

# Survey on Risk Classification in Agile Software Development Projects in Latvia

Oksana Nikiforova<sup>1\*</sup>, Kristaps Babris<sup>2</sup>, Jānis Kristapsons<sup>3</sup> <sup>1-3</sup>Riga Technical University, Riga, Latvia

Abstract - Software development method, which does not have any faults or gaps in project implementation, has not been elaborated so far. Due to this reason, the authors have decided to perform this study to make it easier for the companies, which use one of the agile development methods, to better foresee potential risks and to deal with their consequences. The aim of the research is to identify and classify risks in agile software development methods and the related projects based on the obtained survey data. To achieve the goal, the authors have developed evaluation criteria, as well as implemented practical questionnaire in various software development companies. From the obtained survey data, the risks are classified according to various factors, i.e., the changing highest and lowest priorities and needs in various projects. Thus, the obtained research results can be applied in various areas of project development, changing the order of priority factors.

#### Keywords - Agile, Risk Classification, Software Development.

#### I. INTRODUCTION

Today, there are many different software development methods in the world, which help implement various projects from start to finish, thus ensuring fast, efficient and flexible work. One of the best known and most widely used industries is agile software development, which includes several specific methods [1], such as Scrum, Kanban, Lean and others.

However, regardless of the design method chosen, each of them has its own potential risks that make it difficult to work on projects [2]. Consequently, the study area of this article is risk analysis in agile software development. As countless software development projects take place around the world on a daily basis, people are constantly faced with potential risks and challenges. Many of the risks have already been identified over the years and ways have been devised to eliminate them. However, in this study, at the choice of the authors, five different capabilities of software development methods are specifically described, by performing risk analysis and classification according to several criteria, based on both different sources of information and data obtained from the surveyed companies.

The object of the present research is the risk capabilities of software development projects. The whole study is based on the obtained data on how different risks affect the implementation of projects. In order for companies to deal with them in a timely manner, risks need to be sorted or categorised, for example, by probability of occurrence, consequences, time consumption or other factors relevant to the implementation of projects. The aim of the study is to identify and classify risks in agile software development methods and the related projects, based on the software analysis and the obtained survey data. Depending on the implementation of the project and its specific objective or priority, the analysed risks can be categorised in different ways so that the results of the study can be applied as widely as possible. In order to achieve the expected result, it is necessary to perform the following research tasks:

- to identify potential risks to capability development based on information sources;
- to examine in practice how companies that implement software development deal with specific risks and are aware of them;
- to analyse the obtained data, classify risks and draw conclusions about the study.

The paper is structured as follows. The first section contains short introduction of the paper. The second section theoretically describes and analyses various agile software development risk categories and specific risk examples. In this way, a selection is created with the risks to be included in the company survey. The third section justifies the choice of methodology and the purpose of the questions in the questionnaire. Information on how companies react to possible problems in the implementation of projects and whether they are aware of the risks described in the work is clarified in the fourth section. In the fifth section, risks are classified according to the probability of their occurrence in different companies. Finally, conclusions are made about the obtained results and the future research perspectives are substantiated, as well as the possibilities of their use in real companies.

#### II. RISKS

Using several sources of information [2]–[6], an initial analysis of risk categories is created based on logical judgments. More than fifteen different categories are summarised and explored. However, different sources of information sometimes call the same category under a different name or use different synonyms. Consequently, there are not so many categories as they can be combined, but it must be taken

This is an open access article licensed under the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0), in the manner agreed with Sciendo.

<sup>\*</sup> Corresponding author's e-mail: oksana.nikiforova@rtu.lv

<sup>©2020</sup> Oksana Ņikiforova, Kristaps Babris, Jānis Kristapsons.

into account that they have to be explained so that people do not misunderstand them. The authors have selected ten specific categories, which are analysed below.

As soon as the initial risk categories are created to be reflected in the company questionnaire, it is necessary to add specific examples of risks to them. However, it has been found out that it is not possible to list the number of specific risks in the world, because almost every software development project has something unique. Most of them have everything in common, but the nuances are different. Consequently, either two or three risk examples have been added to each risk category, for a total of 23 specific risks. They have not been sought specifically for any of the agile software development methods, but for agile development in general, based on the fact that companies that use capability development are surveyed and it cannot be determined in advance what specific or even hybrid method they use. It is therefore wiser to include risks that may be present in all development methods. It is worth noting that the risks listed below have been minimally reworded to better fit the questionnaire and be more useful.

