of which has 6–10 employees. Such a team usually includes several BE developers, FE developers, and QA specialists. One of the project participants plays the leading role of the team leader and reports to other teams and divisions of the IT company on the success of the work, maintains communication with them.

The "Web Constructor" IT project is international and includes employees from different countries of the world.

When planning the work of the teams of the "Web Constructor" IT project, the provisions of the Agile SCRUM methodology were used [21]. The main type of iteration of this IT project is a sprint lasting 3 weeks.

When planning a specific sprint of the "Web Constructor" IT project, the tasks that are allocated in the backlog of this sprint can be attributed to one of the following classes:

- tasks that must be fully completed during the current sprint;
- tasks that must be completed during the next sprint;
- tasks that will be performed during several sprints.

If the task is selected with the intention of obtaining a finished version of the program (i.e. as a release task), the team should spend no more than 2 weeks on its implementation, because it must be checked by test engineers and customers during acceptance testing (UAT) in the software product environment. If the task is planned without taking into account obtaining a finished version of the program (i.e. as a non-release task), development can take place within the entire sprint. Therefore, testing the task will be planned within the next sprint.

The main technologies used in the implementation of the "Web Constructor" IT project and forming its technological stack are: JS, React, JAVA, SQL, HTML, CSS, Redux, TS, LESS, AWS.

5.6.2 Overview of the causes of changes in the long-term "Web Constructor" IT project

The risks and changes that occur during the implementation of the "Web Constructor" project are closely related to its features and classification. Therefore, before starting the experimental verification of the obtained research results, it is necessary to consider the current problems that may lead to changes in the "Web Constructor" IT project and the "Construct" system.

The most relevant problem encountered during the implementation of the long-term "Web Constructor" IT project is the problem of changing the characteristics of tasks during their execution. Changing these characteristics in the process of developing a project task after assessing the time required for its execution is a very common phenomenon [29].

The characteristic of the task is a description of what must be performed during the implementation of the task. The characteristics are described and divided by BA specialists into characteristics on the BE side and characteristics of the FE. Sometimes the same characteristic may require work from two parts (BE and FE).

In this case, for the requirements descriptors, it is proposed to include such a requirement in both descriptors.

When performing the "Web Constructor" IT project, the characteristics of the tasks are described in a table format and each of them is given a universal identifier. If the task is technical in nature, then the characteristics can be explained and described by the system architect. At the time of discussing the task before selecting it for execution during the work sprint, the characteristics should already be described and the team should familiarize themselves with them in advance. This provides an opportunity to resolve misunderstandings about the task before its final discussion (grooming, SCRUM stage) [21].

Depending on the characteristics, the tasks are divided into technical and business tasks. Within the framework of business tasks, customer requirements are fulfilled, technical tasks are solved in order to improve the technical performance and quality of the created product.

Another group of relevant problems of changes in the long-term "Web Constructor" IT project are problems that arise during the project implementation.

Firstly, such problems include restrictions on direct communication between individual departments. As a result of such restrictions, communications become more complicated and sometimes take place as a dialogue in the form of electronic correspondence. Such complications, in turn, can slow down the editing of elements of the Construct system and lead to changes that should be made already at the time of work on the relevant tasks.

Secondly, such problems include situations when the design department initiates changes in designs during the implementation of a certain functionality, that is, when the task has already passed the stage of evaluation by the development team. Examples of the most frequent changes that arise due to this reason include the following: replacing images, changing colors, changing fonts, changing sizes, introducing new designs for individual cases or screen sizes, etc.

Thirdly, such problems include situations in which designers do not always inform BA specialists about changes to graphic designs after making them. Direct communication between the development team and designers in the "Web Constructor" IT project is absent and occurs through communication with BA specialists. Such changes can make the characteristics described by BA specialists irrelevant and create the need to make changes at the task level [26].

The problems considered generate a set of changes that arise directly during the implementation of the IT project. If such changes are insignificant and the team has the opportunity to perform them within the task it is working on, these changes are made immediately. If the changes are critical and complex, they are made as

a separate task and given a higher priority. A task of this type is called a change request (Change Request, CR) [30]. In this case, another task planned within the current sprint will be replaced by this CR, if the team has the opportunity to complete it, or the requirements for the tasks in progress are simplified.

