

Anastasiia STRIELKINA¹, Artem TETSKYI²

¹ Software Development Hub LLC, Kharkiv, Ukraine

² National Aerospace University “Kharkiv Aviation Institute”, Kharkiv, Ukraine

METHODOLOGY FOR ASSESSING SATISFACTION WITH REQUIREMENTS AT THE EARLY STAGES OF THE SOFTWARE DEVELOPMENT PROCESS

The **subject matter** of the research is the process of satisfaction with requirements during software development. A qualitative requirements engineering stage for the system being designed to fulfill all business goals, please the client, and eventually satisfy the end user, is one of the key prerequisites for effective implementation of any IT project. The level of satisfaction with requirements must rise as a prerequisite for the project's success through requirement engineering. To ensure that a product or service meets the needs and expectations of its users or consumers, it is critical to satisfy these requirements. The primary **purpose** of the proposed study is to establish a methodology for quantitatively assessing the satisfaction with requirement level considering various characteristics of requirements before the development phase begins. The **tasks** to be solved are: to investigate the up-to-date state of the subject area; to develop a methodology for assessing satisfaction with requirements; to provide and investigate the proposed methodology on the real-life example; to recommend actions to increase the level of satisfaction with requirements. The suggested **methodology**, as opposed to others, considers such characteristics of the requirements as atomic, completeness, consistentness, conciseness, feasibility, unambiguousness, testability, prioritized, understandability, security, and performance to obtain a quantitative assessment of satisfaction with requirements level. The **result** of this paper is a methodology for quantitative assessing the satisfaction with requirements considering different characteristics of requirements before the development phase begins. This study is significant and necessary since, in the majority of cases, previous research does not offer comprehensive quantitative and measurable methods for determining the degree to which requirements for certain characteristics are satisfied. Also, it is demonstrated how the created methodology may be used with actual requirements. There are additionally recommendations for strengthening satisfaction with requirements. **Conclusions.** The proposed methodology is extensible, unlike others, which means that the characteristics and rating scale can actually change depending on the requirements, goals, and other features of the IT project.

Keywords: requirements; software development; satisfaction with requirements; requirements engineering; requirements characteristics.

1. Introduction

In the increasingly competitive business world nowadays, businesses benefit greatly if they can rapidly deliver software of foreseeable quality and performance on a minimal budget. However, not all projects and businesses are capable of handling this.

1.1. Motivation

A study by the Standish Group found [1] that approximately 28 % of IT projects are considered failures because they are either canceled before completion or do not deliver the expected benefits. In fact, a study by the Project Management Institute [2] found that poor requirement management was the most frequently cited factor contributing to project failure, with 46 % of surveyed organizations citing it as a top challenge. Other studies have also identified poor requirement management as a leading cause of project failure.

Some common issues with requirements include:

1. Lack of clarity or precision: Requirements that are poorly defined, ambiguous, or contradictory can lead to misunderstandings and miscommunications, which can lead to project delays or failures.

2. Lack of completeness: Incomplete requirements can lead to scope creep (i.e., the addition of new features or functionality that was not originally included in the project plan) or the development of features that do not meet the needs of the users.

3. Lack of traceability: Without traceability, it can be difficult to ensure that all requirements have been addressed in the final product, which can lead to defects or user dissatisfaction.

4. Changing requirements: If requirements change frequently or significantly during a project, it can be difficult to manage the project effectively and deliver a high-quality product on time.

Hereby, one of the main conditions of successful implementation of any IT project is a qualitative requirement engineering stage for the system being developed to

meet all the business needs and satisfy the customer and eventually the end user [3].

Also, it should be emphasized that without the high-grade requirements engineering process may occur an expectation gap, i.e. the gap between what is really needed to be developed and what the developing team provided based on what they knew at the starting stages of the project [4]. Although most of the construction costs are allocated for the planning stage. However, a majority of these costs are reflected in the final construction phase. During the planning phase, the savings are the greatest [5]. The influence on costs during project development is shown in Fig. 1.

A condition for the success of the project through requirement engineering is to increase the level of satisfaction with requirements. Satisfaction with requirements refers to the degree to which a product or system meets the needs and expectations of its stakeholders. In the context of business stakeholders, satisfaction with requirements refers to the extent to which the software being developed meets the functional and non-functional requirements specified by the client or user. And in the context of a developing team, satisfaction belongs to how the requirements are clear, understandable, implementable by them.

Satisfaction with requirements is crucial because it helps ensure that a product or service meets the needs and expectations of its users or customers. When requirements are not satisfied, it can lead to a number of problems, such as:

1. Poor quality. If requirements are not satisfied, the product or service may not function as intended, which can lead to a poor user experience and customer dissatisfaction.
2. Increased costs. If requirements are not identified and addressed during the development process, it can lead to rework and additional costs as problems are identified and addressed later on.

