Сучасний стан наукових досліджень та технологій в промисловості. 2023. № 2 (24)
ISSN 2522-9818 (print)
ISSN 2524-2296 (online)

UDC 004.41
DOI: https://doi.org/10.30837/ITSSI.2023.24.039

D. GOBOV, O. ZUIEVA

IDENTIFYING THE DEPENDENCIES BETWEEN IT PROJECT CONTEXT AND BUSINESS ANALYSIS DOCUMENT CONTENT

The subject matter of the article is software requirements documentation practices in IT projects. **The goal** of the work is to identify what information is included in business analysis deliverables and how the project context influences the content of business analysis documents. The following **tasks** were solved in the article: to examine the industrial standards and experience of business analysts and requirements engineers in requirements specification and modeling activities in software development, to create and conduct a survey on practices in requirement documentation activities in IT projects, to define practitioners' preferences regarding business analysis documents' content, and to define how project context influences document content. The following methods are used: three hundred and twenty-four practitioners from Ukrainian companies were surveyed about their current preferences in business analysis document creation, their experience, and project profile attributes in which techniques were used. The Chi-Square test of independence and Cramer's V effect size measure were applied to define statistically significant dependencies between project context and business analysis document content. The following results were obtained: a list of the most commonly used elements in business analysis documents was defined. Seventy-eight statistically significant associations for pairs "project context – business analysis document content" were found based on the p-value of the Chi-Square test, for eighteen of which the significance of the identified dependencies was confirmed using the Cramer's V effect size measure. Conclusions: It is concluded that project context influences the content of business analysis documents in IT projects. The most influential factors are the business analyst's experience, team distribution, company size and type, template usage, and the purpose of using the requirements documents. The found dependencies can guide the selection of business analysis document structure and the creation of project-specific templates during the creation business analysis approach and business analysis information management approach.

Keywords: business analysis document; requirements engineering; project context; Cramer V; empirical study.

Introduction

According to the Project Management Institute (PMI) definition in [1], business analysis is the set of activities performed to support the delivery of solutions that align with business objectives and provide continuous value to the organization. In most cases, the solution includes an IT component in the form of software, hardware, or software-hardware systems. The primary outcomes of business analysis activities, including IT projects, are requirements and design, which define needs and solutions on different levels. Network analysis of business analysis tasks and artifacts [2] showed that solution scope, future state description, and requirements are the most influential elements. These business analysis outputs laid the groundwork for all future development and testing activities. The form of organizing business analysis results can vary depending on the selected Software Development Life Cycle (SDLC), stakeholders' preferences, company standards, and previous business analysts' experience. All individual models and specifications must be fitted together into requirements architecture to ensure that all requirements form a single whole that supports the overall business objectives and produces a valuable outcome for stakeholders [3]. Usually, requirement architecture creation starts from predefined templates or frameworks, which can be defined at the company or project level.

Guidelines for business analysis and requirements engineering developed by international specialized organizations PMI [1], IIBA [3], and IREB [4] do not prescribe the structure of business analysis documents. The international standard developed by the IEEE [5] offers three requirements document templates: Stakeholder requirements specification, System requirements specification, and Software requirements specification, of which the latter is focused on describing requirements for software systems. RUP [6], one of the popular iterative software development methodologies, offers a two-level model for describing requirements: a Vision document for describing the top-level boundaries of a solution and a System requirements specification document for describing detailed requirements. The main limitation of these approaches is that they are more focused on plan-driven approaches, while today, most projects are implemented on an adaptive approach. Secondly, these templates are recommendations and should be adapted by business analysts to the project's specifics. According to a survey [7], only 5% of practitioners use the document templates recommended in the standards, and more than 50% use the standards developed by the company. All this indicates the relevance of studying current practices for creating business analysis documents, their structure, and the impact of the project context on the tasks of documenting requirements.

This study was conducted to analyze the current preferences of business analysts and requirement engineers regarding business analysis documents' content for software development projects. We also wanted to define how a project context influences the probability of choosing specific document content elements. Using a survey, we studied the experience of practicing business analysts and requirement engineers working in Ukrainian and international companies. The survey results were analyzed via statistical analysis in conjunction with the Chi-Square test and Cramer-V measure.

This article is organized as follows. Section II includes a review of related works describing elicitation activities and technique selection. Section III is devoted to the survey results, and section VI includes the result of statistical analysis. Section V concludes the paper with a discussion of the findings of our study and future work.

Analysis of last achievements and publications

Most of the related works are dedicated to analyzing requirements document structure and templates used as a basis for creation. Several of these works are devoted to the analysis of the structure of the document "Software Requirements Specification" (SRS). Rączkowska-Gzowska and Walkowiak-Gall [8] conducted a survey among 163 practitioners from commercial software development projects. The frequency of availability of certain types of information in the SRS and the project context's influence on them were analyzed. The project's domain was found to affect the information in the SRS. For example, the structure of the processed data, reports, and dictionaries were rarely used in the SRS for projects in the telecommunications industry. Statistical analysis was not performed to check the influence between project context and SRS structure nor for the use of templates to develop documentation. Franch, Palomaresm Quer, Chatzipetrou, and Gorschek [9] analyzed current practices regarding artifacts in SRS, templates, and tools for document creation based on the results of 24 interviews at 12 Swedish companies. The most used artifact categories were natural language, use case, and user story. Most respondents used organizational standards. Limited participants did not allow statistical analysis to define dependencies between project context and business analysis artifacts and templates. Wagner, Fernández, Felderer, and Kalinowski [10] studied the practitioners' preferences in functional requirements documenting. The degree of formality and specification techniques were combined to allow participants to select the most appropriate options. The most popular ways to document function requirements were free-form textual domain/business process models, free-form textual structured requirements lists, and use case models. The study [11] presented the result of the worldwide survey where except functional requirements were analyzed ways for specification of non-functional requirements.

Other business analysis outputs such as business requirements, user interfaces, and assumptions, were not analyzed. Abdalazeim and Meziane [12] reviewed existent approaches of generation natural language specifications from the object UML model. Medeiros et al. [13] studied requirements document content in agile projects. The approach Requirements Specification for Developers was proposed. This approach assumes that each functional and non-functional requirement is defined in terms of data entities model, acceptance criteria, and user interface mockup points of view, while the last one is non-mandatory. An industrial case study was performed to evaluate the proposed approach and showed that this approach allows business analysts to meet the developer's expectations in most cases. The disadvantages of this approach include ignoring business and stakeholder requirements, as well as ignoring modeling techniques. Heck and Zaidman [14] studied quality criteria for Requirements Specifications in Agile projects. Twenty-eight quality criteria were described and categorized. One of the most important quality criteria for agile and plan-driven projects is completeness on the single requirement's level and requirement specification document's level.