Another group of urgent problems of changes in the long-term "Web Constructor" IT project is associated with the need for interaction of individual subprojects with external organizations or internal teams. The results of the activities of these organizations and teams can significantly affect the progress of these subprojects and the entire IT project. For example, during the development of the server part, the JAVA programming language and the Adobe Experience Manager (AEM) platform are used. But the use of this platform in some cases, unfortunately, leads to difficulties and limitations in expanding the functionality. This leads to the fact that some tasks cannot be completed on time. In some cases, such tasks are postponed and a letter is written to the AEM developers with a request to fix the problems found.

Similar problems also arise during the development of the user interface of the "Construct" system. There is a separate internal subproject that provides a common library with components to be used at the "Web Constructor" IT project level. This leads to problems integrating individual system components with the library being created.

Some similar characteristics of tasks in parallel subprojects are implemented differently, which also leads to problems and complicates the integration process with related components and external systems.

Similar problems arise when using external libraries (for example, node package manager). There have been situations when updating the library version led to significant system defects and required replacing the used library or making a decision to use an old version of the current library. For such a decision, it was necessary to take the risk that refusing updates could cause system security problems. In such cases, it is possible to decide to wait for updates to the necessary library, because the development team of this library may know about the existing problem and plan to solve it in future releases.

A separate group of problems arises in the process of solving tasks of estimating the time required to perform a certain task and implement changes at the task level. There are cases when the team cannot estimate the time required to perform a task or implement changes due to insufficient information on the technical features. In such cases, it is proposed to perform the task for the purpose of more detailed research and estimation of the time required to perform the task or implement the change. Such a task is called a proof of concept (POC). However, the team may not always be aware of the risks and propose conducting such a study. In such cases, during

development, it becomes necessary to make changes to the characteristics or to calculate the amount of time required to perform the task.

Estimation of the time required to perform tasks in different teams of the "Web Constructor" IT project occurs differently. In development teams, time estimation is carried out by planning using Poker Planning technology. Scores are given in the range from 1 to 13 by all team members. The meeting at which the assessment process takes place usually takes place a week after the meeting to discuss the task with the BA specialists. This, in turn, leads to the fact that the assessments may not be accurate and relevant, because the team may not remember all aspects of the assessed task.

Among other problems that are the causes of changes in the long-term "Web Constructor" IT project, the following types of problems should be noted:

- a) problems arising from the work of the project teams under stressful conditions;
- b) problems arising from the need to compensate for work;
- c) problems arising from the complexity of the existing organizational structure of the IT company;
- d) problems arising from the low level of experience of individual IT company specialists.

Type of problem (a) is caused by the fact that due to high competition, IT project tasks must be performed quickly and efficiently. This requirement sometimes leads to overtime work by employees or work on weekends. Frequent cases of such work can lead to emotional fatigue of employees and worsen the effectiveness of the team and individual project participants.

Type of problem (b) is caused by the international crisis of the IT market. The consequence of this crisis is a situation in which customers prefer budget solutions and refuse to implement the latest technologies due to financial resources. This leads to increased competition in the market and a decrease in the technological level of the created "Construct" system.

Type of problem (c) is caused by the incomplete knowledge and skills of members of the project teams regarding the rights and obligations of managers of individual divisions of the IT company. Team members are not personally familiar with higher-level managers and may not know which link in the organizational structure to contact with various issues. This can lead to the problem of a slow response when an employee contacts its manager, and the manager is forced to go further through the links of the organizational structure of the IT company's management.

A type of problem (d) is caused by the low level of experience of a certain group of employees in working with parts of the developed system. This is due to the long duration of the implementation of the IT project "Web Constructor" – a significant

part of the system components was implemented by employees, some of whom are no longer working on the project. Among the problems of this group, one can single out the problem of poor-quality process of transferring information about the project and poor-quality process of adapting employees of the IT company.

According to the results of the review of the problems that occur during the implementation of the "Web Constructor" IT project, it was proposed to classify these problems for experimental verification of the obtained research results as follows:

- problems of quality of estimating the time required to complete tasks and make changes;
- problems of medium-term forecasting;
- problems of changing task characteristics;
- problems of an emotional nature;
- problems of the level of compensation;
- problems of the need for overtime work;
- problems of a technical nature;
- problems of communication;
- problems of the level of knowledge of employees;
- problems of organizational structure complexity;
- problems caused by the use of outdated documentation and the lack of a quality adaptation process for employees performing the project.