3. Delays. If requirements are not properly defined or understood, it can lead to delays in the development process as issues are identified and addressed.

4. Reduced competitiveness. If a product or service does not meet the needs and expectations of its users or customers, it may be less competitive in the market.

Overall, satisfaction with requirements is important because it helps ensure that a product or service is of high quality, meets the needs of its users or customers, and is delivered on time and within budget.

1.2. Related works analysis

In a comprehensive study of the importance of requirement engineering in ensuring the success of a project [6], it was investigated the various methods and approaches to requirement engineering, highlighting the benefits and drawbacks of each. The authors begin by discussing the various challenges that can arise when requirements are not properly defined and managed, such as scope creep, misunderstandings, and project delays. They also present a number of case studies and real-world examples to illustrate the importance of proper requirement engineering in practice. They identify key themes and findings from this literature, including the need for effective communication and collaboration among team members, the importance of involving stakeholders early in the process, and the value of using various requirements elicitation techniques to ensure that all necessary requirements are captured.

Only a few studies in the literature demonstrate approaches for satisfaction with requirements.

The authors of the study [7] provided a comprehensive and well-organized review of the existing literature devoted to the software satisfaction with requirements. The authors conducted a thorough search of the literature and analyzed a large number of studies to identify key themes and trends in the field. One of the key strengths

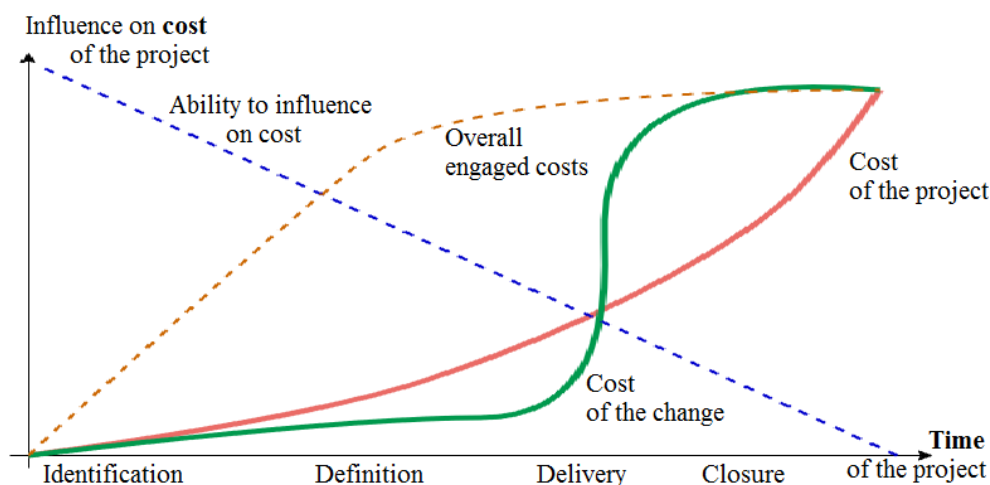


Fig. 1. Influence on costs during project implementation

of this review is the authors' use of a systematic approach to identifying and selecting relevant studies. This ensures that the review is thorough and covers a wide range of research in the field. Additionally, the authors provide a detailed summary of each study, highlighting the key findings and implications for practice. One area where the review could be improved is in the discussion of the limitations of the studies included in the review. While the authors do discuss some of the limitations of individual studies, a more in-depth analysis of the overall limitations of the literature as a whole could have been helpful.

The authors of [8] have conducted a thorough study comparing the satisfaction with requirements in traditional and Agile software development methodologies. One of the key study strengths is its use of a large sample size, with data collected from over 200 software development projects. This allows for a high level of statistical significance and generalizability of the findings. The authors also use various measures to assess satisfaction with requirements, including both subjective and objective measures, which add to the robustness of the results. The findings of the study suggest that agile software development methodologies tend to result in higher levels of satisfaction with requirements compared with traditional methodologies. This is an important finding, as satisfaction with requirements is a key factor in the success of a software development project. The authors also provide insight into the specific factors that contribute to higher levels of satisfaction with requirements in agile development, such as frequent communication and collaboration with stakeholders, and the ability to quickly adapt to changes in requirement.

A well-written and comprehensive review of the current state of research on satisfaction with requirements in software engineering is provided in [9]. The authors provide a thorough overview of the various approaches and techniques used to ensure that software requirements are satisfied during the development process. The strength of this review is its focus on practical implications for software engineers. The authors not only summarize the existing research, but also provide recommendations and best practices for improving satisfaction with requirements in software projects.

The authors of [10] did an excellent job of explaining the different methods and providing examples to illustrate their points on a comprehensive review of the various approaches used to ensure that the requirements of a system are satisfied. One of the standout features of this paper is the authors' ability to clearly articulate the benefits and limitations of each approach. They do a thorough job of presenting the pros and cons of each method, which makes it easy for readers to understand the trade-offs in choosing a particular approach. Additionally, the authors provide a useful summary of

the state of the art in satisfaction with requirements approaches and highlight areas where further research is needed. This is particularly useful for those looking to stay up to date on the latest developments in this field.