According to the survey conducted, 10 existing risk categories and 23 popular examples of software development risks are described. Risks and their categories are described in a random order, based on popularity and other factors.

#### A. Organisational Risks

This category includes various risks associated with the organisation and management of project development [6]. Often projects do not clearly define the software development schedule and time. Sometimes, an inexperienced project manager is hired who may not be able to explain the steps and other important information so clearly. Another problem is that team members work on several projects at the same time, which can affect the project and its development in an unfavourable way. These are just a few examples of organisational risks.

# B. Budget and Financial Risks

This category includes all risks associated with both budget planning and other necessary spending [3]. However, there must always be financial preparedness for unexpected changes in circumstances that may affect the financial situation, so it is important to plan and reserve funds [7]. In addition to the specific suitability of the product owner's payer for the implementation of the project itself, it is therefore necessary to be able to assess the possible consequences of starting the project in question. One also needs to be able to anticipate unforeseen costs for security enhancements, which are very important software products.

### C. Technology Risks

These risks characterise the problems associated with both the technical development of the final product and the use of the technology itself in the project [6]. It is necessary to be able to predict what technical equipment will be needed to implement the project, as it may be too outdated and unable to meet the necessary requirements. As a result, various product development quality losses can occur, which again affect other risk categories [7]. For example, there is an impact on organisational risks due to longer time required for development, or an impact on budgetary risks due to unforeseen costs.

#### D. Security Risks

Safety risks are all risks that endanger the development of the final product and its use in real life. The authors specifically emphasise and add to this category the risks associated with human abuse of product security and control systems. In order to have solutions to these problems, the development team must have a very good knowledge of the basic principles of software development security [2]. It is also important to protect personal data, which in most cases must not be disclosed to third parties without the user's permission. Care must also be taken to extend deadlines in order to improve safety, as this must be foreseeable at the beginning of project planning.

#### E. Human Risks

Human risk is an influential risk category in software development projects. This is an important category because no project can be implemented without human resources or a development team. However, at the same time, people risks are among the most dangerous, as they involve various staff changes and conflicts that can lead to critical consequences for the entire project. Teams often consider such risks to be insignificant and think they can be addressed, but it is important for developers to be united and without significant disagreements with each other [7]. This category also includes risks related to people's lack of knowledge in developing specific projects. Such problems can occur if team members do not assess themselves accurately enough or are unable to identify all the requirements for the development process.

# F. Requirement Change Risks

Requirement change risks include a variety of issues related to things, functions, or other features that apply to a development product and need to be changed. These requests for change most often come from the product owner or customer because there is a desire to add additional options that are invented during the project development. However, just as well, change can be invented by the development team itself or even by the project manager if there is a real need or goal [6]. But at other times, there may be situations where the requirements of the customer of the product are misunderstood, and consequently various corrections have to be made, which again affect other risk categories, such as budget, organisational or even human risks due to new conflicts.

#### G. Market and Business Risks

A category of risks such as market and business risks is also distinguished. These risks are very closely related to the sale or actual use of the final product in a human or business environment. This risk will exist in absolutely all projects [6]. The product owner must be able to determine as successfully as possible the sales and sales opportunities of the final version of the developed product. This must be taken into account, as products are designed for profit, so it is necessary to be aware of changes in market competition and conditions. It is therefore advisable to always prepare a risk reduction plan and methods.

# H. Performance Risks

The next risk category is performance risk, which already explains in the title that its function is to group all the execution risks that are encountered during the execution of projects [2]. Although projects are practically implemented by development teams, there is sometimes a need for third-party developers, who most often specialise in a specific field. Assistance is needed if the development team does not have enough time to implement the project, lacks knowledge in a specific area or is simply more profitable. However, there is a possibility that a third party may not supply your part of the product, which is a high risk. This category also includes the risk of the product non-delivery to the customer of the product.

# I. Legal Risks

Legal risks are based on various legal principles, documentation and other legal factors that may affect the progress of the project. Here are some specific examples of how the size of the final dossier may exceed what was originally intended. This would affect other risk categories, in particular those related to planning and budgeting, which would again lead to unforeseen losses. However, if the customer of the product is dissatisfied with the work done by the development team and the final product is incomplete or does not meet the requirements, then the customer does not approve the product. There may be legislative changes in the country or in a region of other scale [3], which can even lead to a chain of risks that are detrimental to the full execution of the project.