5.6.3 Description of the progress and results of experimental verification of the obtained results

It was decided to conduct an experimental verification of the obtained research results within the framework of the tasks and iterations of the "Web Constructor" IT project by one development team (the description of this team is given in **Subsection 5.6.1**). The IT project manager was appointed responsible for the implementation of the stages of the improved method.

Before the implementation of the improved method, the method based on the Beckhard and Harris's model was used during the project implementation [10].

During the implementation of Stage 1 of the improved method, the manager conducted a survey of the team regarding the perception of the change management process operating in the team at the task level. An anonymous questionnaire (Google Forms) was created for the survey and sent to each project participant. The questions and answers of the project participants are given in **Table 5.3**.

Table 5.3 Assessment of employees' attitude to the method based on the Beckhard and Harris's model

Questions for evaluation on a scale from 0 to 10	Employee								Average score per question
	1	2	3	4	5	6	7	8	
Speed of change implementation (10 – fast)	6	5	4	6	7	4	5	3	5
Comfort of change management processes (10 – comfortable)	5	4	5	6	7	6	8	7	6
Presence of stressful situations during change implementation (10 – absent)	7	8	6	7	7	6	8	6	6.875
Need for overtime work (10 – absent)	8	9	8	10	9	10	9	9	9
Difficulty of change assessment (10 – not difficult)	6	7	5	6	7	8	5	4	6
Average employee rating	6.4	6.6	5.6	7	7.4	6.8	7	5.8	6.575

According to the survey results, the average score of the current change management method was 6.575.

Then, with the participation of all project team members, an analysis of the use of descriptors was conducted.

A conditional name was chosen for each descriptor for use in the descriptor databases. The main descriptors at the time of discussion were the following:

- number of task characteristics (C_Amount);
- number of task characteristics to be implemented on the client side (C_FE_Amount);
- number of task characteristics to be implemented on the server side (C_BE_Amount);
- planned number of days to complete the task (W_Days_Planned);
- number of days spent to complete the task (W_Days_General);
- number of task characteristics at the time of completion of work on it (C_Amount_Final);
- difference between the number of task characteristics at the time of taking the task into work and the number of characteristics that occurs at the time of completion of work on the task (C_Delta);
- the difference between the number of days planned for the task at the time of its initiation and the number of days spent on its implementation at the time of its completion (W_Days_Delta).

During Stage 2, information was collected in the form of descriptors by entering them into the descriptor database.

The tasks that were performed during the last five sprints were analyzed. The description of the descriptors for these tasks is given in **Table 5.4**.

Table 5.4 Descriptors for tasks performed during the last five sprints

Task ID	C_Amount, pieces	C_Amount_ Final, pieces	C_FE_ Amount, pieces	C_BE_ Amount, pieces	W_Days- Planned, days	W_Days- General, days
WEBCO_2342	80	92	30	50	15	20
WEBCO_2343	80	94	65	15	23	31
WEBCO_2344	20	25	10	10	8	10
WEBCO_2345	35	38	15	20	7	8
WEBCO_2346	42	47	12	30	12	15
WEBCO_2347	76	85	26	50	9	14
WEBCO_2348	12	12	12	0	5	5
WEBCO_2349	56	61	26	30	16	18
WEBCO_2350	15	20	10	5	5	7
WEBCO_2351	71	75	25	46	14	16
WEBCO_2352	12	12	0	12	3	3
WEBCO_2353	7	10	7	0	3	4
WEBCO_2354	2	2	2	0	1	1
WEBCO_2355	1	8	1	0	2	6
WEBCO_2356	35	40	32	3	18	20
WEBCO_2357	8	10	1	7	4	5
WEBCO_2358	54	65	53	1	50	55
WEBCO_2359	34	53	25	9	74	84
WEBCO_2375	50	43	30	20	15	12
WEBCO_2376	40	33	25	15	12	9
WEBCO_2377	35	28	20	15	10	8
WEBCO_2380	20	15	12	8	7	5
WEBCO_2384	70	62	40	30	20	16
WEBCO_2385	50	42	28	22	15	12
WEBCO_2386	45	32	25	20	13	8
WEBCO_2387	40	30	22	18	12	8
WEBCO_2388	35	32	20	15	11	10
WEBCO_2342	80	92	30	50	15	20

During the next (sixth) sprint, descriptors were also entered into the descriptor database. The values of these descriptors are shown in **Table 5.5**.