In the paper [11], the authors provide a comprehensive overview of the various approaches and techniques used to ensure satisfaction with requirements in software development. The authors cover a wide range of approaches and techniques, including traditional techniques such as requirement elicitation and analysis, as well as more recent approaches such as agile methodologies and model-driven development. They also discuss the challenges and issues that can arise when trying to ensure satisfaction with requirements, and provide recommendations for addressing these challenges.

The authors of the study [12] conducted a systematic literature review of the existing research on measuring satisfaction with requirements in agile software development. They identified a number of challenges, including the lack of a standard definition of satisfaction with requirements, the difficulty of tracking and measuring satisfaction over time, and the need for appropriate metrics and methods for assessing satisfaction. The paper also discusses various approaches to measuring satisfaction with requirements, including subjective measures (such as surveys and interviews), objective measures (such as defect density and performance metrics), and mixed methods (combining subjective and objective measures). Overall, the paper concludes that while there is a lack of consensus on the best approach to measuring satisfaction with requirements in agile software development, it is important to consider both subjective and objective measures and to use a combination of methods to provide a more comprehensive view of satisfaction.

The authors of the study [13] examine the relationship between satisfaction with requirements and project success in agile software development. The authors begin by discussing the importance of satisfaction with requirements in software development, highlighting how it is a key factor in determining the success of a project. Then they describe the agile software development approach, which emphasizes flexibility and adaptability in the development process. The authors then present the results of their study, which involved surveying a sample of agile software development projects to examine the relationship between satisfaction with requirements and project success. The results of the study showed a strong positive correlation between satisfaction with requirements and project success, with projects that had higher levels of satisfaction with requirements also experiencing higher levels of success.

Satisfaction with requirements should be an integral property in the creation of any project. But it should be especially important for critical areas of activity, in particular for the cybersecurity of IoT systems, industrial

and web-oriented systems and networks [14, 15].

Seven various techniques of prioritizing security requirements were found, summarized, and compared in [16] study. Eliciting, assessing, and documenting security needs are all part of Security requirements Engineering. Although all security needs are taken into consideration, it is impossible to implement all security measures that offer protection from all potential threats. In addition to competing with time and money, security standards must contend with the limitations they place on a software's accessibility, features, and functionality. As a result, the process of prioritizing security needs becomes a crucial effort in the field of risk and trade-off analysis. Software developers may make informed choices about which security needs are most important by using a sound prioritizing approach. Although earlier research has suggested various security need prioritizing strategies, no research effort has yet offered a thorough assessment and comparative analysis of the available techniques.

Agile development frequently uses user stories as requirements. However, very few studies have evaluated the effectiveness of user stories in the real world. The authors of [17] carried out a case study. They referenced an analysis report from a real-world project where an organization sought to upgrade or purchase a new hotline system; and asked IT professionals to draft the new system's requirements based on the analysis report, user stories, and anything else they deemed important. The completeness, accuracy, verifiability, and traceability criteria from IEEE 830 [18] were used to evaluate the practitioners' responses. Additionally, key objectives such as learnability and maintainability that are crucial to most projects were completely overlooked in the responses. The practitioners frequently suggested incorrect or constrictive treatments. Most responses featured user stories that were difficult to verify or that, if implemented separately, would have resulted in a complicated user interface. Relying on the user stories in this project would have been disastrous. Even if the user stories may have been better, they still would not address all of the project's requirements.

The innovative nature of projects, the complexity of forecasting external and internal conditions, the impact of the human factor, and the increasing frequency of changes in product demand all contribute to increased uncertainty and the inability to plan team activities with a certain degree of accuracy. In this sense, the challenge of forming an adaptable project team that can function well under the aforementioned circumstances is the subject matter of the article [19]. This duty pertains particularly to the field of software development. The study formalizes the process of choosing the members of the software development team, considering the subjectivity and ambiguity of the data that influences the choice of candidates for the team. The goal of this effort was to develop

an operation research-based decision-making model using fuzzy sets as a mathematical tool. The degree of expertise of team applicants should be considered, as well as the uncertainty of estimations of the project needs. The final result is a mathematical model of an optimization problem with two constraints as finding a team composition and to create a team that has the greatest total of skills for all indicators. Making a team that can quickly fulfill the new and current needs for the project personnel will be made feasible by solving the problem in line with the provided mathematical model.

