

A. M. KOPP, D. L. ORLOVSKYI

CAPTURING SOFTWARE REQUIREMENTS FOR BUSINESS PROCESS MODEL ANALYSIS AND IMPROVEMENT

This paper considers the problem of software requirements capturing for business process model evaluation and recommendations generation to suggest how detected modeling errors may be eliminated in order to improve business process model correctness. Existing software tools for business process model analysis are based on metrics and thresholds that allow assume presence of errors in business process models, but they are not capable to recommend structural changes required to eliminate detected modeling errors. Therefore, business process model analysis and improvement tasks become relevant, since early detection and elimination of possible errors presented in business process models may allow organizations avoid extra costs that may occur on any of the steps of business process lifecycle. Since business process models are shared organizational assets, it is also necessary to consider the capabilities of collaborative and concurrent usage of business process models by multiple stakeholders. Hence, the workflow of business process model analysis and improvement was analyzed and its key steps, which describe expected user and software behavior, were formalized using the IDEF0 functional model. Functional software requirements were captured in the form of user stories according to modern agile practices of software development. These user stories are based on features and drawbacks of existing software tools for business process model analysis. Captured user stories were formalized using the use case modeling. Non-functional software requirements were also captured; they include maintainability, readability, testability, understandability, integrability, and complexity requirements. Captured requirements can be used to design and develop business process model analysis and improvement software capable to provide recommendations to eliminate detected modeling errors.

Keywords: business process model, model analysis, modeling errors, software requirements, software tool.

A. M. КОПП, Д. Л. ОРЛОВСКИЙ

ВИЗНАЧЕННЯ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ АНАЛІЗУ ТА ВДОСКОНАЛЕННЯ МОДЕЛЕЙ БІЗНЕС-ПРОЦЕСІВ

У даній роботі розглядається проблема визначення вимог до програмного забезпечення для аналізу моделей бізнес-процесів та формування рекомендацій щодо усунення виявлених помилок моделювання для підвищення коректності моделей бізнес-процесів. Існуючі програмні засоби аналізу моделей бізнес-процесів базуються на метриках та порогових значеннях, що дозволяють припустити про наявність помилок у моделях бізнес-процесів, проте не надають структурні зміни, необхідні для усунення виявлених помилок моделювання. Тому актуальною стає проблема аналізу та вдосконалення моделей бізнес-процесів, оскільки раннє виявлення та усунення можливих помилок, представлених у моделях бізнес-процесів, може дозволити організаціям уникнути зайвих витрат, які можуть виникнути на будь-якому з етапів життєвого циклу бізнес-процесів. Оскільки моделі бізнес-процесів є спільними організаційними ресурсами, також необхідно враховувати можливості спільного та одночасного використання моделей бізнес-процесів декількома зацікавленими сторонами. Отже, було проаналізовано робочий процес аналізу та вдосконалення моделей бізнес-процесів, а його основні етапи, що описують очікувану поведінку користувачів та програмного забезпечення, були формалізовані за допомогою функціональної моделі IDEF0. Функціональні вимоги до програмного забезпечення були відображені у вигляді користувацьких історій згідно із сучасними практиками гнучкої розробки програмного забезпечення. Дані користувацькі історії засновані на особливостях та недоліках існуючих програмних засобів для аналізу моделей бізнес-процесів. Визначені користувацькі історії були формалізовані за допомогою моделювання варіантів використання. Також були враховані нефункціональні вимоги до програмного забезпечення, які включають придатність до підтримки та тестування, читабельність, зрозумілість, інтегрованість та складність. Визначені вимоги можуть бути використані для проектування та розробки програмного забезпечення для аналізу та вдосконалення моделей бізнес-процесів, здатного пропонувати рекомендації з усунення виявлених помилок моделювання.

Ключові слова: модель бізнес-процесу, аналіз моделі, помилки моделювання, вимоги до програмного забезпечення, програмний засіб.