A set of studies is dedicated to analyzing business analysis techniques used to specify and model requirements. Despite the fact that they do not directly analyze the composition of business analysis artifacts, such techniques as the user story, the use case, and the family of UML diagrams significantly affect the content of business analyst outputs. Jarzębowicz and Połocka [15] conducted a study that involved 42 professionals from Polish IT companies. The survey asked the participants to indicate which requirement specification or modeling techniques they considered applicable to different projects. The survey allowed respondents to choose from a list of 15 techniques, and they could choose any number of techniques that they considered applicable to the given context. However, due to the limited number of participants, the study did not evaluate the statistical significance of the results. Moreover, the survey only captured the participants' preferences, not their current practices. Gobov [16] analyzed the current practices of using specification and modeling techniques in IT projects implemented in Ukraine. The study identified the most commonly used requirements analysis techniques in IT projects: use case, user story, and acceptance criteria were defined as the most popular specification techniques; activity diagram, business process model, and sequence diagram were identified as the most popular modeling techniques. Statistically significant relationships between the context of the project and the techniques used in it were also identified. At the same time, these studies do not define the structure and content of business analysis documents.

Based on the results of the analysis of existent publications, it can be concluded that the task of choosing content for business analytical documents in IT projects remains relevant and unresolved.

The purpose of this article is to identify current practices regarding business analysis document content and to define the influence of project context on it**.**

Solving of the problem

A survey was conducted to collect information on current practices for creating business intelligence documents, which was attended by 328 practicing business analysts and requirements engineers from Ukrainian and foreign companies. The structure of the questionnaire in terms of information about the context of the project was built based on the questionnaire from the NAPIRE initiative [11]. The list of possible sections in business analysis documents was compiled based on international standards and templates reviewed in the literature review and contained the following options:

- Assumptions
- Background (rationale and context for the new solution decision)
- Business requirements
- Business rules
- Constraints
- Cost-benefit analysis
- Data models
- Dependencies/Integrations
- Deployment specifics
- Functional requirements
- Glossary
- Goals & Objectives
- Non-functional requirements
- Open questions
- Problem statement
- Risks
- Stakeholders analysis (including target audience/personas)
- Success metrics (project/solution)
- Technical Interfaces
- Usage scenarios
- User interface(s)

Four participants were excluded from the analysis as they did not provide complete information about the project context. The survey was shared in English and Ukrainian languages, and Google Forms was used as the platform for data collection. The survey was subsequently distributed within the local Business Analysis communities and via professional and social networks, along with personal contacts established within the top 10 Ukrainian IT companies. The data collection period spanned one month, during which the responses were gathered. Following this phase, the collected data underwent merging and coding processes in preparation for further analysis. The survey results dataset has been made publicly available through the Mendeley Data repository.

According to the survey results illustrated in Fig. 1, the majority of respondents (69%) identified functional requirements as the most crucial section in software requirements documents. Business requirements were also commonly included, with 63% of respondents highlighting them in their documents. The high percentage of functional requirements highlighted suggests that this section describes the key features and functionalities of the software, which are typically the primary focus of the development process. Meanwhile, emphasizing business requirements highlights the importance of understanding stakeholders' needs and objectives.

Glossary and non-functional requirements followed closely behind, with 62% and 60% of respondents highlighting them. This indicates the importance of clear communication and understanding of technical terms and concepts used in the requirements, as well as software quality aspects such as usability, security, and performance, which are crucial for ensuring that the software meets the expected level of quality.

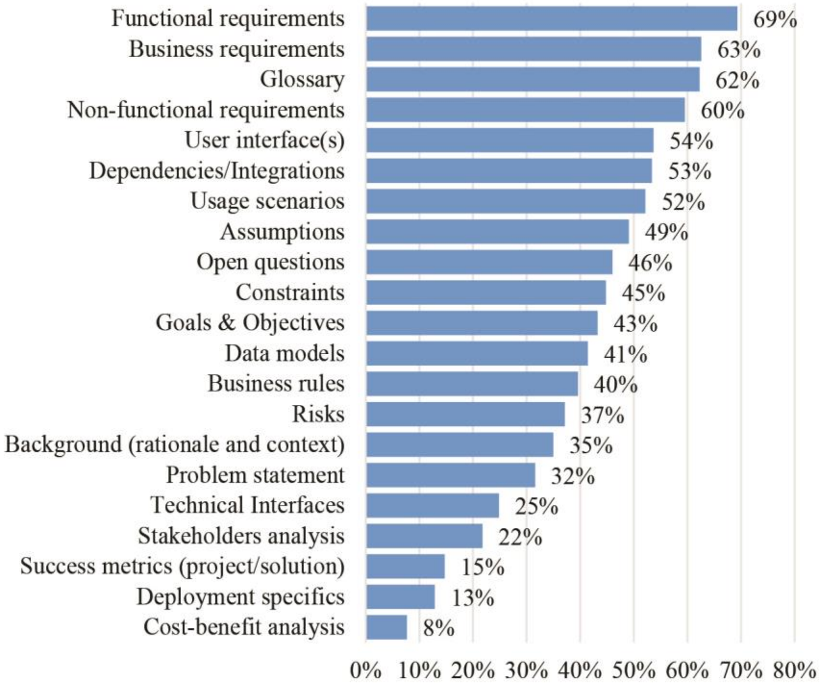

Fig. 1. Business analysis document contents

Other sections and subsections are highlighted in documentation in less than 60% of cases. The least highlighted sections include success metrics (project/solution), deployment specifics, and cost-benefit analysis. Success metrics (project/solution) were highlighted by only 15% of respondents. It suggests that defining and agreeing upon quantifiable and measurable indicators of success can be challenging for different stakeholders. As a result, business analysts may focus on more concrete aspects of software requirements, such as functional and non-functional requirements, leaving success metrics to be defined at a later stage in the software development process.

Deployment specifics (13%) and cost-benefit analysis (8%) were the least highlighted sections. Deployment is typically considered a separate phase of the software development lifecycle, and business analysts may focus on requirements relevant to the design and development phases rather than deployment. Cost-benefit analysis is usually performed at a higher level of project planning, and project managers or other stakeholders may perform it at a later stage in the planning process, which is why business analysts may not include it in the software requirements documentation.

Given that the majority of respondents work on Agile projects, it may have influenced the results. The agile methodology is an approach to software development that emphasizes the rapid delivery of working software over comprehensive documentation, according to one of its basic principles [17]. In [18], it is noted that a complete requirements document is not needed for agile and user-centered design since the requirements change rapidly, making the document quickly outdated and rendering the engineer’s efforts mostly wasted. Thus, the software requirements document should define only the minimum functional requirements. As a result, business analysts often prioritize the essential documentation sections, such as functional and business requirements, that are crucial for the success of the development process. Therefore, it is necessary to analyze the relationship between contextual factors such as company type, project approach, etc., and the specific sections of the highlighted business analysis documentation.

The previous analysis examined the relationship between project context factors and documentation section inclusion. The Chi-Square test of independence was used to test the relationship, considering a significance level of 0.05. However, the Chi-Square test has limitations when comparing tables of different dimensions. Cramer's V, a chi-square-based association measure, was used to adjust the results to address this [19]. Cramer's V ranges from 0 to 1, where 0 indicates no association and 1 represents a perfect association. For this research, we adopted SPSS's definition of effect size: $V \leq 0.2$ for the weak association, $0.2 < V \leq 0.6$ for the moderate association, and $V > 0.6$ for the strong association. The choice to use Cramer's V was justified due to the categorical nature of the variables, the need to assess the strength of association, the use of contingency tables for analysis, and the availability of a standardized metric for comparison.