# J. Environmental Risks

This category is most common for foreign or multinational companies that need to carry out different levels of communication or orders. This includes weather-related risks that delay, for example, the delivery of hardware from around the world. As a result, the development schedule is delayed and affected [5]. In Latvia, not so often, but in other countries, natural disasters affect communication among team members who are not in the same office. This risk category is one of the most unpredictable, as it is impossible to control natural factors. In general, it can be concluded that the risk categories are closely interrelated, as many factors influence each other and, consequently, the categories interact with each other. One risk can affect even several categories, so the distribution made above with different examples may differ according to the views of different people. This is the most understandable way of categorising agile software development risk categories.

#### III. SURVEY OF RISK CLASSIFICATION IN IT COMPANIES

This section summarises risk management capabilities in software development using specific parts of the risk management framework. Based on information sources [8], the authors have compiled the most popular issues that software developers include in risk management. The most popular of them are the following:

- 1) risk description;
- 2) risk category;
- 3) risk manager;
- 4) level of risk assessment;
- 5) cause of risk;
- 6) consequences of risk;
- 7) solution strategy;
- 8) risk status;
- 9) importance of project;
- 10) date of risk occurrence;
- 11) date of risk elimination;
- 12) loss of time;
- 13) cost loss;
- 14) probability of the risk;
- 15) the impact of the risk;
- 16) the implementation of the solution strategy;
- 17) the person responsible for the solution strategy.

These parts of the framework are then included in a questionnaire to find out if in practice companies use any of them. The obtained information on the characteristics of risk framework is clarified by surveying various companies.

Like most of the questionnaires, this one has originally been intended to reflect data on specific companies that are well known or at least heard by the people of Latvia. The first question is about **the name of the company**, which helps analyse the data more easily because the questionnaire is anonymous and its data on specific companies are not reflected or disclosed in this paper. This fact does not change the findings of the survey and, as a result, several companies agreed to provide information on the risks and implementation of projects in their companies. The authors assume that if the questionnaire were not anonymous, several companies would not complete it because even under the existing rules, some companies refused to conduct the survey.

The next question is about the **number of employees in the company**. Several possible answer variants are given (1-10, 11-30, 31-50, 51-100, 101 and more employees), so that the respondents would have to spend as little time as possible for writing their answers. This question will be useful for analysing the results of the questionnaire according to the most popular methods in both large and small companies. These answers also provide insight into the differences in risk management, accounting and other factors between large and small companies.

With a similar goal as the previous one, the question of the **number of employees in the project team** is also created. Here, the answer options are given logical intervals (1-5, 6-10, 11-15, 16 and more employees). From the obtained results, it is possible to compare how the size of project teams is affected by the size of the company.

One of the most important questions of the survey is **about agile software development methods in companies**. Nine answer options are given with the possibility to add the method suggested by the respondent. With the help of this question, the most popular agile development methods among the surveyed companies are clarified. This is important to know because it is necessary to compare whether the obtained data are in line with the theory, i.e., whether *Scrum* is still considered to be the most common method. Based on the obtained results, the most dangerous risks can also be determined for each development method separately.

By asking how many projects have been completed using one of the agile software development methods, the authors find out whether these methods are used on a daily basis. If the results of the responses provide information that there are, for example, ten or fewer implemented projects, this does not immediately mean that agile development has been implemented recently or is not used frequently, as conclusions and data analysis on company size, project size and other influencing factors are needed.

A section on the main purpose of the study, which is based on classifying and identifying risks in agile software development projects, then begins with the question of whether a total of 30 **companies will take into account the potential risks in project implementation**. Here are three possible answers, which provide an opportunity to confirm or deny the question, or to note that risks are taken into account depending on the project. To this question, the authors envisage two variants – either they calculate risks or calculate them depending on the project because rarely any company thinks about problems at all.

The next question clarifies information on **risk monitoring and how it is reported within the company**. As in the previous questions, this one provides three levels of choice that cover the answers. The first option is that the company uses a variety of tools, technologies and an extensive reporting system. The second option is that the risks are partially monitored throughout the project without using a dedicated system. The final possibility is to note that the risks are not monitored and controlled at all, which will be in a very small percentage of companies.

When asked **if the company has ever encountered any of these risk categories**, respondents should mark one of the ten already given or add their own category. The authors have theoretically already obtained and analysed information on existing and several very similar categories that exist in agile software development projects around the world. This question is asked in order to achieve the goal of the study and use this as one of the risk classification criteria, later dividing the risks according to the probability of occurrence in different projects.