A. M. КОПП, Д. Л. ОРЛОВСКИЙ

ОПРЕДЕЛЕНИЕ ТРЕБОВАНИЙ К ПРОГРАММНОМУ ОБЕСПЕЧЕНИЮ ДЛЯ АНАЛИЗА И СОВЕРШЕНСТВОВАНИЯ МОДЕЛЕЙ БИЗНЕС-ПРОЦЕССОВ

В данной работе рассматривается проблема определения требований к программному обеспечению для анализа моделей бизнес-процессов и формирования рекомендаций по устранению выявленных ошибок моделирования и повышению корректности моделей бизнес-процессов. В основе существующих программных средств для анализа моделей бизнес-процессов лежат метрики и пороговые значения, которые позволяют предположить о наличии ошибок в моделях бизнес-процессов, но не предлагают структурные изменения, необходимые для устранения выявленных ошибок моделирования. Поэтому актуальной становится проблема анализа и совершенствования моделей бизнес-процессов, поскольку раннее выявление и устранение возможных ошибок, представленных в моделях бизнес-процессов, может позволить организациям избежать лишних расходов, которые могут возникнуть на любом из этапов жизненного цикла бизнес-процессов. Поскольку модели бизнес-процессов являются общими организационными ресурсами, также необходимо учитывать возможности совместного и одновременного использования моделей бизнес-процессов несколькими заинтересованными сторонами. Поэтому был проанализирован рабочий процесс анализа и совершенствования моделей бизнес-процессов, а его основные этапы, описывающие ожидаемое поведение пользователей и программного обеспечения, были формализованы при помощи функциональной модели IDEF0. Функциональные требования к программному обеспечению были отражены в виде пользовательских историй в соответствии с современными практиками гибкой разработки программного обеспечения. Данные пользовательские истории основаны на особенностях и недостатках существующих программных средств для анализа моделей бизнес-процессов. Определенные пользовательские истории были формализованы при помощи моделирования вариантов использования. Также были определены нефункциональные требования к программному обеспечению, которые включают пригодность к поддержке и тестированию, читабельность, понятность, интегрируемость и сложность. Определенные требования могут быть использованы для проектирования и разработки программного обеспечения для анализа и совершенствования моделей бизнес-процессов, способного предлагать рекомендации по устранению выявленных ошибок моделирования.

Ключевые слова: модель бизнес-процесса, анализ модели, ошибки моделирования, требования к программному обеспечению, программное средство.

Introduction. In the recent decades business process management (BPM) has become one of the most popular and commonly used disciplines, which combines together knowledge and experience from both management and IT (Information Technology) domains. Methods and tools of business process design, analysis, and automation are the baseline of BPM. A business process can be considered as the structured set of activities (or tasks) that takes different resources (materials, information, documents etc.) as input and produces products or services valuable for customers as output [1].

Since BPM combines expertise of managerial and IT disciplines, it has attracted a lot of attention because of its process management capabilities. These capabilities allow organizations to improve performance, reduce costs, and to shorten execution time of business processes. Business process modeling is the key technique of BPM. It is used to [2]:

- Document organizational activity (e.g. to provide process regulations for employees).
- Analyze organizational activity (e.g. to search for errors and measure performance).
- Improve described processes (e.g. by eliminating detected errors).

Hence, tasks of business process model analysis and improvement become relevant, since early detection and elimination of errors presented in business process models may allow organizations avoid extra costs that may occur on all the phases of business process lifecycle [3].

Research object includes the business process model analysis and improvement workflow.

Research subject includes software requirements for business process model analysis and improvement.

Research goal is to improve the IT used for business process model analysis and improvement by introducing the software requirements, which are based on the latest research in this field and are suitable for collaborative and concurrent usage of business process models by multiple stakeholders (involved business parties).

Related work. Business process models are special enterprise models that serve to reflect specific features of business process. Business process models describe the business process structure, which includes the set of all nodes (activities, events etc.) and arcs (sequence flows, data flows etc.) that belong to the business process graph of any complexity and topology. When modeling business processes, their topology should be considered in order to design as understandable models as possible, which reflect real scenarios of business processes. Also labels of nodes and arcs that belong to the business process graph, which display names of tasks and transitions, may be considered as well [4].

There are various kinds of business process models (e.g. activity models and behavioral models [5]) that allow describing business processes from different perspectives and that serve their own purposes.

According to the recent survey [6], BPMN (Business Process Model and Notation) is the most popular business process modeling notation, which is used by almost 64% respondents. Almost 18% of survey participants use EPC

(Event-driven Process Chain) notation that is also used to create behavioral models. As for activity modeling, IDEF0 and DFD (Data Flow Diagram) diagrams are used only by 4% of respondents.

Therefore, it is quite naturally that existing software solutions used for business process model analysis support BPMN or at least EPC (as extremely popular once in 90s and early 2000s) modeling notations.