Table 5.5 Descriptors for tasks performed during the sixth sprint

Task ID	C_Amount, pieces	C_Amount_ Final, pieces	C_FE_ Amount, pieces	C_BE_ Amount, pieces	W_Days- Planned, days	W_Days- General, days
WEBCO_2389	30	26	18	12	10	9
WEBCO_2395	20	20	12	8	8	8
WEBCO_2396	25	21	15	10	9	7
WEBCO_2397	30	35	18	12	10	13
WEBCO_2398	22	24	14	8	8	9
WEBCO_2399	35	35	20	15	11	11
WEBCO_2360	65	75	40	25	20	25
WEBCO_2361	25	59	15	10	8	23

During Stage 3 of the sixth sprint, when performing task WEBCO_2360, it became necessary to make changes at the task level. The task included updating the designs of the bottom menu component of the page (footer). During the task, it was discovered that changes to the graphical interface designs were necessary. The designer updated the graphical interface, and the business analyst updated the component characteristics in accordance with the design updates.

During Stage 4, the difference between the task states without changes and with changes was studied in detail. The team was explained the changes necessary for implementation.

The development team requested Stage 5 of the improved method. To build statistical models of changes, a descriptor was selected that determines the difference between the number of task characteristics at the time the task was started and the number of characteristics that occur at the time the task was completed (C_Delta). The change indicator was chosen as the difference between the number of days planned for the task at the time of its start and the number of days spent on its implementation at the time of completion of work on it (W_Days_Delta).

First, the prerequisite for using regression analysis was checked. It was carried out for data collected on the basis of the first five sprints. Using the developed software, a Gaussian distribution was constructed, presented in Fig. 5.12 [5].

The mean, variance, and standard deviation of the descriptor values collected for the first five sprints were calculated. The mean was 1.52. The variance was 62.72. The standard deviation was 7.92. There were no outliers outside 3σ . The check showed that there were no outliers in the data, so it was decided to build a model using the OLS method. The model calculations were performed for the descriptors collected from the first five sprints.

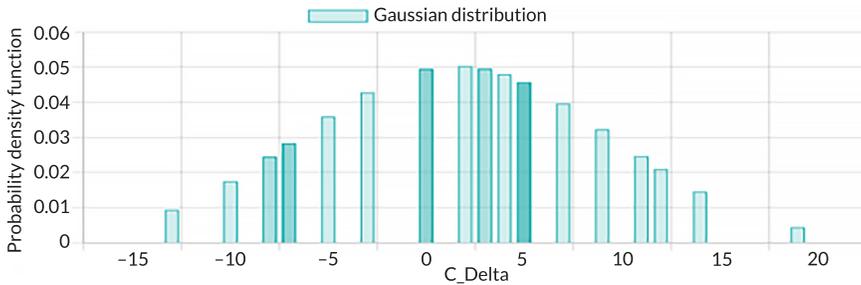


Fig. 5.12 Gaussian distribution for the data on the number of changed characteristics collected for the first five sprints
Source: [5]

The model obtained as a result of the OLS calculations is represented by the formula

$$y_i = 0.46 \cdot x_i + 0.26, \quad (5.6)$$

where y_i – the value of the change indicator W_Days_Delta (system property); x_i – the value of the descriptor C_Delta .

The change model, built using the OLS method for the data of the first five sprints, is shown in the graph (Fig. 5.13) [5].

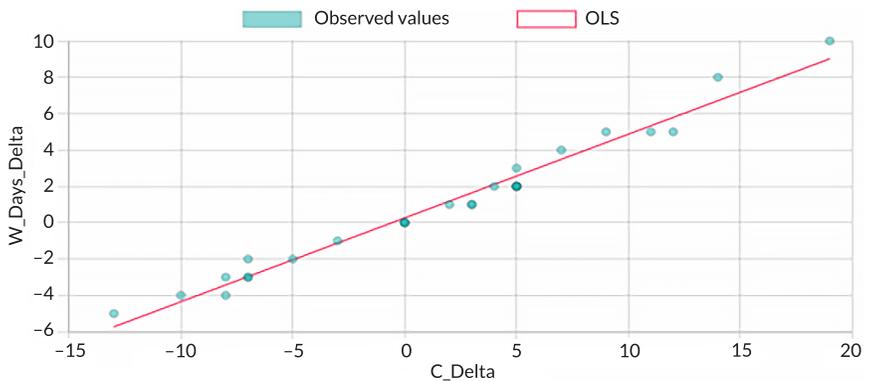


Fig. 5.13 Change model built using the OLS method for data from the first five sprints
Source: [5]

After building the model, its performance was checked using the following indicators: R^2 , Q^2 , Q_{LOOCV}^2 , F -criterion and standard deviation σ . The calculated performance indicators for the change model, built on the basis of descriptors for the tasks of the first five sprints, are given in **Table 5.6**.