Financed new items that are in demand in the high-tech product marketplaces are essential to the growth of businesses in crucial industries. Investors' interest in a company is influenced by how innovative and competitive its products are. To obtain capital from prospective investors, the company should create a fresh, broad portfolio of orders. The study [20] creates a strategy to support investments in new orders that are founded on the investigation of the complicated product's component architecture. The problems of justifying and choosing a diverse portfolio of orders, simulating and evaluating the viability of a portfolio of orders, and analyzing the product component architectural innovation and investment attractiveness are presented and resolved. The component technique that the study suggests allows for evaluating the new product's design in terms of innovation and investment attractiveness. The suggested multifactor design of the experiment considers all feasible possibilities and evaluates the key parameters of the new product, including investment appeal, prices, delivery schedules, and order fulfillment hazards. To identify the best choice given the enterprise's constrained capabilities, lexicographic ordering of the possibilities is used.

There are several quantitative approaches that can be used to assess the satisfaction with requirements before starting the development of a product or service [21]. Here are a few examples:

1. Rating scales: This approach involves rating each requirement on a scale, such as a 1-5 scale, to indicate the level of satisfaction.
2. Surveys: Surveys can be used to gather data on the satisfaction with requirements from stakeholders. This can be done through online surveys or by distributing paper surveys.
3. Customer satisfaction index (CSI): The CSI is a measure of customer satisfaction with a product or service. It can be used to assess the satisfaction with requirements for a particular project or product.
4. Requirements traceability matrix (RTM): An RTM is a tool that helps track the progress of requirements throughout the development process. It can be used to assess the satisfaction with requirements by tracking whether they have been implemented and tested.

5. Requirements prioritization: Prioritizing requirements can help determine which ones are most important to stakeholders and therefore should be given the highest level of satisfaction.

6. User testing: User testing can be used to gather data on the satisfaction with requirements from actual users of the product or service. This can involve conducting usability tests or focus groups.

1.3. Objectives and Structure

Unfortunately, after analyzing the different studies, it could be stated that the literature available does not provide in most cases complex quantitative and measurable approaches to assessing satisfaction with requirement level for different characteristics, which makes this study relevant and needed.

It is proposed to use such characteristics to determine whether a set of requirements has been satisfied in [21]:

1. Atomic: stand-alone and capable of being understood regardless of other requirements or designs.

2. Complete: sufficient to direct subsequent development and at the relevant level of detail to continue the work.

3. Consistent: the requirements do not conflict with each other or with the overall system design.

4. Concise: excludes unnecessary and irrelevant content.

5. Feasible: the requirements can be implemented within the given constraints (e.g. time, budget, resources).

6. Unambiguous: clearly stated in such a way that it is obvious whether the solution meets the corresponding need.

7. Testable: The requirements can be easily tested to ensure they have been satisfied.

8. Prioritized: ranked, grouped, or aligned in terms of importance and value compared to all requirements.

9. Understandable: presented using common audience terminology.

The authors suggest that for the completeness of the assessment also use such characteristics as:

10. Security: The requirement results in a secure system and protects sensitive data.

11. Performance: The requirement results in a system that meets the required performance criteria (e.g. response time, throughput).

Thus, the development of a methodology for quantitative assessing the satisfaction with requirements considering different characteristics of requirements before the development phase begins is important, as this paper aims to achieve.

The paper is structured as follows. The paper begins with the inducement of the research and current state-of-

the-art. Next, the methodology itself for quantitative assessing the satisfaction with requirements considering different characteristics of requirements before the development phase begins is proposed. An example of using the developed methodology with real requirements is presented. Recommendations for increasing satisfaction with requirements are also provided. The last section concludes and discusses future research step.

2. Description of the Methodology

The methodology for assessing satisfaction with requirement level refers to the systematic process that is followed to ensure that the requirements of a project are accurately identified, documented, and fulfilled. This process is crucial for the success of any project, as it helps ensure that the final product meets the needs of the stakeholders and delivers the desired outcomes.

There are several main steps involved in the methodology for assessing satisfaction with requirement level, which can actually vary depending on the specific needs and goals of the project. Some of the key steps in this process include:

Step 1. *Determination of system components to be developed.* At this step, a list of all identified system components to be developed is compiled:

$$C = \left\{ \bigcup_{j=1}^n C_j \right\} = \{C_1, C_2, \dots, C_n\}, \quad (1)$$

where $C_j \subseteq C, (j = \overline{1, n})$ is a component of the system, n is a total number of components.

By necessity, the components may be divided into subcomponents, and each subcomponent, in turn, into elements, i.e. the partitioning of the components is conducted.

Step 2. *Identification of requirements for each system component.* For each determined system component C_j the requirements shall be elicited:

$$R = \left\{ \bigcup_{i=1}^m R_i \right\} = \{R_1, R_2, \dots, R_m\}, \quad (2)$$

where $R_i (i = \overline{1, m})$ is a requirement, m is a total number of requirements.

Step 3. *Documentation of the requirements.* The identified requirements should be documented in a clear and concise manner, using appropriate tools and techniques such as user stories, use cases, and requirements traceability matrices.