To answer a question similar to the previous one – whether the **company has ever encountered any of these examples of risks** – the respondent has to mark specific, previously given options. 23 specific risks have been selected, and their selection process has been described in previous sections. This can be considered as the question that most directly affects the study and its purpose, as these answers provide the most important information. It is very necessary to find out whether the examples of risks given by the authors are actually encountered by companies working in Latvia in the implementation of everyday projects. The results of this question will also be the basic criterion for risk classification.

The next question is which projects have the most popular risk mitigation activities and their categories. Compared to

many previous questions, this one is open-ended, because the authors want to know the most important and most frequently used risk mitigation activities. Initially, the authors decided to add this question to each risk noted by the respondent. But then the authors concluded that it would take too much time for the respondents to fill in the questionnaire and, consequently, they would lose interest in completing the questionnaire. Therefore, this question was not marked as mandatory. However, if the answers are obtained, then they are to be compared with the theory analysed in the previous sections.

There is also a question whether **the company has ever had to face the risk of forcibly terminating or canceling a project**. This question is also defined as a question of choice, as the authors are not sure if this happens in real life. However, one of the above risks should appear as an answer to this question, as it will prove that the chosen examples of risks are commonplace and can have a great impact on various projects.

One of the important issues is **whether the company uses any of the frameworks for risk management**. The aim of this question is to find out whether the sources of information that the authors have analysed theoretically have any risk accounting frameworks that are applied in practice. The answers to this question will allow comparing whether the risk management of large and small companies differs.

The penultimate question of the questionnaire clarifies **who** is responsible for the risks in the projects and their monitoring. Based on the answers, the authors obtain information about the responsible persons and their positions, depending on the development method, the size of the company and other influencing factors. The questionnaire concludes with information on whether managing risks in agile software development is difficult. There are three possible answers that confirm, deny, or give the opportunity to note that the complexity depends on the project and the size of the team. The authors predict that the third variant will be the most common answer because, according to the acquired theoretical knowledge, it most often happens in development projects.

#### IV. DATA ANALYSIS

In this section, the analysis of the survey results is performed, and the answers to the questions are presented with the help of graphs.

Each of the created graphs is analysed and specific conclusions are drawn, depending on the results obtained and those previously predicted. One of the main goals of this section is to help authors understand which evaluation criteria are best for risk classification.

The first question of the questionnaire provided answers about the name of the company. Since the data on the respondents are not disclosed in the paper, the authors give a visual idea of the companies, naming them Company 1, Company 2, Company 3 and so on. Responses were received from 13 companies. The obtained data represent information to be analysed with regard to risk classification.

The answers to the second question provide the information on the number of employees in each company. Fig. 1 shows that 61 percent of the surveyed companies have more than 101 employees, which means that the companies are large enough and competent to provide useful information. However, no company falls into categories of 1 to 10 and 51 to 100 employees, which will partly affect the analysis of data for companies of this size, as it will not be possible to analyse in detail the methods, risks and other influencing factors inherent in these two categories.



Fig. 1. Number of employees in the surveyed companies.

However, regardless of the number of employees in the company, it has at least one or more separate project teams to be able to implement the wishes of the customers. To find out how many employees are in the respondent's project team, the third question was asked. The obtained answer variants from all surveyed companies can be seen in Fig. 2.

More than half or 54 percent of the surveyed companies have project teams with the number of employees from 6 to 10. Exactly such a limit on the number of people in the project is defined in the Scrum method as the optimal number of people. Using the knowledge of this theory, it is concluded that most of the surveyed companies adhere to the limits defined by Scrum, 11-30 (8 %), 31-50 (31 %), 101 and more (61 %). The other categories divide the remaining 46 percent into three equal parts, or each category corresponds to two companies.

To draw conclusions about the size of project teams in relation to the size of the company, Fig. 3 is created. It compares data between the number of employees in specific companies and the number of employees in a project team. In fact, data analysis takes place between the Fig. 1 and Fig. 2.

As can be seen from the obtained results, regardless of the size of the company, project teams with 6 to 10 employees are the most common among the surveyed companies. However, the data obtained do not provide any corroborating facts, as there is only one company with 11 to 30 employees and two companies with 31 to 50 employees. Consequently, the data are not as representative as for companies with 101 employees or more, as there were a total of eight such companies.



Fig. 2. Number of employees in project teams in the surveyed companies.



Fig. 3. Number of employees in project teams according to the number of employees in the surveyed companies.