An analysis was conducted using Cramer's V statistic to examine the dependencies between context factors, including the business sector, company type, company size, project (team) size, project category, team distribution, way of working, templates usage, experience, and way of using requirements, and the specific sections and subsections that business analysts highlighted in their documentation. The outcomes are presented in Table 1.

Table 1. *Cramer's V for project context and document contents associations*

Business Analysis Document Contents	Business Sector	Company Type	Company Size	Project (team) Size	Project Category	Team Distribution	Way of Working	Templates Usage	Experience	Requirements used as:				
										basis for the implementation	source for the tests	used in customer acceptance	part of the contract	reminder for further discussions
Assumptions	0	0.231	0.216	0	0	0.201	0.142	0.275	0.191	0	0.207	0	0	0.167
Background	0	0	0	0	0.164	0.173	0	0.214	0.156	0	0.208	0	0.138	0.234
Business requirements	0	0	0	0	0	0	0	0	0.196	0	0	0.128	0	0
Business rules	0	0	0	0	0	0.12	0	0	0.184	0	0	0.181	0	0
Constraints	0	0	0	0	0	0	0	0.186	0.189	0	0.11	0.216	0	0.156
Cost-benefit analysis	0	0	0	0	0	0	0	0	0	0	0	0	0.127	0
Data models	0	0	0	0	0.182	0.143	0	0	0.232	0	0.144	0.122	0	0.183
Dependencies/Integrations	0	0.122	0	0	0	0	0	0.185	0	0	0	0.166	0	0
Deployment specifics	0	0	0	0	0	0	0	0.158	0	0	0	0.128	0.143	0.200
Functional requirements	0	0	0	0	0	0	0	0.185	0.184	0	0.113	0.14	0.117	0
Glossary	0	0	0	0	0	0	0	0	0.167	0.143	0.149	0	0.149	0
Goals & Objectives	0	0	0	0	0	0	0.158	0	0.222	0	0	0	0.224	0.182
Non-functional requirements	0	0	0.136	0	0.208	0	0	0.196	0.221	0	0.165	0	0.174	0.131
Open questions	0	0	0.138	0	0	0.179	0	0	0	0	0.167	0	0	0.13
Problem statement	0	0	0	0	0.166	0	0.152	0	0.229	0	0	0	0	0.161
Risks	0	0	0	0	0	0.12	0	0	0.238	0	0	0	0.111	0.124
Stakeholders analysis	0	0	0	0	0	0	0	0	0.174	0	0	0	0	0
Success metrics	0	0	0	0.163	0	0	0	0	0.18	0	0	0	0	0
Technical Interfaces	0	0	0	0	0	0	0	0	0.172	0	0	0	0	0
Usage scenarios	0	0	0	0	0	0	0	0	0.204	0	0	0.171	0	0.142
User interface	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Based on the data presented in Table 1, it can be concluded that the business sector, or the project's domain, does not have a statistically significant impact on the inclusion of any of the sections presented in the documentation. This may indicate that the content of the documentation is not driven solely by the specific needs of the business sector. Instead, other factors such as the distribution of the development team, the maturity of the organization's software development process, and the business analyst's expertise may be more influential in determining the content of the documentation. While each business sector may have unique requirements and specifications for its software projects, the overall process of gathering, documenting, and communicating software requirements may be similar across different industries. Business analysts may use similar techniques and methodologies to identify and document software requirements. The sections commonly highlighted in the research (functional requirements, business requirements, glossary, and non-functional requirements) may be considered fundamental components of software requirements documentation necessary for all software projects, regardless of the sector. For example, business analysts may follow a template or framework that is not tailored to the specific needs of the business sector. Furthermore, as mentioned previously, software requirements often focus on the functionality and behavior of the software system rather than the specific business domain. Therefore, the content of software requirements documentation may not vary significantly across different business sectors, as the focus is on the software system itself rather than the business domain it supports. However, it is essential to note that while the business sector may not have a statistically significant impact on the inclusion of specific sections in software requirements documentation, it may still impact the specific requirements and specifications included in the documentation.

Company type moderately impacts the Assumptions section in software requirements documentation (V = 0,231) and weakly impacts the section on system integration and dependencies (V = 0,122). A survey revealed that 57% of Outstaff and Outsource IT business analysts highlighted Assumptions, compared to 42% from IT Product companies and 26% from non-IT companies. Outstaff and Outsource IT companies may prioritize Assumptions due to risks in the requirements-gathering process and are more likely to document system integration and dependencies. IT Product companies may focus on meeting market needs and emphasize assumptions less. In comparison, non-IT companies may have more intimate knowledge of business context and stakeholders and emphasize assumptions less.

Most of the dependencies on contextual factors were found for the Assumptions section: 8 out of 14 contextual factors have a statistically significant impact on the presence of the Assumptions section in the documentation, with 5 of them having a moderate level of influence and 3 being weak. In addition to company types, factors such as company size (V = 0.216), team distribution (V = 0.201), and template usage (V = 0.275), as well as using requirements documentation as a source for tests (V = 0.207), have a moderate impact on the use of the Assumptions section by business analysts. The associations between the factors of team distribution and company size and the Assumptions section are presented in Fig. 2.

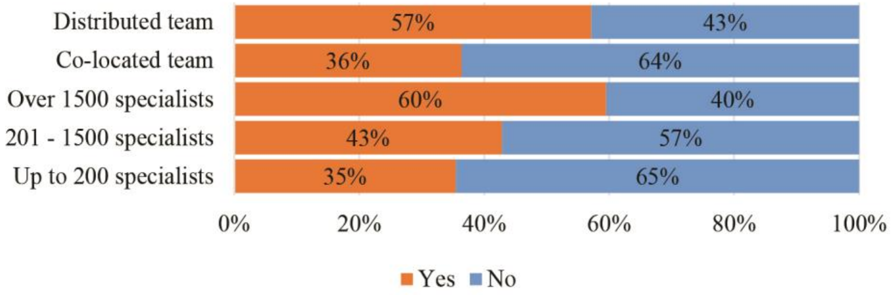

Fig. 2. Team distribution and company size associations to the Assumptions section

From Fig. 2, it can be concluded that in larger companies, where there may be more bureaucracy and a higher level of risk aversion, there may be a greater emphasis on documenting assumptions to ensure that all stakeholders have a shared understanding of the project goals and potential risks. The Assumptions section is used in documentation in 60% of large companies, compared to 35% of small companies. Smaller companies may have a more informal approach to documentation and risk management, which could result in a lower level of emphasis on documenting assumptions. Similarly, the impact of team distribution on using the Assumptions section can be explained by the fact that distributed teams often face more significant communication challenges than co-located teams. In distributed teams, the Assumptions section is used 20% more often than in co-located teams. A distributed team may need standardized templates and clear documentation to ensure that all team members have a shared understanding of the requirements. The moderate impact of template usage (V = 0,275) can be explained by the fact that templates provide a structured format for documenting requirements, making it easier to ensure that all required information is included in the documentation. It can increase the likelihood that the assumptions section is consistently and thoroughly documented across different projects.

Using requirements documentation as a source for tests has a moderate impact on using the Assumptions section (V = 0,207) because it provides valuable insights into the intended use and limitations of the software being developed, which can inform the testing process. Thus, a focus on testing may encourage business analysts to document assumptions.