Existing software tools are based on the metrics and corresponding thresholds in order to check the correctness of business process models. There are following software solutions:

- **BPMN Quality.** It is the stand-alone Java-based application for evaluation of BPMN models [7]. It contains the modules used to extract business process structure data from XML-like documents (BPMN uses interexchange file format based on XML syntax), to calculate metrics, and interpret obtained results by comparing metrics' values to threshold values. This software tool uses control flow complexity (CFC) metrics for evaluation of business process models [8].
- **BPMN Quality Tool.** It is the plug-in for BPMN modeling tool Business Process Visual Architect that is also developed using Java [9]. This plug-in provides calculation of process model metrics including the coefficient of network connectivity (CNC), CFC, and source code metrics [10].
- **Plug-ins for ProM (process mining software tool)** that provide calculation of density, connectivity, size, and other metrics for EPC models [11]. This tool only calculates metrics but the interpretation is user responsibility.
- **BEBoP (understandaBility vErifier for Business Process models).** It is another Java-based tool for BPMN models evaluation [12]. Unlike previous software tools, this is a web-based system, which may be integrated with other systems or may be extended. The analytical capabilities of this tool are based on process modeling guidelines [13].

Hence, all of the considered software tools are based on metrics and thresholds that allow assume presence of errors in business process models. But these software tools are not capable to recommend structural changes required to eliminate possible modeling errors.

Problem statement. The research problem includes requirements capturing in order to design the software for business process model evaluation and recommendations generation to suggest how detected modeling errors may be eliminated in order to improve business process model correctness. It is also necessary to consider the capabilities of collaborative and concurrent usage of business process models by multiple stakeholders.

Workflow analysis. Before functional requirements are captured, it is required to analyze the business process model analysis and improvement workflow.

Functional model (fig. 1) of this workflow has been designed using the IDEF0 notation.

It is shown that business process model analysis and improvement workflow includes the following steps:

- Model designer (business user) uploads a created business process model prepared using one of the available modeling notations (BPMN, EPC etc.).
- Uploaded business process models are processed by the software tool, as the result model metrics and detected errors are calculated and displayed.
- Required changes, based on modeling guidelines are generated by the software tool, are displayed for processed models in order to suggest the user how detected errors could be eliminated.

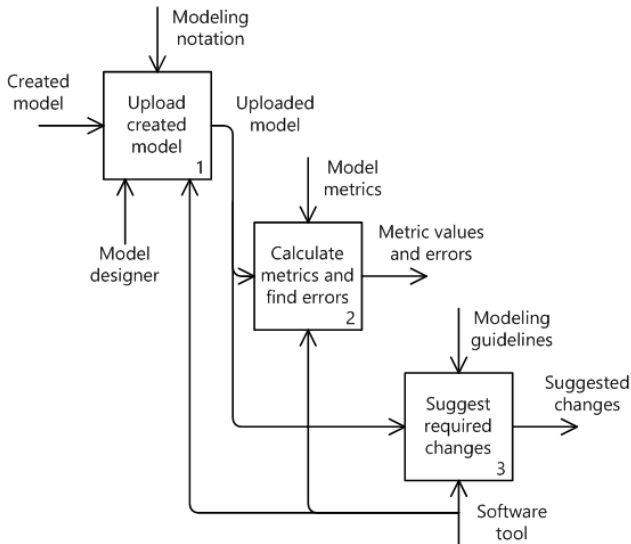


Fig. 1. Business process model analysis and improvement workflow

Captured user stories. Software requirements were captured with respect to the analyzed workflow. There are functional requirements presented as “user stories” widely used in agile methodologies [14]. Captured user stories are estimated using Fibonacci numbers: 1, 2, 3, 5, 8, and 13.

Users need to upload models for analysis (table 1).

Table 1 – Requirement to upload a new model

Title	FR01	Priority	8
User story	User	Model designer	
	Task	Upload a new model	
	Goal	Access a model for analysis	
Acceptance criteria	Context	User wants to upload a model	
	Action	User selects a file from disk	
	Outcomes	Model is appeared in a list of stored models	

Existing models need to be updated (table 2).

Table 2 – Requirement to update existing models

Title	FR02	Priority	8
User story	User	Model designer	
	Task	Update existing model	
	Goal	Track recent model changes	
Acceptance criteria	Context	User wants to update a model	
	Action	User selects stored model and picks new file from disk	
	Outcomes	File is updated for a stored model	

Also existing models need to be deleted (table 3).