The fourth question of the questionnaire is about the use of agile software development methods in companies. Here the authors indicated the most popular agile development methods based on the information sources. Respondents also had the opportunity to add their own method if it was not among them. To make the data more transparent, the authors included in Fig. 4 only the methods that were marked in the survey or added. According to the answers given, the methods *Dynamic Systems Development Method (DSDM)* and *Crystal* were not mentioned at all. However, respondents added a method such as the *Scaled Agile Framework (SAFe)* and one unnamed hybrid method, as it was a combination of several agile development methods specific to the company's needs.

From the answers, it is possible to draw several important conclusions. Comparing the information obtained by the authors from several literature sources with the practical data obtained from the questionnaire, it is proved that the *Scrum* method is still the most common among software development companies. The *Kanban* method lags behind it minimally or only by two companies, being in the project teams of eight companies. Thirteen companies took part in the survey, but the total number of methods used reached 25. As can be seen from the previous figures, the questionnaire was filled in by companies of different size, so other answer options included several methods because rarely any company in all project teams chose the development method.



Fig. 4. Frequency of using agile development methods in the surveyed companies.

However, all the other seven methods shown in Fig. 4 were not used in more than one of the companies surveyed, so this would affect the risk classification results for these methods in particular.

The fifth question aimed at gaining a broader insight into the experience of the surveyed companies on the completed projects, implementing one of the agile software development methods. The results of these responses allow concluding that most of the surveyed companies have implemented projects in the range of 1 to 10 (see Fig. 5). Out of the surveyed companies, only three have implemented more than 101 projects with any of the capabilities of software development methods, which madekes these companies the most experienced ones. The number of completed projects in the range of 21–50 has not been implemented in any company at all.

In order to gradually start collecting data on the risks in the implementation of projects, the authors formulated the sixth question to find out whether companies took into account the potential risks in software development projects at all. The majority or 62 percent of the surveyed companies consider the potential risks and only 38 percent evaluate the significance of the project before thinking about the risks. In addition, none of the respondents gave a negative answer to this question, namely, that they did not take into account risks.

As all companies either take risks into account or decide to do so depending on the project, the next question was related to the procedure of monitoring risks in companies and project teams. According to the results of the survey, 7 out of 13 companies use various tools, technologies and extensive reporting systems to monitor risks (Fig. 6). In five companies, risks are partially monitored within the whole project, without a special system. However, in one company the risks are not monitored at all, which leads to the conclusion that there are still companies or project teams that do not pay attention to the risks until they are exposed to them.

The next question deals with one of the most important issues, namely, whether companies have ever encountered any of the given risk categories. For a better review of the data, the results obtained are shown in Fig. 7.

From the information available, it is possible to identify that four (organisational, technological, safety and human risk categories) of the ten categories are repeated in ten out of thirteen companies, which is a very high rate compared to the environmental risk category identified by only one company out of all respondents. The risks of changing claims are the second type of most common risks with 9 cases. Budget or financial risks and execution risks are the third most popular type of risk in a row, where each of them has been identified in 8 cases. Market or business and legal risks are the fourth type of most common risks, identified by 7 and 6 companies, respectively. This number is also relatively large because basically every second of the surveyed companies faces this type of risk. However, the authors did not include in the chart one category that was added by the respondent. This was a category called dependence on development partners, which could be combined with organisational risks or performance risks.



Fig. 5. Number of implemented projects in the surveyed companies using agile development methods.



Fig. 6. Risk monitoring in the surveyed companies.

The next or ninth question was of equal importance, in which the respondents had to note what specific examples of risks they encountered in the implementation of projects in their company. Data of all 13 companies are provided in Fig. 8.

For some of the risk examples, it is not possible to read the full names in Fig. 8, but only the data obtained are presented here in order to be able to see how often the risks recur among the surveyed companies. The full names of the risks can be found in Table I, in the section on risk classification, where the data are fully compared and analysed, distinguishing between different evaluation criteria.



Fig. 7. Frequency of risk categories in the surveyed companies.



Fig. 8. Frequency of risk examples in the surveyed companies.

However, based on these results, it can be immediately concluded that 10 of the 13 companies surveyed are exposed to the following three risks:

- the project development schedule is not clearly defined and understood;
- 2) team members work on several projects at the same time;
- 3) the product owner wants to add additional features to the product.