The way of working (WoW) and the number of years of analyst experience have been statistically confirmed to have a weak impact on using the Assumptions section. The impact of the WoW on the usage of the Assumptions section may be weaker than other contextual factors because the documentation practices may be more influenced by the specific organization or project rather than the overall methodology being used. The more substantial influence of company type, company size, team distribution, and template usage factors can explain the weaker impact of analyst experience. In large distributed teams, the importance of documenting assumptions may be emphasized when training new analysts.

The use of templates not only affects the Assumptions section but also influences the presence of the Background section in the documentation (V=0.214), which outlines the necessary context and prerequisites for implementing the solution. The association between an analyst's use of templates and the inclusion of the Background section is shown in Fig. 3.

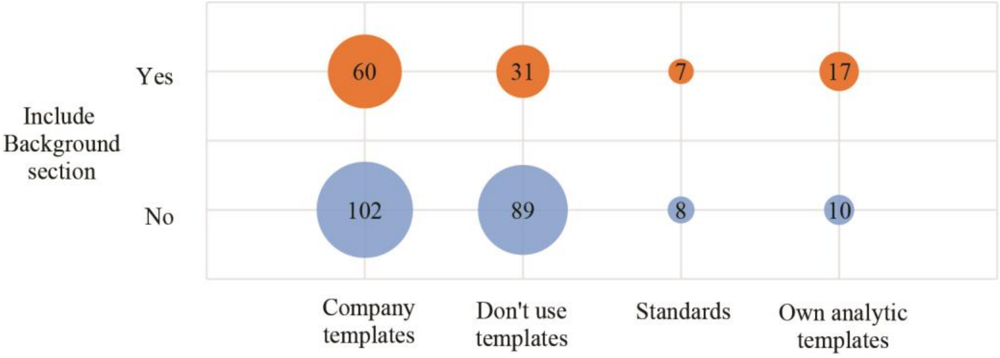

Fig. 3. Templates usage associations to the Background section

According to Fig. 3, most analysts use company-approved documentation templates that typically exclude the Background section. One possible explanation is that the Background section may not be deemed relevant or necessary for the specific project. Sometimes, the focus may be on the system's specific requirements instead of providing a complete overview of the project's context and history. Additionally, there may be concerns about the documentation's length and complexity. Analysts who do not use templates also tend to skip this section in their documentation, as 74% of respondents in this group answered negatively. It can be attributed to the fact that analysts are attempting to document only the essential information, focusing on the project's specific aspects, and may view the Background section as not directly linked to the requirements.

Only 5% of respondents use templates presented in standards such as ISO/IEC/IEEE. Half of the analysts who use standard templates include the Background section, while the other half do not. In contrast, the majority (63%) of those who use self-developed templates include the Background section in their documentation. The lower frequency of utilizing templates from standards and the mixed results regarding the inclusion of the Background section indicate that more work is needed to standardize software documentation templates to ensure that they include all necessary sections and adequately reflect the field's best practices. Analysts who use self-developed templates are more inclined to include the Background section, implying that they consider it essential for comprehending the project's context and objectives.

The survey data shows 63% of respondents use templates when creating documentation. However, according to [9], in Swedish companies, the proportion is higher at 75%, although it should be noted that the number of respondents in the second study is significantly smaller.

The use of templates also impacts the inclusion of the constraints, dependencies, integration, deployment specifics, functional and non-functional requirements sections because templates provide a standardized and structured approach to documenting a project, making it easier for analysts to ensure that all necessary information is included. The survey results show that 77% of respondents who use templates approved by the company create a separate section for functional requirements. Similarly, 70% of respondents who use their own templates also separate functional requirements. In contrast, respondents who do not use templates or rely on standards highlighted functional requirements as a separate section only in 60% of cases. It underscores the importance of using templates and the need to improve templates presented in standards.

In addition to the template usage, the inclusion of the Background section in the documentation is moderately influenced by two factors: using the documentation as a source for tests (Cramer V = 0.208) and as a reminder for further discussions (Cramer V = 0.234). Using the documentation for tests highlights the importance of the Background section in providing essential information for effective test planning and execution. Similarly, using the documentation as a reminder for discussions emphasizes the role of the Background section in facilitating communication and collaboration among project stakeholders. The Background section acts as a shared reference point by incorporating relevant background information, promoting understanding and effective communication. These findings underscore the significance of providing adequate context and reference material in software requirements documentation. Recognizing the value of the Background section in supporting tests and discussions can enhance the quality and effectiveness of the requirements engineering process.

Most of the dependencies were found for the experience factor. It has a statistically significant influence on the separation of 16 out of 21 documentation sections presented in the study – on 6 of which experience has a moderate level of influence, and on ten, it has a weak influence. The experience of the analyst has a moderate impact on such documentation sections as Data models (Cramer’s V = 0,232), Goals & Objectives (Cramer’s V = 0,222), Non-functional requirements (Cramer’s V=0,221), Problem statement (Cramer’s V=0,229), Risks (Cramer’s V = 0,238) and Usage scenarios (Cramer’s V = 0,204). Fig. 4 shows associations between the analyst’s experience factor and documentation sections, on which it has a moderate impact.

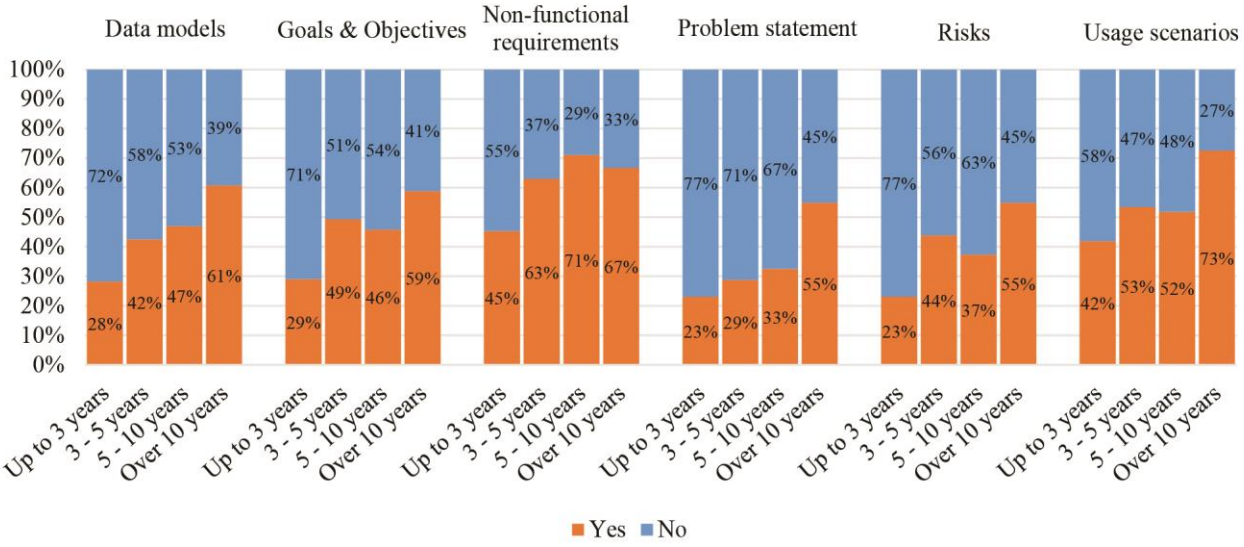

Fig. 4. Impact of business analyst experience on document content

According to Fig. 4, the separation of the mentioned sections in documentation increases with the analyst’s experience. Among respondents with over ten years of experience, sections such as Data models, Goals & Objectives, Non-functional requirements, Problem statements, Risks, and Usage scenarios are highlighted on average 30% more often than respondents with less than three years of experience. It can be explained by the fact that experienced analysts gain a deeper understanding of project requirements over time, allowing them to identify better and articulate critical project components. Experience also enables analysts to identify and analyze relationships between different project elements, which can prove invaluable in problem-solving and project planning. Experienced analysts are exposed to a broader range of project scenarios, and their combination of technical expertise, business acumen, and practical experience enables them to develop an intuitive sense of what is required to achieve successful project outcomes.