Table 5.6 Performance indicators for the change model

Method	Regression equation	R^2	Q^2	Q_{LOOCV}^2	F	σ
OLS	$y_i = 0.46 \cdot x_i + 0.26$	0.98	0.98	0.97	481.61	0.59

The values of the calculated indicators showed that the model is workable and can be used to estimate the change indicator.

During the sixth sprint, issues were resolved regarding updating the list of characteristics of the WEBCO_2360 task after it was put into operation. The initial number of characteristics was 65 units. The team was tasked with choosing one of the proposed options:

- add 20 new characteristics to the current task;
- add 10 priority new characteristics to the current task and create a task with 10 other characteristics to be performed during the next sprint.

The calculation of the change implementation time estimates under the conditions of adding 10 and 20 new characteristics is given in **Table 5.7**.

Table 5.7 Calculation of the change implementation time required for the WEBCO_2360 task due to changes in the number of characteristics

Method	Regression equation	Time of change (20 characteristics)	Time of change (10 characteristics)
OLS	$y_i = 0.46 \cdot x_i + 0.26$	9.46	4.86

Using the resulting change estimate, it was decided to add 10 new features during the current sprint. The project manager assigned people responsible for making changes and checking them.

The changes took 5 days, which was slightly longer than planned. This fact showed that using the improved method, an estimate was obtained that was close enough to the actual value. The calculation of the ratio of the predicted value to the actual value is given in **Table 5.8**.

During the seventh sprint, issues were addressed regarding the reduction of the list of characteristics of the WEBCO_2361 task after it was taken into operation. The initial number of characteristics was 55 units. After taking the task into operation,

it was found that 7 characteristics were no longer relevant. The development team requested Stage 5 of the improved method.

Table 5.8 Calculation of the ratio of the predicted value of time to the actual value of time

Method	Forecast change, days	Actual change, days	Ratio of forecasted and actual results
OLS	4.86	5	0.97

To build statistical models of changes, it was decided to use the descriptor C_Delta and the change indicator W_Days_Delta , as for previous calculations. The data collected on the basis of the first six sprints were checked for outliers. The Gaussian distribution for the values of these descriptors, which were collected during the previous six sprints, is shown in **Fig. 5.14** [5].

The calculated mean was 2.4. The variance was 82.6. The standard deviation was 9.09. The three-sigma test showed the presence of an outlier in the data with a value of 34.

Since the test showed the presence of outliers, models were built using Huber regression. The models were calculated on the descriptor values of the first six sprints using the following values of the parameter δ : 1.345, 0.8, 0.1, 0.02, and 1.5. A fitness check was performed for the found change models. The results of the calculations are given in **Table 5.9**.

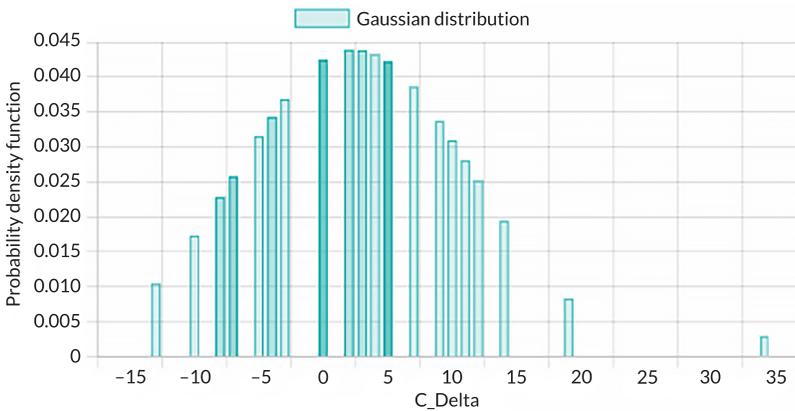


Fig. 5.14 Gaussian distribution for descriptor values collected during the previous six sprints
Source: [5]

Table 5.9 Models calculated using Huber regression and their fitness indicators

Parameter δ	Regression equation	R^2	Q^2	Q_{LOOCV}^2	F	σ
1.345	$y_i = 0.25 \cdot x_i + 0.01$	0.76	0.76	0.78	51	2.09
0.8	$y_i = 0.24 \cdot x_i + 0.01$	0.74	0.74	0.76	46	2.16
0.1	$y_i = 0.37 \cdot x_i + 0.01$	0.94	0.94	0.93	259	1.02
0.02	$y_i = 0.41 \cdot x_i + 0.02$	0.97	0.97	0.97	517	0.74
1.5	$y_i = 0.25 \cdot x_i + 0.01$	0.76	0.76	0.78	51	2.07

According to the calculated performance indicators, the value $\delta = 0.02$ was chosen for further calculations.