Step 4. *Assessment of each requirement.* Each identified and documented requirement should be assessed. The requirements shall be assessed for these characteristics of requirements and design quality as atomic, completeness, consistentness, conciseness, feasibility, unambiguously, testability, prioritization, understandability, security, and performance.

In this study, it is proposed to assess each indicator by 0 or 1 point, where 1 – the requirement does not match the characteristic, and 2 – the requirement matches the characteristic. Such an assessment should be conducted for each identified requirement R_i for each component C_j . the number of points could be modified using other quantitative approaches described before.

Also, it is worth mentioning that the points should be given by the expert group. The expert group should include representatives of both internal and external stakeholders.

Step 5. *Assessment of the level of satisfaction with requirements.* For each assessed requirement, the assessment of the level of satisfaction with the requirement shall be conducted. In this paper, the level of satisfaction with requirement is considered as

$$S_i = \sum_{l=1}^L L_{li}, \quad (3)$$

where L_l ($l = \overline{1, 11}$) are corresponding characteristics assessments R_i : L_{1i} is an indicator of atomic characteristic; L_{2i} is an indicator of completeness characteristic; L_{3i} is an indicator of consistentness characteristic; L_{4i} is an indicator of conciseness characteristic; L_{5i} is an indicator of feasibility characteristic; L_{6i} is an indicator of unambiguously characteristic; L_{7i} is an indicator of testability characteristic; L_{8i} is an indicator of prioritization characteristic; and L_{9i} is an indicator of understandability characteristic; L_{10i} is an indicator of security characteristic; and L_{11i} is an indicator of performance characteristic.

Step 6. *Determination of the boundary and recommended levels of requirements for the component to be developed.* The boundary (acceptable, admissible) level of satisfaction with requirements for the component to be developed it is determined as follows:

$$S_b = k \cdot \max \left(\sum_{l=1}^L S_{il} \right), \quad (4)$$

where $\max \left(\sum_{l=1}^L S_{il} \right)$ is the maximum sum of values of

all system characteristics; and k is an acceptability factor, and $0 < k \leq 1$. It should be noticed that the higher the level of system/subsystem/component criticality is, the greater k value should be used.

Accordingly, the boundary satisfaction level for specific requirements changes in this paper from $S_{\min \text{ res}} = 0$ to $S_{\max \text{ res}} = 11$.

The recommended value of satisfaction with requirements S_0 shall be set, it also shall vary from $S_{\min \text{ res}} = 0$ to $S_{\max \text{ res}} = 11$.

Also, it is worth mentioning that an acceptability factor and recommended levels of satisfaction with requirements should be given by the expert group.

Step 7. *Determination of the result satisfaction with requirement level for the component to be developed.* To avoid possible risks, it is proposed to choose the minimum value of all obtained S_i , i.e. $S_{\text{res}} = \min(S_1, S_2, \dots, S_i)$ as a resultant assessment. The satisfaction level S_{res} shall be determined as $\text{satisfaction}(S_{\text{res}}) = \{\text{High, Medium, Low}\}$, where

$$\text{satisfaction}(S_{\text{res}}) = \begin{cases} \text{High, when } S_i > S_b \\ \text{Medium, when } S_0 < S_i \leq S_b \\ \text{Low, when } S_i \leq S_0 \end{cases}, \quad (5)$$

where if $S_i > S_b$, then S_i is considered as satisfied (High), therefore the requirements do not need to be refined; if $S_0 < S_i \leq S_b$, then corrective measures are necessary to increase level of satisfaction (Middle); if $S_i \leq S_0$, then S_i is significantly low and requires mandatory elaboration (Low).

Step 8. *Result generation and report preparation.* The last step provides data systematization. At this stage, the data obtained at the previous step are systematized and visualized to assess the level of satisfaction with requirements. The step involves the systematization of all information in the form as an example of a Table 1.

Consequently, this table systematically combines all before mentioned steps in one convenient place.

3. Case: An Example of using the presented approach

To demonstrate the presented methodology of requirement level satisfaction assessment the requirements to signing-up (registration) feature of an under NDA project was chosen. This feature is the first to be developed,

Table 1

C _j	R _i	Characteristics											k	S _b	S ₀	S _i
		L _{1i}	L _{2i}	L _{3i}	L _{4i}	L _{5i}	L _{6i}	L _{7i}	L _{8i}	L _{9i}	L _{10i}	L _{11i}				
Result:																S _{res}
Result:																S _{res}

Since all characteristics of the signing-up feature are crucial from the perspective of business stakeholders, the value is set to 0.9 by stakeholders who are conducting the assessment, so $S_b = S_0 = 10$.

Table 2 presents results of assessment by an expert group of stakeholders of the project.