Consequently, experienced analysts are able to develop stronger documentation skills, such as the ability to clearly communicate complex ideas, anticipate potential issues, and organize information effectively. These skills enable them to produce more comprehensive and informative documentation, which leads to the inclusion of additional sections. Therefore, fostering a corporate culture that involves experienced analysts

ISSN 2522-9818 (print)
ISSN 2524-2296 (online)

creating company documentation templates and transferring knowledge to less experienced analysts is critical. Using templates and knowledge transfer will lead to better documentation of project requirements and, subsequently, more effective implementation of software development projects.

The data presented in Fig. 4 demonstrate a correlation between analysts' experience and using the Data models section in the documentation. As analysts gain more experience, there is an observed increase in highlighting the Data models section. Specifically, after surpassing the three-year experience threshold, there is a 14% rise in frequency and an additional 14% increase after ten years of experience. These findings indicate that experienced analysts recognize the importance of data modeling and consistently incorporate it into their documentation practices. The research findings (fig. 1) indicate that only 41% of respondents use data modeling. These findings are consistent with a study conducted by Kassab et al. [20]. According to their research, 60% of respondents expressed their requirements using natural language, suggesting an informal approach, while only 33% utilized semi-formal methodologies like UML, class diagrams, and sequence diagrams. In addition, the study shows a significant impact of the absence of data models on the result and overall satisfaction of team members. Among respondents who did not employ any modeling methodology, only 69% felt that the finished product adequately met customers' needs, and a mere 48.5% reported that end users found the finished product easy to use.

In contrast, those who employed a modeling methodology had more positive perceptions, with 87% indicating that customer needs were met and 82% stating that the product was easy to use. These findings strongly suggest that incorporating data modeling into the documentation process positively influences the quality of the final product and customer satisfaction. Semi-formal representations like UML were mainly associated with higher end-user satisfaction levels. Among respondents who utilized semi-formal representations, 86% believed the finished product was easy to use, compared to 59% of those who used informal approaches. Additionally, 90% of respondents using semi-formal representations felt that the finished product's capabilities aligned well with customer or user needs, while the corresponding figure for those using informal methods was 65%. Therefore, including data modeling and using semi-formal representations in requirements documentation positively impacts the outcome and overall satisfaction of team members and end-users.

The sections Goals & Objectives and Risks demonstrate a similar trend (fig.4). After three years of experience, the inclusion of these sections increases by 20% but experiences a slight decline after five years. The utilization of the Goals & Objectives section decreases by 4%, while the Risks section decreases by 6%. However, after reaching the 10-year experience threshold, analysts observe a resurgence with a subsequent increase of 13% and 18% in utilization. The slight decline in utilizing the Goals & Objectives and Risks sections after five years of experience may be attributed to several factors. One possibility is that as analysts become more experienced, they may focus more on the execution and delivery of projects rather than explicitly documenting goals and risks. Also, it can be influenced by the nature of projects. However, after reaching the 10-year experience threshold, there is a resurgence in utilizing the Goals & Objectives and Risks sections. This resurgence could be attributed to a deeper understanding of the value of explicitly documenting goals and risks, particularly in complex projects or those with higher stakes, which experienced analysts often do. Analysts with extensive experience may recognize that clearly defining goals and effectively managing risks contribute to project success and stakeholder satisfaction.

Among analysts with 5–10 years of experience, there is a slight decrease in the usage of the Usage scenarios section, while analysts with over ten years of experience show a significant increase of 21%. It can be explained by the evolving role of usage scenarios in requirements engineering and the changing needs of stakeholders throughout the project lifecycle. Usage scenarios describe user interactions and ensure the system meets their needs. In the early stages of an analyst's career, the focus is mainly on functional and non-functional requirements, with less emphasis on usage scenarios. However, as analysts gain more experience, they recognize the importance of capturing user interactions and ensuring usability and satisfaction. Experienced analysts proactively include usage scenarios to capture critical workflows, system behavior, and interaction patterns. The significant increase among analysts with over ten years of experience may also be attributed to the growing adoption of user-centered design practices and the recognition of usability and user experience in software development. Analysts with extensive experience have encountered scenarios where neglecting usage scenarios led to suboptimal system performance or user dissatisfaction. They have witnessed the positive impact of considering usage scenarios, such as improved user acceptance and satisfaction.

Among analysts with over ten years of experience, there is a slight decrease in the usage of the Non-functional requirements section (67%) compared to analysts with 5 to 10 years of experience (71%). It suggests a nuanced trend in considering non-functional requirements. Non-functional requirements specify the quality attributes of a software system, such as performance, reliability, security, and usability. They are crucial in shaping the system’s behavior and meeting user expectations. Shift towards agile methodologies, emphasizing collaboration and face-to-face communication, could reduce the reliance on formal documentation, including non-functional requirements. The observed decrease in utilization among experienced analysts can be attributed to their understanding of non-functional requirements, efficient communication channels, and agile methodologies. Nonetheless, alternative approaches should be employed to address non-functional requirements and ensure system quality and stakeholder satisfaction.

Lastly, using the Problem statement section as a separate entity shows a gradual rise (fig. 4). Before reaching the 10-year experience mark, most analysts do not single out this section. However, among analysts with over ten years of experience, there is a notable 22% increase in its usage as a distinct section, causing the number of analysts who emphasize it to surpass those who do not. The increased emphasis on the Problem statement section reflects the maturity and expertise acquired by analysts over time, highlighting their growing ability to accurately identify and address a software project’s core challenges and objectives. A well-formulated problem statement should create an awareness of the issue and encourage innovative thinking without prescribing a solution or introducing biases towards any particular strategy. Emphasizing the problem statement as a distinct section is essential in the long term because it ensures that the team concentrates on the appropriate problem and comprehends the fundamental causes. This approach can help prevent being constantly reactive to issues. Skilled analysts tend to have a more strategic mindset, which accounts for the considerable rise in the use of the Problem statement section in this respondent group.

It should be noted that the level of analyst experience significantly affects only the "Non-functional requirements" section among the top five sections most frequently highlighted separately by analysts (fig. 1). For the "Business requirements," "Functional requirements," and "Glossary" sections, there is a minor impact of experience level, but the trend of highlighting separate sections in the documentation remains consistent, proportional to the respondents’ experience level (fig. 5).