The model built using Huber regression for the data of the first six sprints is shown in the graph in Fig. 5.15 [5].

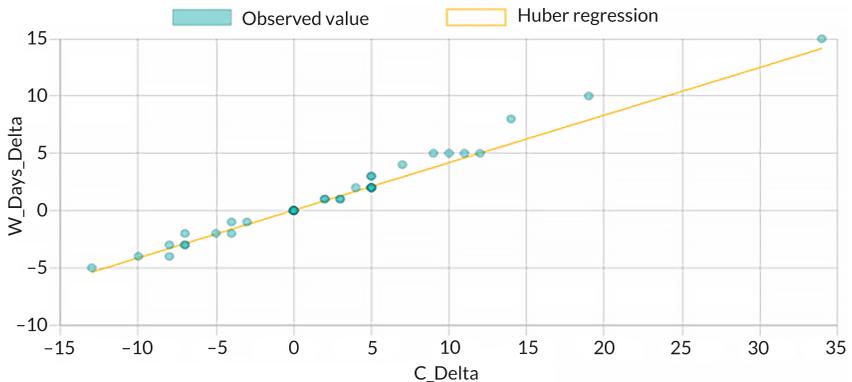


Fig. 5.15 Model built using Huber regression, for data from the first six sprints
Source: [5]

The calculated performance indicators of the model showed that the model is workable and can be used to estimate the indicator of changes in task execution time. The calculation of the estimate of released days under the condition of changing the number of characteristics of the task WEBCO_2361 is given in Table 5.10.

For task WEBCO_2361, it was decided to remove 7 characteristics from the task description in the current sprint. 4 days less were spent on solving the changed task. Calculations of the ratio of the predicted value to the actual value are given in Table 5.11.

Table 5.10 Calculation of the estimate of released days under the condition of changing the number of characteristics of the task **WEBCO_2361**

Method	Regression equation	Change indicator
Huber regression	$y_i = 0.41 \cdot x_i + 0.02$	2.85

Table 5.11 Calculation of the ratio of the predicted value to the actual value for task **WEBCO_2361**

Method	Forecasted value, days	Actual value, days	Ratio of forecasted and actual results
Huber regression	2.85	3.5	0.81

Thus, the calculated change estimates turned out to be close to real changes. Therefore, it was decided that the results of the experimental verification of the improved method and the elements of the corresponding IT are adequate to the real processes of implementing the long-term "Web Constructor" IT project.

5.7 Discussion of the research results

Improving the method based on the Beckhard and Harris's model using the descriptor approach made it possible to solve the following tasks:

- conduct a quantitative assessment of changes based on the values of descriptors that are formed in the process of implementing sprints of a long-term IT project by its performers;
- automate the solution of the problem of quantitative assessment of changes by forming and analyzing statistical models of the dependence of change indicators on descriptors.

The developed IT elements of quantitative assessment of changes allow to improve the quality of change management in a long-term IT project. This possibility was achieved through the use of the descriptor approach and statistical models of quantitative assessment of changes. The use of software implementation developed by IT greatly simplifies the work on assessing changes in the time parameters of individual tasks of a long-term IT project [5].

Unlike the derived method based on the Beckhard and Harris's model [10], the use of the improved method allowed to improve the performance indicators of the change assessment process. The results of comparing the values of these indicators

calculated for the derived method based on the Beckhard and Harris's model and after the implementation of the improved method are given in **Table 5.12**.