Reasons why these assessments were made by the stakeholders expert group:

1. R₁ – the feature should include requirements only for registration, and the initial includes the login as well; the initial requirement is implicit and could be taken in different ways, it is necessary to avoid any ambiguity; the requirement does not comply with the accepted rules for writing requirements.

2. R₂— it is immediately clear that these requirements from a security and privacy viewpoint should not apply to the average user, and most likely relate to another feature; also without specifying details can lead to performance problems; the requirement does not comply with the accepted rules for writing requirements.

Table 2

C _j	R _i	Characteristics											k	S _b	S ₀	S _i
		L _{1i}	L _{2i}	L _{3i}	L _{4i}	L _{5i}	L _{6i}	L _{7i}	L _{8i}	L _{9i}	L _{10i}	L _{11i}				
1	1	0	0	1	1	1	0	0	1	1	1	1	0.9	10	10	7
	2	1	1	0	1	0	1	1	0	0	0	0				5
	3	1	0	1	0	1	0	0	1	0	1	1				6
	4	1	1	1	0	1	0	0	1	0	1	1				7
	5	1	1	1	0	1	1	1	1	1	1	1				10
Result:																5

3. R_3 – the initial requirement does not require the development of additional functionality; in this case the requirement can be combined with requirement R_1 and can be added as an additional acceptance criteria (e.g. a list of allowed business domains); there is also a question in the description that makes it unclear whether to implement it or not – the requirement does not comply with the accepted rules for writing requirements.

4. R_4 – the initial requirement leads to the development of another feature for registration with a user's Google account; the requirement does not comply with the accepted rules for writing requirements.

5. R_5 – the initial requirement includes technical details, which can lead to narrowing the perspective of the developing team (designers, developers, etc.).

The obtained result shows that the requirements do not meet the set L_b and the satisfaction level is equal to Low. In such a case, after the made assessment and the resultant assessment is obtained, it is needed to redefine the requirements. The list of requirements redefined after the assessment is the following:

1. Requirement 1 (R_1): As a User, I want to sign up via my email and password to the system, so that I can have access to my account.

2. Requirement 4 (R_4): As a User, I want to sign up via Google to the system, so that I can have access to my account.

The updated requirements provide a more correct description of the required functionality and satisfy the stakeholder at the appropriate level.

4. Recommendations to increase the level of satisfaction with requirements

Satisfaction with requirements is important in software development because clear and accurate requirements serve as the foundation for a successful project. When requirements are well-defined, it is easier for developers to understand what needs to be built, which can lead to more efficient development and a higher quality end product. On the other hand, if requirements are unclear or incomplete, it can lead to misunderstandings, rework, and delays, which can negatively impact the overall satisfaction with the project.

There are several ways to increase satisfaction with requirements in a software development project:

1. Involve stakeholders early and often in the requirements gathering process. Engaging stakeholders from the beginning will help ensure that their needs and concerns are fully understood and incorporated into the requirements.

2. Use clear, concise language in requirements. Ambiguity and inconsistency can lead to misunderstandings and frustration. Using a clear and concise language can help reduce the likelihood of these issues.

3. Use of visualization techniques. Visualizing requirements using techniques such as user stories, wireframes, and prototypes can help stakeholders better understand the requirements and provide more meaningful feedback.

4. Use user-centered design principles. By focusing on the needs and preferences of the end users, you can create requirements that are more likely to be satisfactory to them.

5. Prioritize requirements. It is not always possible to meet all requirements, so it's important to prioritize them based on their importance and the resources available. This can help ensure that the most important requirements are addressed first.

6. Regularly review and update requirements. As the project progresses, it is important to review and update the requirements to ensure they are still relevant and accurate. This can help prevent misunderstandings and keep the project on track.

5. Discussion

The results of this study show the importance of considering various characteristics of requirements to assess the level of satisfaction with requirements. The methodology proposed in this study is unique in that it considers a number of important characteristics, such as atomic, completeness, consistency, conciseness, feasibility, unambiguousness, testability, prioritized, understandability, security, and performance. This comprehensive approach is a significant contribution to the field of satisfaction with requirements in software development.

However, the methodology is still in its early stages and may need further refinement and improvement. For example, the characteristics and rating scale used in the methodology may need to be adjusted based on the basis of the specific requirements and goals of each project. Additionally, there is a need for further research to investigate the interplay between different characteristics and to develop more robust methods for assessing satisfaction with requirements.

In conclusion, this study represents a significant step forward in the field of satisfaction with requirements in software development. The methodology proposed in this study provides a valuable tool for practitioners and researchers who are looking to improve the quality and effectiveness of their requirement processes. Further research and refinement of the methodology will be necessary to fully realize its potential and ensure that the needs and expectations of end users are met in future projects.

6. Conclusion

The significance of satisfaction with requirements cannot be overstated. Poor requirement specification and implementation can lead to costly rework, delays, and ultimately, project failure. On the other hand, effective satisfaction with requirements can lead to increased efficiency, improved user satisfaction, and successful project delivery.