According to the survey results, the Non-functional requirements section is the only category the Project Category statistically significantly influences. The relationship between the Project Category and the inclusion of the Non-functional requirements section is depicted in Fig. 6.

Among the four project categories identified in the research, the inclusion frequency of the Non-functional requirements section varies (fig. 6). In development from scratch projects, where software is built from the ground up, the Non-functional requirements section is separate in 69% of cases. This high percentage indicates an early focus on specifying and addressing non-functional aspects such as performance, security, reliability, and usability. For User interface engineering projects involving redesigning an existing system’s user interface, the Non-functional requirements section is separate in 61% of cases. It indicates that non-functional aspects are still significantly considered even when the main focus is on the user interface.

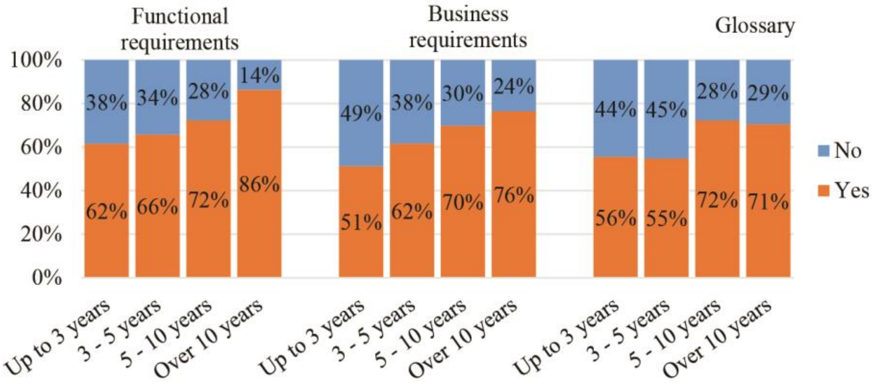

Fig. 5. Impact of business analyst experience on document content (weak association)

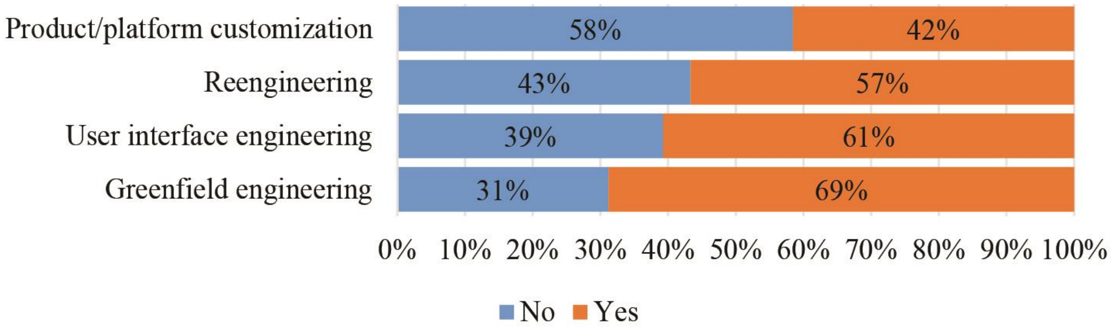

Fig. 6. Project Category associations to the Non-functional requirements section

In Reengineering projects, which involve redesigning and reimplementing an existing system, the Non-functional requirements section is separate in 57% of cases. Although slightly lower than the previous categories, it underscores the importance of addressing non-functional aspects during reengineering. The lower percentage of analysts highlighting the Non-functional requirements section separately in Reengineering projects can be attributed to several reasons. Firstly, the focus is improving existing systems and prioritizing functionality over non-functional aspects. Secondly, working with legacy systems may lead analysts to assume pre-established non-functional requirements. Extensive analysis of the current system’s architecture takes precedence over extensive documentation. Time and budget constraints also result in a pragmatic approach, where non-functional requirements may be addressed within functional requirements or communicated informally.

Conversely, in Product/Platform customization projects, the Non-functional requirements section is separate in only 42% of cases, suggesting that non-functional aspects may receive less emphasis in this category. Importantly, in this category, non-functional requirements are often integrated within other documentation sections rather than isolated as separate. The variation in including the Non-functional requirements section across project categories can be attributed to differing priorities, constraints, and contexts associated with each project type. Development from scratch projects may require more careful consideration of non-functional requirements due to their clean slate approach. Conversely, Product/Platform customization projects may rely on pre-existing frameworks or platforms where non-functional requirements are already established and documented implicitly.

No statistically significant influence factors were found for the "User interface" section. It suggests that company type, size, experience level, and other contextual factors did not significantly impact whether business analysts include a separate "User interface" section in their documentation. The absence of statistically significant influence factors for the "User interface" section indicates that its inclusion is likely determined by other factors not considered in this research study. The decision to include the "User interface" section may be more influenced by project-specific considerations, client requirements, or the expertise and preferences of individual analysts rather than general contextual factors.

Including the Deployment specifics section in software requirements documentation is moderately influenced by utilizing the documentation as a reminder for further discussions. Fig. 7 presents the ratio of analysts who use the documentation for discussions and the percentage of those who include the Deployment specifics section.

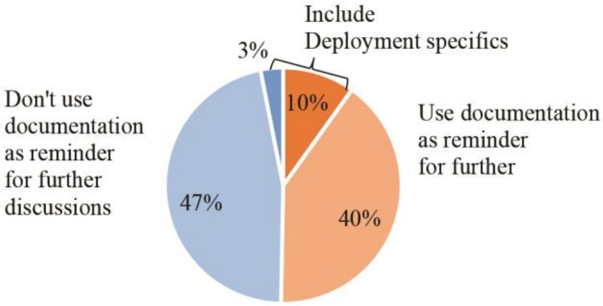

Fig. 7. Associations of usage documentation as a reminder for further discussions in the Deployment specifics section

The ratio shows an equal distribution between analysts who use the documentation as a reminder for further discussions and those who do not (fig. 7). Among those who use the documentation for discussions, the inclusion of the Deployment specifics section is three times more frequent compared to those who do not. It implies that utilizing the documentation for discussions increases the likelihood of including the Deployment specifics section. Fig. 7 illustrates a relatively low percentage of usage for the Deployment specifics section. This section usually contains pertinent information regarding deployment considerations, such as hardware requirements, installation processes, or system configurations. As previously mentioned, business analysts tend to focus more on requirements related to the design and development phases rather than deployment-specific details. The moderate influence of employing the documentation for discussions suggests that continuous communication and collaboration among project stakeholders encourage the inclusion of the Deployment specifics section.

The inclusion of the Constraints section in software requirements documentation is moderately influenced by the use of documentation in customer acceptance (Cramer's V = 0.216). This factor explicitly impacts the Constraints section, indicating that documentation used in customer acceptance affects whether or not the Constraints section is included. Fig. 8 illustrates the association between the use of documentation in customer acceptance and the inclusion of the Constraints section.