Table 5.12 Results of comparing the values of indicators of the change assessment process

Indicator	Application of the derivative method based on the Beckhard and Harris's model	Application of the developed information technology for quantitative assessment of changes	Change in the indicator
Proportion (in percent) of time changes canceled due to impossibility of their implementation Ch_{cp}	7	5	Decreased by 2
Proportion (in percent) of time changes that were accepted by the customer and approved as successfully implemented Ch_{sp}	92	95	Increased by 3
Difference (in hours) between the time spent on work and the estimated time, δCh_T	7	2	Decreased by 5
Proportion (in percent) of changes that were completed on time Ch_{tp}	79	86	Increased by 7
Employee attitude assessment towards change management processes (from 1 to 10)	6.575	8.025	Improved by 14.5%

The developed improved method has the following limitations:

- the duration of the project must be large enough to allow collecting the necessary amount of information for analysis;
- there is a need for additional storage of information for analysis;
- models built using this approach are relevant only for specific projects on the basis of which data was collected for building models (however, exceptions are possible).

The main limitation of the use of the developed IT elements is the need to operate IS and IT management of the work of project teams, which could provide the formation of derived data arrays for further calculation of the values of the selected descriptors of the tasks of a long-term IT project. In addition, a significant drawback of the developed IT is a significant increase in the duration of model calculations with an increase in the number of descriptors taken for analysis. To overcome this drawback, it is necessary to conduct additional research to reduce the time and computational complexity of the algorithms for implementing the developed IT [5].

Research on the development of an improved method for quantitative assessment of changes is proposed to focus on determining the possibility of its application for assessing and predicting other (except time) parameters of changes that arise during the implementation of a long-term IT project. To conduct such research, it will also be necessary to conduct additional research to determine the best indicators that characterize changes and the success of the change assessment process in IT projects of various types.

5.8 Conclusions

In the process of research, a method based on the Beckhard and Harris's model was improved for quantitative assessment of changes in a long-term IT project. To improve the method, it was proposed to use a descriptor approach. The improved method, unlike the existing one, allows for quantitative assessment of changes based on the values of descriptors that are formed during the implementation of sprints of a long-term IT project by its performers.

The results of improving the change assessment method were implemented in the form of IT elements of an automated solution to the problem of quantitative assessment of changes in the long-term IT project management system. A service architecture was selected for implementation, and the developed IT was considered as a means of implementing the corresponding service. The IT architecture was defined, a technological stack was proposed, and elements of the software implementation of the service were developed.

To experimentally verify the obtained research results, it was decided to test these results during the implementation of the long-term "Web Constructor" IT project. The test was performed on the results of the work of one project development team. The calculation used data obtained during the team's execution of seven project sprints.

The calculations were performed for two cases of changes that occurred during the sixth and seventh sprints. In the first case, the ratio of the predicted time estimate to the actual time spent on implementing the change was 0.97, in the second case – 0.81. These results showed that the predicted values of the indicators quite accurately coincide with the values of the time actually spent on implementing the changes. In general, the use of the improved method improved the indicators of change implementation during the implementation of the IT project (**Table 5.12**) and improved the team's attitude to the change management process by 14.5%.

Conflict of interest statement

The authors declare that there is no conflict of interest in relation to this paper, as well as the published research results, including the financial aspects of conducting the research, obtaining and using its results, as well as any non-financial personal relationships.

Use of artificial intelligence statement

The authors declare that they did not use artificial intelligence tools in preparing this manuscript.

References

1. Sakhno, Ye. Yu., Sidin, E. P., Korniiets, K. Ye. (2015). Modeling the effectiveness of realization the long-term projects. *Scientific Bulletin of Polissia*, 2 (2), 87–94.
2. Nastanova do zvodu znan z upravlinnia proiektamy. *Nastanova PMBOK (2021)*. Project Management Institute, Inc. Available at: https://learn.ztu.edu.ua/pluginfile.php/274061/mod_resource/content/1/PMBOK7_Ukr_ForPersonalUseOnly.pdf
3. *Managing change in organizations: a practice guide (2013)*. Project Management Institute, Inc., 127.
4. ISO/IEC/IEEE 12207:2017. *Systems and software engineering – Software life cycle processes (2017)*. IEEE. <https://doi.org/10.1109/ieeestd.2017.8100771>
5. Vasylytsova, N., Popova, A. (2025). Information technology for quantitative assessment of changes in a long-term IT project. *Management Information System and Devises*, 184, 5–21. <https://doi.org/10.30837/0135-1710.2025.184.005>
6. Zosym, M. (2025). Adkar model. Maxym Zosym. Available at: <https://www.max-zosim.com/adkar-model/>
7. The Prosci ADKAR Model. A powerful yet simple model for facilitating individual change. Prosci. Available at: <https://www.prosci.com/methodology/adkar>
8. AIM Accelerating Implementation Methodology. Peacock Hill Consulting. Available at: <https://imaworldwide.com/the-aim-methodology/>
9. Change Management – Embrace Evolve Thrive. AIM Seize The Future. Available at: <https://www.aim.com.au/leadership-strategy/courses/change-management-embrace-evolve-thrive>