These risks are stable in comparison with the other given examples, which are analysed in more detail in the risk classification section. However, two examples from the respondents were not included in the graph, one mentioning the impact of the Covid-19 virus on the project and the other being defined as sloppy and unsuccessful. The next question of the questionnaire – what are the most popular risk mitigation activities or their categories in the surveyed companies – was answered by 62 percent of respondents, more than half. It means that not all companies are allowed to disclose such information or did not know what activities to name. This question was designed as open-ended, so the authors, summarising the obtained data, modified the answer variants in writing without changing their idea and principles. Fig. 9 shows the answers provided, from which it can be seen that the most used risk mitigation activity among the surveyed companies is holding regular meetings. This answer was followed by work planning and risk analysis, with a frequency of 2 for each activity. Other respondents provided more information indicating different risk mitigation activities implemented by companies.

The next question provides information on the risks that have once led to the forced termination of projects by one of the surveyed companies. The answers obtained give an idea of the risks, which may not have been included in the survey, but actually have significant consequences for the projects. The results show an opportunity to conclude which are the most dangerous risks in the implementation of projects.



Fig. 9. The most popular risk mitigation activities in the surveyed companies.

Nine of the thirteen companies, accounting for 69 percent of the total, have never faced specific risks leading to the forced cancellation of projects. This is to be welcomed, as the authors predicted that the results would be more negative for companies, i.e., there would be more risks leading to the termination of projects. However, the advent of the relatively new Covid-19 virus has caused project losses to the two companies surveyed. The risk of realising non-compliance with deadlines has had the same effect.

To manage risks, one way to list and summarize them is to create a framework. Therefore, the next question is whether companies use such a method at all, and if so, what is included in the framework or column names.

Fig. 10 shows the most popular risk management criteria among the surveyed companies. The number of companies that do not use any of the framework categories at all is marked in orange. Using calculations, 69 percent of the given companies use the risk framework with one of the given categories. Nine of the risk framework categories are used by more than five of the nine (using the framework) companies above the 50 percent mark.

The next question clarifies the information about the responsible persons in the respondent companies. Variants of answers are given with the possibility for the respondents to add their own variant (Project Manager – 62%, Project Management Office – 15%, Product Owner – 15%, Risk Manager – 8%). Developers and the risk committee were not included as responsible persons, as these groups were not identified in any of the companies surveyed. Respondents also

did not mention that no one was responsible for monitoring risks, which was good because risks were given more attention in project development.

According to the final question of the questionnaire that concerned the level of difficulty to manage risks in software development, respondents almost unanimously answered that it depended on the project and the size of the team. There was only one difference in the opinion, which acknowledged that it was difficult to manage risks regardless of the project or team. The information obtained is not large enough to use a visual image.



Fig. 10. Use of frameworks for risk management in surveyed companies.

#### V. RISK CLASSIFICATION

According to the data obtained from the questionnaire, there are specific facts about the most common risks and, consequently, the most popular risk categories faced by various companies working in Latvia on a daily basis. Specific criteria are needed to assess and allocate risks. According to the results of the questionnaire, the main objectives of risk classification are to determine:

- 1) examples of risks with the highest probability of joining various projects;
- 2) risk categories with the highest probability of occurrence in various projects;
- examples of risks with the highest probability of occurrence in Scrum and Kanban methods;
- 4) risk categories with the highest probability of occurrence in the Scrum and Kanban methods;
- 5) examples of risks with the highest probability of joining different projects, taking into account the size of the company;
- 6) risk categories with the highest probability of occurrence in different projects, taking into account the size of the company.

The following risk classification objectives are chosen because they are the most influential, important and relevant ones for software development companies. To determine risk examples and risk categories that are most likely to occur, they are divided into three groups according to the following scale:

- 1) low probability of occurrence (0–30 % of cases);
- 2) medium probability of occurrence (31–69 % of cases);
- 3) high probability of occurrence (70–100 % of cases).

This division of groups is made because it is easier to perceive and remember three groups than five or more. The highest to the lowest probabilities of occurrence are each assigned limits in the range of 30 percent, and the average probability of occurrence is assigned limits in the range of 40 percent. The following breakdown is made to make it easier to calculate the probability of joining and to represent the data at project management meetings.

To determine the probability of occurrence (in percentage) and, accordingly, the group to which a particular risk example or risk category belongs, the following formula is used:

# Risk probability % = $\frac{Number of responses}{Maximum responses} \cdot 100$ .

Each risk is assigned a specific probability of occurrence based on the data provided by the surveyed companies. It is assumed that if the risk example is repeated 13 times among the surveyed companies, then the risk is assigned a 100 percent probability of occurrence. Accordingly, if it is not repeated once, then a 0 percent probability of occurrence is assigned. To make the data in the tables clearer and easier to understand, each probability of occurrence uses its own colour tone:

- 1) green low probability of occurrence;
- 2) yellow medium probability of occurrence;
- 3) red high probability of occurrence.