A comprehensive analysis of satisfaction with requirements in software development was presented. Existing literature on the topic was reviewed, and common challenges and best practices were identified. As a result, in this paper, a methodology for quantitative assessing the satisfaction with requirements considering different characteristics of requirements before the development phase begins has been developed. The provided methodology allows assessing the level of satisfaction with requirements depending on different characteristics.

It is important to emphasize that the presented methodology, in contrast to others, is scalable, that is, the characteristics and scale of assessments can change depending on the needs and expectations of the project.

The next steps of the research will be dedicated to the improvement of the presented methodology by introducing new characteristics and their dependencies, and transparent decision-making systems (for the least influence of the human factor on the part of interested stakeholders). Also, in this regard, there is a need to consider the software implementation of the following methodology processes.

Contributions of authors: conceptualization – **Anastasiia Strielkina**; methodology – **Anastasiia Strielkina, Artem Tetskyi**; analysis – **Anastasiia Strielkina, Artem Tetskyi**; writing – original draft preparation – **Anastasiia Strielkina, Artem Tetskyi**; writing – review and editing – **Artem Tetskyi**.

All the authors have read and agreed to the published version of the manuscript.

References

1. The Standish Group. Available at: <https://www.standishgroup.com/> (accessed 20.12.2022).
2. Project Management Institute | PMI. Available at: <https://www.pmi.org/-/> (accessed 20.12.2022).
3. Bormane, L., Gržibovska, J., Bērziša, S., Grabis, J. Impact of Requirements Elicitation Processes on Success of Information System Development Projects. *Information Technology and Management Science*, 2016, vol. 19, no. 1, pp. 145-156. DOI: 10.1515/itms-2016-0012.
4. Wiegers, K. E., Beatty, J. *Software requirements*. Redmond, WA, Microsoft Press Publ., 2013. 672 p.
5. PMI. *A guide to the Project Management Body of Knowledge (PMBOK guide) and the standard for Project Management*. Newton Square, PA, Project Management Institute Publ., 2021. 250 p.
6. Hussain, A., Mkpojiogu, E. O., Kamal, F. M. The role of requirements in the success or failure of software projects. *International Review of Management and Marketing*, 2016, vol. 6, no. 7, pp. 306-311.
7. Holbrook, E. A., Hayes, J. H., Dekhtyar, A. Toward automating satisfaction with requirements assessment. *17th IEEE International Requirements Engineering Conference*, 2009, pp. 149-158. DOI: 10.1109/RE.2009.10.
8. Batool, A., Motla, Y. H., Hamid, B., Asghar, S., Riaz, M., Mukhtar, M., Ahmed, M. Comparative study of traditional requirement engineering and agile requirement engineering. *15th International Conference on Advanced Communications Technology (ICACT)*, 2013, pp. 1006-1014.
9. Beecham, S., Baddoo, N., Hall, T., Robinson, H., Sharp, H. Motivation in Software Engineering: A systematic literature review. *Information and software technology*, 2008, no. 50, pp. 860-878. DOI: 10.1016/j.infsof.2007.09.004.
10. Ranjan, B. S. C., Siddharth, L., Chakrabarti, A. A systematic approach to assessing novelty, requirement satisfaction, and creativity. *Ai Edam*, 2018, no. 32, pp. 390-414. DOI: 10.1017/S0890060418000148.
11. Horkoff, J., Yu, E. Comparison and evaluation of goal-oriented satisfaction analysis techniques. *Requirements Engineering*, 2013, no. 18, pp. 199-222. DOI: 10.1007/s00766-011-0143-y.
12. Fitriani, W. R., Rahayu, P., Sensuse, D. I. Challenges in agile software development: A systematic literature review. *International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, 2016, pp. 155-164. DOI: 10.1109/ICACSIS.2016.7872736.
13. Lindsjörn, Y., Sjøberg, D. I., Dingsøy, T., Bergersen, G. R., Dybå, T. Teamwork quality and project success in software development: A survey of agile development teams. *Journal of Systems and Software*, 2016, no. 122, pp. 274-286. DOI: 10.1016/j.jss.2016.09.028.
14. Strielkina, A. Information technology for dependability assessment and providing of healthcare IoT systems. *Radioelectronic and computer systems*, 2019, no. 3, pp. 48-54. DOI: 10.32620/reks.2019.3.05.
15. Morozova, O., Nicheporuk, A., Tetskyi, A., Tkachov, V. Methods and technologies for ensuring cybersecurity of industrial and web-oriented systems and networks. *Radioelectronic and computer systems*, 2021, no. 4, pp. 145-156. DOI: 10.32620/reks.2021.4.12.

16. Kanneh, S., Anu, V. Security Requirements Prioritization Techniques: A Survey and Classification Framework. *Software*, 2022, vol. 1, no. 4, pp. 450-472. DOI: 10.3390/software1040019.