Table 3 – Requirement to delete existing models

Title	FR03	Priority	3
User story	User	Model designer	
	Task	Delete existing model	
	Goal	Avoid outdated models	
Acceptance criteria	Context	User wants to delete a model	
	Action	User selects existing model to be deleted	
	Outcomes	Model is disappeared from a list of stored models	

Users should be able to perform analysis of existing models (table 4).

Table 4 – Requirement to analyze stored models

Title	FR04	Priority	13
User story	User	Model designer	
	Task	Analyze a model	
	Goal	Obtain model metrics and detect possible errors	
Acceptance criteria	Context	User wants to analyze a model	
	Action	User selects stored model to be analyzed	
	Outcomes	Metrics and detected errors are displayed and stored	

After a model is analyzed and metrics with possible detected errors are shown, users should be able to receive recommendations on how exactly detected errors could be eliminated (table 5).

Table 5 – Requirement to obtain recommendations for model improvement

Title	FR05	Priority	5
User story	User	Model designer	
	Task	Obtain recommendations for model improvement	
	Goal	Eliminate detected errors	
Acceptance criteria	Context	User wants to obtain recommendations	
	Action	User selects already analyzed model	
	Outcomes	Recommendations for model improvement are displayed	

Obtained analysis results should be downloadable as a stand-alone report (table 6).

Table 6 – Requirement to save model analysis report

Title	FR06	Priority	8
User story	User	Model designer	
	Task	Save model analysis report	
	Goal	Use analysis results any time	
Acceptance criteria	Context	User wants to save report	
	Action	User selects already analyzed model	
	Outcomes	Report file is downloaded	

Users should be able to login into a workspace that is personal for each user (table 7).

Table 7 – Requirement to login to a system

Title	FR07	Priority	5
User story	User	Model designer	
	Task	Login to a system	
	Goal	Work in a personal space	
Acceptance criteria	Context	User wants to login	
	Action	User inputs name and password	
	Outcomes	Workspace is opened or warning message is displayed	

Also users should be able to logout (table 8).

Table 8 – Requirement to logout of a system

Title	FR08	Priority	2
User story	User	Model designer	
	Task	Logout of a system	
	Goal	Prevent unauthorized access	
Acceptance criteria	Context	User wants to logout	
	Action	User clicks a logout button	
	Outcomes	Login form is displayed	

User stories considered above have been captured by analyzing features of existing software tools [7; 9; 11; 12]. It was taken into account that existing tools do not support multi-user access and do not suggest recommendations for business process model improvement.

As it is shown in tables above, captured user stories were estimated by their size (table 9).

Table 9 – Prioritized user stories

User story	Task	Priority
FR08	Logout of a system	2
FR03	Delete existing model	3
FR05	Obtain recommendations for model improvement	5
FR07	Login to a system	5
FR01	Upload a new model	8
FR02	Update existing model	8
FR06	Save model analysis report	8
FR04	Analyze a model	13

Use cases. Captured user stories demonstrate generic functional capabilities of business process model analysis and improvement software, which were formalized as use-cases outlined in fig. 2.

There use-cases of business process model analysis and improvement software are following:

- User logs in to a system.
- User uploads a business process model.
- User updates existing business process model.
- User deletes existing business process model.
- User calls for business process model analysis.
- User requests for recommendations to improve business process model correctness
- User downloads analysis report.
- User logouts of a system.

Non-functional requirements. Among the plethora of non-functional requirements we have captured:

- Maintainability. Any changes must be quick and cost efficient, so all classes and methods must be documented. Uncommented code units must not exceed 20% of all units.
- Readability. Developers need to follow declared indents and keep naming conventions to provide a consistent code style. Source code that violates established rules must not exceed 10% of KLOC (kilo lines of code).
- Testability. Maintainers should be able to detect and fix defects. Thus, unit tests must ensure 90% coverage.
- Understandability. User interface (UI) should be easy for perception: UI elements must not exceed 7 items per screen.
- Integrability. The system must accept BPMN 2.0 exchange file format to be integrated with almost any modern business process modeling suites. It must be possible to integrate with at least one of version-control systems (Subversion, Git etc.) to ensure collaborative work on process models.
- Complexity. It highly affects maintainability and readability. Hence, no class method may exceed 100 LOC (lines of code).