As with the distribution of probabilities of entry, based on the obtained data on the number of employees, companies are divided into three groups according to the following scale:

- 1) small company (1–30 employees);
- 2) medium company (31–100 employees);
- 3) large company (101 employees and more).

This breakdown of enterprises is made because, as with probabilistic determination, it is easier to perceive and remember three groups rather than more, and there is not enough data to make a more detailed breakdown. Referring to the information provided in Fig. 1, a size distribution of enterprises is made; respectively each group has its own number of enterprises, i.e., there are one small enterprise, four mediumsized enterprises and eight large enterprises. As there is only one small enterprise, risks of large and medium-sized companies are analysed in more detail. These results are used and analysed in further research on risk classification. The calculated results of risk classification for risk examples are provided in Table 1, which lists all risk examples with high, medium and low probability of occurrence.

# A. Probability of Occurrence of Risk Examples in Different Projects

Referring to the data in Table I, 3 out of 25 risks have a high probability of occurrence in software development projects, as their frequency in the surveyed companies reaches 77 percent. This means that companies and project teams should pay most attention to defining the project development schedule, drawing the attention of team members to the specific project and the product owner's desire to add additional features to the product. The average probability of occurrence is marked as 10 out of 25 or 40 percent of all risk examples in the table. Average risks can be in every third or even every second company, so it is important to take them into account. Almost half or 48 percent of all risks mentioned in the work are the least likely. This does not mean that they can have fewer consequences, but they are less common among the companies surveyed.

# B. Probability of Occurrence of Risk Examples in Scrum and Kanban Methods

For these two methods alone, there is enough data from the results of the questionnaire to allow for in-depth research on risk examples. Initially, the authors intended to analyse risk examples for each of the agile development methods; however, due to insufficient results, it is not rational to do so.

According to the results in Table I, the *Scrum* method contains 4 out of 25 risk examples with a high probability of occurrence. The risk that the product owner wants to add additional features to the product is present in 9 out of 10 companies that implement the *Scrum* method. The second place (or 80 percent of cases) is taken by the risk that team members work on several projects at the same time. The third and fourth places are divided between the risks that the project development schedule is not clearly defined and understood and that new legislation is adopted. Eight out of 25 risks have a medium probability of occurrence, but most have a low probability of occurrence, which is in 13 cases. Only one of the risks did not occur at all in any of the surveyed companies, which can be seen in the last row of Table I.

The Kanban method is indicated among eight respondents, which is two cases less than the Scrum method. But even such a number of responses is sufficient to draw different conclusions about the risk examples with the highest, medium and lowest probability of occurrence for a particular Kanban method. The results obtained are shown in Table I. Compared to the Scrum method, only two examples of high-probability risks are identified, both reaching 75 percent. One is that team members work on several projects at the same time, and the other is that the project development schedule is not clearly defined and understood. Both of these risks are also at the top of the Scrum method, as these risks are most common in all agile software development methods. However, the Kanban method has two examples of risk in the medium category more than Scrum, but two less in the large category. This gives an identical low probability of occurrence for both methods, reaching 13 out of 25 risk examples.

# *C. Probability of Occurrence of Risk Examples Depending on the Size of Companies*

First, data analysis is started for examples of risks in large companies. The obtained probabilities of occurrence are provided in Table I. There is only one example of risk with a high probability of occurrence in large companies, which characterises that the project development schedule is not clearly defined and understood. This example has a high probability of joining all categories, which means that it is very popular among all project developers. However, the medium and low probability risks are the same, each reaching 12 examples in its own group. Of the companies surveyed, the large ones have five examples of risks that have a 0 percent probability of occurrence. This is one-fifth of all risks, which means that only 20 of the examples provided by the respondents are encountered in large companies. Table I provides information that five examples of risks have a high probability of occurrence for medium-sized companies, i.e., by 80 percent or four risks more compared to large companies. The most dangerous risks (in terms of joining) are team members working on several projects at the same time and the product owner wants to add additional features to the product, as their probability of joining is 100 percent or four out of four mediumsized companies face it. Medium-sized companies have only three medium-probability risks, but 17 low-probability risks, of which seven have a 0 percent probability, which only benefits companies.