17. Kuhail, M. A., Lauesen, S. User Story Quality in Practice: A Case Study. *Software*, 2022, vol. 1, no. 3, pp. 223-243. DOI: 10.3390/software1030010.

18. IEEE Software Engineering Standards Committee. IEEE Recommended Practice for Software Requirements Specifications. *IEEE Std 830-1998*, 1998, pp. 1-40. DOI: 10.1109/IEEESTD.1998.88286.

19. Kononenko, I., Sushko, H. Mathematical model of software development project team composition optimization with fuzzy initial data. *Radioelectronic and*

computer systems, 2021, no. 3, pp. 149-159. DOI: 10.32620/reks.2021.3.12.

20. Fedorovich, O., Uruskiy, O., Pronchakov, Y., Lukhanin, V. Method and information technology to research the component architecture of products to justify investments of high-tech enterprise. *Radioelectronic and computer systems*, 2021, no. 1, pp. 150-157. DOI: 10.32620/reks.2021.1.13.

21. IIBA. *A Guide to the Business Analysis Body of Knowledge (BABOK Guide)*. Toronto, Ontario, Canada, International Institute of Business Analysis Publ., 2015. 512 p.

Надійшла до редакції 07.01.2023, розглянута на редколегії 20.02.2023

МЕТОДОЛОГІЯ ОЦІНЮВАННЯ ЗАДОВОЛЕНOSTІ ВИМОГАМИ НА РАННІХ СТАДІЯХ ПРОЦЕСУ РОЗРОБЛЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Анастасія Стрелкіна, Артем Тецький

Предметом дослідження є процес забезпечення задоволеності вимогами при розробленні програмного забезпечення. Якісний етап розроблення вимог до системи, що проектується, для виконання всіх бізнес-цілей, задоволення замовника і, зрештою, задоволення кінцевого користувача, є однією з ключових умов ефективної реалізації будь-якого ІТ-проекту. Рівень задоволеності вимогами має підвищуватись як одна з передумов успіху проекту шляхом розроблення вимог. Щоб гарантувати, що продукт чи послуга відповідають потребам та очікуванням своїх користувачів чи споживачів, дуже важливо забезпечити задоволеність вимогами. Основна **мета** запропонованого дослідження – представити методологію кількісного оцінювання рівня задоволеності вимогами з урахуванням різних характеристик вимог на початку етапу розроблення. **Завдання**, що вирішуються: дослідити сучасний стан предметної області; розробити методологію оцінювання рівня задоволеності вимогами; надати та дослідити запропоновану методику на реальному прикладі; рекомендувати дії щодо підвищення рівня задоволеності вимогами. Представлена **методологія**, на відміну від інших, для отримання кількісної оцінки рівня задоволеності вимогами розглядає такі характеристики як атомарність, повнота, несуперечність, лаконічність, здійсненність, однозначність, тестованість, пріоритетність, зрозумілість, безпечність і продуктивність. Отриманим **результатом** є методологія кількісного оцінювання рівня задоволеності вимогами з урахуванням різних характеристик вимог на початку етапу розроблення. Дане дослідження є значним і необхідним, тому що в більшості випадків попередні дослідження не пропонують вичерпних кількісних та вимірних методів визначення ступеня задоволеності вимогами до тих чи інших характеристик. Також показано використання створеної методології з реальними вимогами. Додатково наведено рекомендації щодо посилення рівня задоволеності вимогами. **Висновок**. Запропонована методологія є розширюваною, на відміну від інших, а це означає, що характеристики та шкала оцінок можуть фактично змінюватись в залежності від вимог, цілей та інших особливостей ІТ-проекту.

Ключові слова: вимоги; розроблення програмного забезпечення; задоволеність вимогами; розроблення вимог; характеристики вимог.

Стрелкіна Анастасія Андріївна – д-р філос. з комп'ютерних наук, бізнес аналітик, Software Development Hub LLC, Харків, Україна.

Тецький Артем Григорович – канд. техн. наук, доц. каф. комп'ютерних систем, мереж і кібербезпеки, Національний аерокосмічний університет ім. М. Є. Жуковського “Харківський авіаційний інститут”, Харків, Україна.

Anastasiia Strielkina – PhD in Computer Sciences, Business Analyst, Software Development Hub LLC, Kharkiv, Ukraine,

e-mail: a.strelkina@sdh.com.ua, ORCID: 0000-0002-7760-7367, Scopus Author ID: 57194779158.

Artem Tetskiy – Candidate of Technical Sciences, Associate Professor of the Computer Systems, Networks and Cybersecurity Department, National Aerospace University “Kharkiv Aviation Institute”, Kharkiv, Ukraine, e-mail: a.tetskiy@csn.khai.edu, ORCID: 0000-0003-1745-2452, Scopus Author ID: 57202894656.