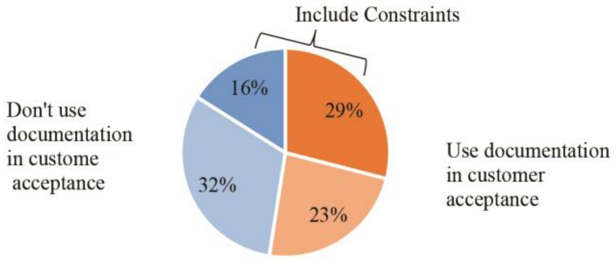

Fig. 8. Associations of usage documentation in customer acceptance to the Constraints section

Fig. 8 shows that the percentage of analysts working on projects where documentation is used in customer acceptance slightly outweighs those working on projects where documentation is not used for this purpose. Moreover, a higher percentage of analysts who use documentation in customer acceptance include the Constraints section in their documentation compared to those who do not. It suggests that utilizing the documentation in customer acceptance increases the likelihood of including the Constraints section. This section in software requirements documentation typically captures any limitations, restrictions, or conditions that need to be considered during the development and implementation of the software. The higher percentage of analysts who include the Constraints section among those who utilize documentation in customer acceptance indicates the value of involving customers in the requirements process and considering their constraints and acceptance criteria. By using documentation in customer acceptance, analysts can better understand and capture the constraints relevant to meeting customer expectations and ensuring the successful adoption of the software.

Conclusions

A survey study was conducted to analyze the current approaches to creating business analysis documents and the influence of project context on them. A survey with 328 business analysts and requirements engineers from Ukrainian and foreign companies gathered information on current practices in creating business analysis documents. The questionnaire was created based on the NAPIRE initiative and international standards. It included the following business analysis document sections: assumptions, business requirements, constraints, data models, functional requirements, non-functional requirements, and others.

The survey results highlight the importance of functional and business requirements, glossary, and non-functional requirements in software requirements documentation. However, success metrics, deployment specifics, and cost-benefit analysis receive less emphasis, possibly due to challenges in defining metrics and the preference for lean documentation in Agile methodologies.

The influence of the project context on the structure of the business analysis document was identified by using the Chi-square test. The significance of statistical dependencies between the pair "project context – document section" was assessed by Cramer's V effect size measure

Cramer's V statistic indicates that the business sector does not significantly impact specific sections' inclusion in software requirements documentation. Factors like team distribution, organizational maturity, business analyst expertise, and documentation usage play a more influential role. However, the business sector can still influence specific requirements and specifications.

Contextual factors significantly influence the inclusion of the Assumptions section. Out of 14 factors examined, 8 have a statistically significant impact, with 5 having a moderate influence. Factors like company size, team distribution, template usage, and using requirements documentation as a test source moderately impact the Assumptions section's use. Outstaff and Outsource IT companies prioritize assumptions due to risks in the requirements-gathering process. System integration and dependencies are weakly impacted by company type, suggesting different company types prioritize sections based on their business models and risk considerations. Larger companies emphasize documenting assumptions for a shared understanding of project goals and risks. Smaller companies have a more informal approach to documentation and risk management, resulting in less emphasis on assumptions. Distributed teams use the Assumptions section more frequently due to communication challenges and the need for standardized templates and precise documentation. In large distributed teams, the importance of documenting assumptions may be emphasized during new analyst training. Using requirements documentation as a source for tests encourages documenting assumptions. Templates provide a structured format, increasing the likelihood of including the assumptions section.

Survey data shows that 63% of respondents use templates in their documentation. The use of templates impacts the inclusion of various sections, including Assumptions and Background sections. Templates provide a standardized approach and facilities, including necessary information. However, using templates from standards like ISO/IEC/IEEE is low, indicating a need for improved and standardized templates that include all necessary sections and reflect best practices. Using templates and knowledge transfer improves project requirements documentation and the effective implementation of software development projects.

Analyst experience significantly influences the inclusion of various sections. Experienced analysts more frequently include Data models, Goals & Objectives, Non-functional requirements, Problem statements, Risks, and Usage scenarios. Emphasizing the Problem statement section becomes more prominent among experienced analysts due to their profound understanding of project requirements, ability to identify critical components, and better communication and problem-solving skills. To improve documentation quality, involving experienced analysts in creating templates and transferring knowledge to less experienced analysts is essential.

Among the top five sections in software requirements documentation, the analyst's experience statistically significant impacts the presence of the Non-functional requirements section. Other sections like Business requirements, Functional requirements, and Glossary are influenced to a lesser extent. The project category also influences the inclusion of the Non-functional requirements section, with higher percentages in Development from scratch and User interface engineering projects.

The presence of the User interface section is not significantly influenced by contextual factors, suggesting project-specific considerations and analyst preferences play a more significant role.

Using documentation as a reminder for discussions increases the likelihood of including the Deployment specifics section.

The use of documentation in customer acceptance affects the inclusion of the Constraints section, highlighting the importance of involving customers and considering their constraints.

Overall, the impact of project context factors on documentation content may vary depending on organizational culture, project complexity, and specific requirements gathering and documentation processes. Improving documentation practices, standardizing templates, and fostering knowledge transfer among analysts can lead to more effective software development projects.

There are several limitations to this study. Despite the fact that the list of sections in business analysis documents for the questionnaire was created on the basis of international standards and codes of knowledge, survey participants could interpret them in different ways. It is important to note that due to the survey's confinement to a single country, the findings cannot be extrapolated to the global software industry without additional study, despite the integration of Ukraine's IT sector into international environments, particularly outsourcing and outstaffing firms, whose employees comprised the majority (65%) of the respondents.

Future research may focus on analyzing the impact of requirements elicitation and requirements documentation techniques on the architecture of business analysis documents, as well as on the methodology for creating document templates, taking into account the context of the project and the standards of the organization.

References

1. *Project Management Institute* (2017), The PMI Guide to BUSINESS ANALYSIS. PMI, Newtown Square, Pennsylvania, 488 p.

2. Gobov, D., Yanchuk, V. (2021), "Network Analysis Application to Analyze the Activities and Artifacts in the Core Business Analysis Cycle", *Proceedings of the 2nd International Informatics and Software Engineering Conference (IISEC),* P. 1–6. DOI: 10.1109/IISEC54230.2021.9672373

3. *International Institute of Business Analysis* (2015), A guide to the business analysis body of knowledge (BABOK Guide) ver. 3., IIBA, Toronto, Ontario, Canada, 514 p.

4. Pohl, K. (2010), "Requirements engineering: fundamentals, principles, and techniques", Springer Publishing Company, 182 p., available at: https://www.bbau.ac.in/dept/dit/TM/requirementsengi.pdf

5. ISO/IEC/IEEE 29148. (2011), "Systems and Software Engineering−Life Cycle Processes−Requirements Engineering". ISO/IEC/IEE, Standard 29148:2011. 83 p., available at: https://www.iso.org/standard/45171.html

6. Jacobson, I., Booch, G., Rumbaugh, J. (1999), "Unified Software Development Process", Addison-Wesley Professional, 463 p.