10. Use a Beckhard & Harris Change Process to Create Your Team's Template to Success. Praxie. Available at: <https://praxie.com/beckhard-harris-change-process-online-tools-templates-web-software/>
11. Bridges Transition Model. William Bridges Associates. Available at: <https://wmbridges.com/about/what-is-transition/>
12. The 8 Steps for Leading Change. Kotter. Available at: <https://www.kotterinc.com/methodology/8-steps/>
13. Malik, P. (2025). The Kübler Ross Change Curve in the Workplace. The Whatfix Blog. Available at: <https://whatfix.com/blog/kubler-ross-change-curve/>
14. Malik, P. (2025). Lewin's 3-Stage Model of Change Theory: Overview. The Whatfix Blog. Available at: <https://whatfix.com/blog/lewins-change-model/>
15. Madumo, G., Marnewick, C., Nyandongo, K. M. (2025). A hybrid approach to manage IT projects. *International Journal of Agile Systems and Management*, 18 (2), 172–199. <https://doi.org/10.1504/ijasm.2025.145451>
16. Mikhnenko, P. A. (2025). Mathematical model and intelligent system for analyzing the intensity of megaproject changes: the role of temporary change management hubs. *Business Informatics*, 19 (2), 54–76. <https://doi.org/10.17323/2587-814x.2025.2.54.76>
17. Chen, S., Wang, C., Yan, K. (2024). Assessing Project Resilience Through Reference Class Forecasting and Radial Basis Function Neural Network. *Applied Sciences*, 14 (22), 10433. <https://doi.org/10.3390/app142210433>
18. Karam, B. A. E., Fissaa, T., Marghoubi, R. (2025). AI-Powered Assessment of Resistance to Change in the Context of Digital Transformation. *International Journal of Advanced Computer Science and Applications*, 16 (6). <https://doi.org/10.14569/ijacsa.2025.0160653>
19. Göçmen, I. S., Cezayir, A. S., Tüzün, E. (2025). Enhanced code reviews using pull request based change impact analysis. *Empirical Software Engineering*, 30 (3). <https://doi.org/10.1007/s10664-024-10600-2>
20. ISO/IEC/IEEE Standard No 15288:2015 (2015). Systems and software engineering – System life cycle processes. ISO/IEC/IEEE International Standard. <https://doi.org/10.1109/IEEESTD.2015.7106435>
21. Malhotra, V. (2020). *Single Reference Guide for Scrum Certification (Professional Scrum Master I (PSM I) and Professional Scrum Product Owner I (PSPO I) Certification)*. [Print Replica] Kindle Edition.
22. Larman, C., Vodde, B. (2016). *Large-Scale Scrum: More with LeSS*. Pearson Education.
23. Van Huffel, S. (Ed.) (1997). *Recent Advances in Total Least Squares Techniques and Errors-in-variables Modeling*. Society for Industrial & Applied, 377.

24. Diego, O., Essam, H. H., Salvador, H. (2021). Metaheuristics in Machine Learning Theory and Applications. Cham: Springer. <https://doi.org/10.1007/978-3-030-70542-8>
25. Yak vybraty pravylnyi tekhnolohichniy stek dlia vashoho proektu. Redstone. Available at: <https://redstone.agency/blog/yak-vybraty-pravylnyi-tekhnolohichniy-stek-dlia-vashoho-proektu/>
26. Doar, M. (2011). Practical JIRA Administration. O'Reilly Media, 140.
27. Kobylianskyi, L. S. (2002). Upravlinnia proektamy. Kyiv, 200.
28. Katrenko, A. V. (2011). Upravlinnia IT-proiektamy. Lviv: Novyi svit-2000, 552.
29. Guizzi, G., Fujita, H. (Ed.) (2023). New Trends in Software Methodologies, Tools and Techniques. IOS Press, 371. <https://doi.org/10.3233/faia371>
30. Roberts P. (2020). The Economist Guide to Change and Project Management. Economist Books, 448.

Scientific Route OÜ®

We invite you to explore our website.

www.route.ee