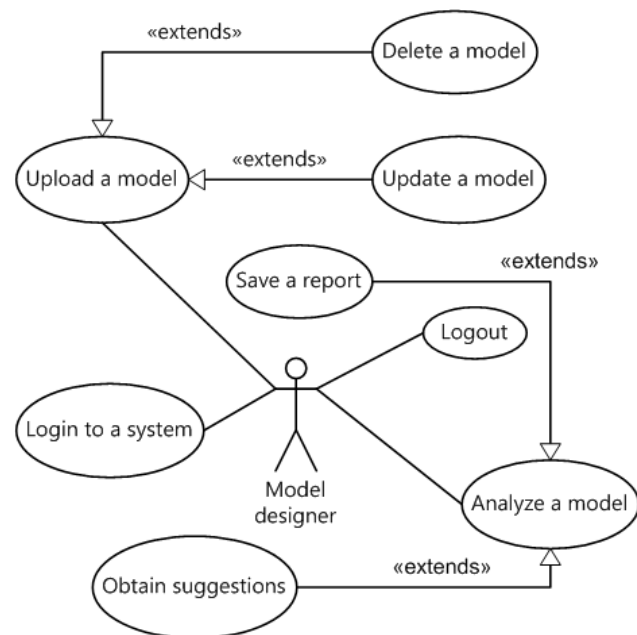


Fig. 2. Functional capabilities of the software tool

Defined non-functional requirements are measurable: integrability constraint is binary, while others are based on threshold values.

Conclusion. The problem of requirements capturing for the business process model analysis and improvement software has been considered. Captured requirements are based on features and lacks of existing software tools, but ensure recommendations generation in order to eliminate detected process modeling errors. While current software tools are mostly desktop or even plug-ins, when capturing requirement it was considered possibility of collaborative

and concurrent usage of multiple stakeholders (designers, business analysts etc.) via integration with version-control systems. A software build using elicited requirements may help organizations to reduce possible expenses caused by fault business process models.

Future research in this field includes software design and prototyping according to captured requirements.

References

1. Hammer M., Champy J. *Reengineering the Corporation: A Manifesto for Business Revolution*. Zondervan, 2009. 272 p.
2. Van der Aalst W. M. P. *Business process management: a comprehensive survey*. Hindawi Publishing Corporation: ISRN Software Engineering, 2013. 37 p.
3. Mendling J., Sanchez-Gonzalez L., Garcia F., La Rosa M. Thresholds for error probability measures of business process models. *Journal of Systems and Software*. 2012. Vol. 5, no. 85. P. 1188–1197.
4. Dijkman R., Dumas M., Garcia-Banuelos L., Kaarik R. Aligning business process models. *IEEE International Enterprise Distributed Object Computing Conference*. 2009. P. 45–53.
5. Kalpic B., Bernus P. Business process modeling through the knowledge management perspective. *Journal of Knowledge Management*. 2006. Vol. 10, no. 3. P. 40–56.
6. Harmon P. *The State of Business Process Management 2016*. BPTrends, 2016. 50 p.
7. Makni L. et al. A tool for evaluation the quality of business process models. *INFORMATIK 2010 Business Process and Service Science – Proceedings of ISSS and BPSC*. 2010. P. 230–242.
8. Kahloun F., Channouchi S. A. Quality criteria and metrics for business process models in higher education domain: case of a tracking of curriculum offers process. *Procedia Computer Science*. 2016. Vol. 100. P. 1016–1023.
9. Sadowska M. An approach to assessing the quality of business process models expressed in BPMN. *E-Informatica Software Engineering Journal*. 2015. Vol. 9, no. 1. P. 57–77.
10. Mendling J. *Detection and prediction of errors in EPC business process models*. Wirtschaftsuniversität Wien. Vienna, 2007. 525 p.
11. Vanderfeesten I. et al. Quality metrics for business process models. *BPM and Workflow handbook*. 2007. Vol. 144. P. 179–190.
12. Corradini F. et al. A guidelines framework for understandable BPMN models. *Data & Knowledge Engineering*. 2018. Vol. 113. P. 129–154.
13. Mendling J. et al. Seven process modeling guidelines (7PMG). *Information and Software Technology*. 2010. Vol. 52. P. 127–136.
14. Rubin K. S. *Essential Scrum: A Practical Guide to the Most Popular Agile Process*. Addison-Wesley Professional, 2012. 452 p.