7. Gobov, D., Huchenko, I. (2022). "Modern Requirements Documentation Techniques and the Influence of the Project Context: Ukrainian IT Experience", *Advances in Computer Science for Engineering and Education. ICCSEEA 2022. Lecture Notes on Data Engineering and Communications Technologies*, Vol. 134. Springer, Cham. P.260-270. DOI: https://doi.org/10.1007/978-3-031-04812-8_22

8. Rączkowska-Gzowska, K., Walkowiak-Gall, A. (2023), "What Should a Good Software Requirements Specification Include? Results of a Survey", *Foundations of Computing and Decision Sciences,* Vol.48(1), P. 57–81. DOI: https://doi.org/10.2478/fcds-2023-0004

9. Franch, X. et al. (2023), "The state-of-practice in requirements specification: an extended interview study at 12 companies", *Requirements Engineering*, 29 April. Vol. 28(3), P. 1–33. DOI: https://doi.org/10.1007/s00766-023-00399-7

10. Wagner, S. et al. (2017), "Requirements Engineering Practice and Problems in Agile Projects: Results from an International Survey", *Proc. XX Ibero-American Conference on Software Engineering (CIbSE),* Argentina, available at: https://arxiv.org/list/cs/1703?skip=450&show=2000

11. Wagner, S. et al. (2019), "Status Quo in Requirements Engineering", *ACM Transactions on Software Engineering and Methodology,* Vol.28 (2), P. 1–48. DOI: https://doi.org/10.1145/3306607

12. Abdalazeim, A., Meziane, F. (2021), "A review of the generation of requirements specification in natural language using objects UML models and domain ontology", *Procedia Computer Science,* Vol.189, P. 328–334. DOI: https://doi.org/10.1016/j.procs.2021.05.102

13. Medeiros, J. et al. (2017), "An approach based on design practices to specify requirements in agile projects", *Proceedings of the Symposium on Applied Computing (SAC 17), Association for Computing Machinery*, New York, NY, USA, P. 1114–1121. DOI: https://doi.org/10.1145/3019612.3019753

14. Heck, P., Zaidman, A. (2016), "A systematic literature review on quality criteria for agile requirements specifications", *Software Quality Journal*, Vol. 26 (1), P. 127–160. DOI**:** https://doi.org/10.1007/s11219-016-9336-4

15. Jarzębowicz, A., Połocka, K., (2017) "Selecting requirements documentation techniques for software projects: a survey study", *Proceedings of Federated Conference on Computer Science and Information Systems* (*FedCSIS, 2017), IEEE,* P. 1189–1198. DOI**:** https://doi.org/10.15439/2017F387

16. Gobov, D. (2023), "Practical Study on Software Requirements Specification and Modelling Techniques", *International Journal of Computing,* P. 78–86. DOI: https://doi.org/10.47839/ijc.22.1.2882

17. Beedle, M. et al. "Manifesto for Agile Software Development", available at: https://agilemanifesto.org/ (last accessed: 11 May 2023)

18. Al-Msie'deen, R., Blasi, A., Alsuwaiket, M. (2021), "Constructing a software requirements specification and design for electronic IT news magazine system", *International Journal of Advanced and Applied Sciences*, Vol. 8(11), P. 104–118. DOI: https://doi.org/10.21833/ijaas.2021.11.014

19. Gobov, D. (2022), "Dependencies between requirements elicitation techniques: a survey study in Ukrainian companies", *Innovative Technologies and Scientific Solutions for Industries*, Vol. 3 (21), P. 5–15. DOI: https://doi.org/10.30837/ITSSI.2022.21.005

20. Kassab, M., Neill, C., Laplante, P. (2014), "State of practice in requirements engineering: contemporary data", *Innovations in Systems and Software Engineering*, 10, P. 235–241. DOI: https://doi.org/10.1007/s11334-014-0232-4

Received 16.05.2023

Відомості про авторів / About the Authors

Гобов Денис Андрійович – кандидат технічних наук, Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського", доцент кафедри інформатики та програмної інженерії факультету інформатики та обчислювальної техніки, Київ, Україна; e-mail: d.gobov@kpi.ua; ORCID ID: https://orcid.org/0000-0001-9964-0339

Зуєва Олександра Валеріївна – кандидат економічних наук, аналітик-консультант CS Ltd., Харків, Україна; e-mail: olekszueva@gmail.com; ORCID ID: http://orcid.org/0000-0001-9661-9657

Gobov Denys – PhD (Engineering Sciences), National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Senior Lecturer at the Department of Computer Science and Software Engineering of the Faculty of Informatics and Computer Science, Kyiv, Ukraine.

Zuieva Oleksandra – PhD (Economic Sciences), Analytics Consultant at CS Ltd., Kharkiv, Ukraine.

ВИЯВЛЕННЯ ЗАЛЕЖНОСТЕЙ МІЖ КОНТЕКСТОМ ІТ-ПРОЄКТУ ТА ЗМІСТОМ ДОКУМЕНТІВ БІЗНЕС-АНАЛІЗУ

Предметом статті є практики документування вимог до програмного забезпечення в ІТ-проєктах. **Мета** роботи полягає в тому, щоб визначити, яка інформація міститься в результатах бізнес-аналізу та як контекст проєкту впливає на зміст документів бізнес-аналізу. У статті сформульовані такі **завдання**: вивчити промислові стандарти й досвід бізнес-аналітиків та інженерів у специфікації та моделюванні вимог; створити та провести опитування щодо практик документування вимог стосовно програмного забезпечення в ІТ-проєктах; з’ясувати уподобання фахівців-практиків із бізнес-аналізу щодо змісту документів; визначити, як контекст проєкту впливає на зміст документів. Застосовано такі **методи**: 324 фахівці українських компаній опитано щодо їхніх уподобань у процесі створення документів бізнес-аналізу, а також їхнього досвіду та атрибутів профілю проєкту, у якому ці техніки використовуються. Для визначення статистично значущих залежностей між контекстом проєкту та вмістом документів бізнес-аналізу було застосовано тест *хі*-квадрат незалежності та показник розміру ефекту V. Крамера. Здобуто такі **результати**: визначено перелік найбільш часто використовуваних елементів у документах бізнес-аналізу. На основі *р*-значення тесту *хі*-квадрат знайдено 78 статистично значущих зв’язків для пар "контекст проєкту – вміст документа бізнес-аналізу", для 80 з них значущість виявлених залежностей підтверджено за допомогою розміру ефекту V. Крамера. **Висновки**. Обґрунтовано, що контекст проєкту впливає на зміст документів бізнес-аналізу в ІТ-проєктах. Найбільш впливовими факторами є досвід бізнес-аналітика, розподіл команди, розмір і тип компанії, використання шаблонів і мета використання бізнес-аналітичних документів. Знайдені залежності можуть керувати вибором структури документа бізнес-аналізу та розробленням специфічних для проєкту шаблонів у процесі створення підходу до бізнес-аналізу й підходу до управління інформацією з бізнес-аналізу.

Ключові слова: документ бізнес-аналізу; інженерія вимог; контекст проєкту; V. Крамера; емпіричне дослідження.

Бібліографічні описи / Bibliographic descriptions

Гобов Д. А., Зуєва О. В. Виявлення залежностей між контекстом ІТ-проєкту та змістом документів бізнес-аналізу. *Сучасний стан наукових досліджень та технологій в промисловості*. 2023. № 2 (24). С. 39–53. DOI: https://doi.org/10.30837/ITSSI.2023.24.039

Gobov D., Zuieva O. (2023), "Identifying the dependencies between IT project context and business analysis document content", *Innovative Technologies and Scientific Solutions for Industries*, No. 2 (24), P. 39–53. DOI: https://doi.org/10.30837/ITSSI.2023.24.039