References (transliterated)

1. Hammer M., Champy J. *Reengineering the Corporation: A Manifesto for Business Revolution*. Zondervan, 2009. 272 p.
2. Van der Aalst W. M. P. *Business process management: a comprehensive survey*. Hindawi Publishing Corporation: ISRN Software Engineering, 2013. 37 p.
3. Mendling J., Sanchez-Gonzalez L., Garcia F., La Rosa M. Thresholds for error probability measures of business process models. *Journal of Systems and Software*. 2012, vol. 5, no. 85, pp. 1188–1197.
4. Dijkman R., Dumas M., Garcia-Banuelos L., Kaarik R. Aligning business process models. *IEEE International Enterprise Distributed Object Computing Conference*. 2009, pp. 45–53.
5. Kalpic B., Bernus P. Business process modeling through the knowledge management perspective. *Journal of Knowledge Management*. 2006, vol. 10, no. 3, pp. 40–56.
6. Harmon P. *The State of Business Process Management 2016*. BPTrends, 2016. 50 p.
7. Makni L. et al. A tool for evaluation the quality of business process models. *INFORMATIK 2010 Business Process and Service Science – Proceedings of ISSS and BPSC*. 2010, pp. 230–242.
8. Kahloun F., Channouchi S. A. Quality criteria and metrics for business process models in higher education domain: case of a tracking of curriculum offers process. *Procedia Computer Science*. 2016, vol. 100, pp. 1016–1023.
9. Sadowska M. An approach to assessing the quality of business process models expressed in BPMN. *E-Informatica Software Engineering Journal*. 2015, vol. 9, no. 1, pp. 57–77.
10. Mendling J. *Detection and prediction of errors in EPC business process models*. Wirtschaftsuniversität Wien. Vienna, 2007. 525 p.
11. Vanderfeesten I. et al. Quality metrics for business process models. *BPM and Workflow handbook*. 2007, vol. 144, pp. 179–190.
12. Corradini F. et al. A guidelines framework for understandable BPMN models. *Data & Knowledge Engineering*. 2018, vol. 113, pp. 129–154.
13. Mendling J. et al. Seven process modeling guidelines (7PMG). *Information and Software Technology*. 2010, vol. 52, pp. 127–136.
14. Rubin K. S. *Essential Scrum: A Practical Guide to the Most Popular Agile Process*. Addison-Wesley Professional, 2012. 452 p.

Received 05.09.2020

Відомості про авторів / Сведения об авторах / About the Authors

Копп Андрій Михайлович – Національний технічний університет «Харківський політехнічний інститут», старший викладач кафедри програмної інженерії та інформаційних технологій управління; м. Харків, Україна; ORCID: <http://orcid.org/0000-0002-3189-5623>; e-mail: kopp93@gmail.com

Орловський Дмитро Леонідович – кандидат технічних наук, доцент, Національний технічний університет «Харківський політехнічний інститут», доцент кафедри програмної інженерії та інформаційних технологій управління; м. Харків, Україна; ORCID: <http://orcid.org/0000-0002-8261-2988>; e-mail: ordm@kpi.kharkov.ua

Копп Андрей Михайлович – Национальный технический университет «Харьковский политехнический институт», старший преподаватель кафедры программной инженерии и информационных технологий управления; г. Харьков, Украина; ORCID: <http://orcid.org/0000-0002-3189-5623>; e-mail: kopp93@gmail.com

Орловский Дмитрий Леонидович – кандидат технических наук, доцент, Национальный технический университет «Харьковский политехнический институт», доцент кафедры программной инженерии и информационных технологий управления; г. Харьков, Украина; ORCID: <http://orcid.org/0000-0002-8261-2988>; e-mail: ordm@kpi.kharkov.ua

Kopp Andrii Mykhailovych – National technical university «Kharkiv polytechnic institute», Senior Lecturer of the Department of Software Engineering and Information Technology Management; Kharkiv, Ukraine; ORCID: <http://orcid.org/0000-0002-3189-5623>; e-mail: kopp93@gmail.com

Orlovskiy Dmytro Leonidovych – PhD in Technical Sciences, Docent, National technical university «Kharkiv polytechnic institute», Associate Professor of the Department of Software Engineering and Management Information Technology; Kharkiv, Ukraine; ORCID: <http://orcid.org/0000-0002-8261-2988>; e-mail: ordm@kpi.kharkov.ua