# D. Probability of Occurrence of Risk Categories in Different Projects

Similar to the calculation of the probability of occurrence of examples in different projects, the probability of occurrence of risk categories is also calculated. As with risk examples, risk categories can have a maximum value of 13 and a minimum value of 0 (100 and 0 percent, respectively). All calculated data using the formula can be seen in Table II below.

According to the obtained calculations, the results are analysed with risk categories and their corresponding probabilities. Table II provides information that four of the ten categories compiled by the authors of the survey have a high probability of joining various software development projects, as they exceed a 70 percent mark.

This means that companies should pay most attention to organisational, technological, security and human risks, 47 as well as 69 percent probability of change risks, which is only one percent behind entering the most dangerous group. Environmental risks are least likely to occur in a single company. This category is the least relevant one because various natural disasters are not as common in the territory of Latvia as in other regions of the world. The company that marked this risk category is international, so there are other problems with environmental risks.

The *Scrum* method has a high probability of occurrence in 60 percent of all risk categories, which is a record high result compared to the previous valuation methods. Human risks and security risks are present in nine out of ten companies that implement *Scrum*.

Human and security risks are followed by risk change requirements in 80 percent of cases and budgetary, organisational and technological risks in 70 percent of cases. There are only three categories with an average probability of joining between 50 and 60 percent.

	Criteria: Different projects, Scrum, Kanban, large		nies, medium c	ompa	nies (probat		n percent (%	b), probability rank fr		om low to high)		
No	Risk		Different projects		Scrum		Kanban		Large companies		Medium companies	
			Rank	%	Rank	%	Rank	%	Rank	%	Rank	
1	The project development schedule is not clearly defined and understood	77	High	70	High	75	High	75	High	75	High	
2	Team members work on several projects at the same time	77	High	80	High	75	High	63	Medium	100	High	
3	The product owner wants to add additional features to the product	77	High	90	High	63	Medium	63	Medium	100	High	
4	The development team lacks the necessary knowledge to develop the product	62	Medium	50	Medium	63	Medium	63	Medium	50	Medium	
5	Understanding the requirements of the product customer	54	Medium	60	Medium	50	Medium	50	Medium	75	High	
6	New legislation is being adopted		Medium	70	High	50	Medium	63	Medium	50	Medium	
7	Inexperienced project manager		Medium	50	Medium	63	Medium	63	Medium	25	Low	
8	Allowing loss of product quality		Medium	50	Medium	50	Medium	50	Medium	25	Low	
9	Disagreement among team members		Medium	40	Medium	63	Medium	50	Medium	50	Medium	
10	Changes in competition and market conditions	38	Medium	40	Medium	25	Low	50	Medium	0	Low	
11	Non-delivery of third-party products	38	Medium	30	Low	38	Medium	38	Medium	25	Low	
12	The customer of the product does not approve the final product	38	Medium	50	Medium	38	Medium	50	Medium	25	Low	
13	Failure to fulfill the customer's expectations	31	Medium	40	Medium	25	Low	13	Low	75	High	
14	Insolvency of the product owner for the implementation of the project	23	Low	30	Low	25	Low	25	Low	25	Low	
15	Unforeseen costs to improve product safety	23	Low	20	Low	13	Low	13	Low	25	Low	
16	The hardware is too outdated	23	Low	30	Low	38	Medium	38	Medium	0	Low	
17	Hacker intrusion into security and management control systems	15	Low	20	Low	25	Low	25	Low	0	Low	
18	Realisation of development product sales opportunities	15	Low	10	Low	0	Low	0	Low	25	Low	
19	Natural disasters affect communication among team members located in different regions of the world	15	Low	20	Low	25	Low	25	Low	0	Low	
20	In order to improve security, the development deadline is extended	8	Low	10	Low	13	Low	13	Low	0	Low	
21	Possibility of practical product development implementation	8	Low	10	Low	0	Low	0	Low	25	Low	
22	The size of the final dossier exceeds the originally intended one	8	Low	10	Low	0	Low	0	Low	25	Low	
23	Careless or unsuccessful task assessment	8	Low	10	Low	0	Low	0	Low	25	Low	
24	Effects of Covid-19	8	Low	10	Low	13	Low	13	Low	0	Low	
25	Due to the weather, the delivery of hardware from different regions of the world is delayed	0	Low	0	Low	0	Low	0	Low	0	Low	

 TABLE I

 COMPARISON OF RISK PROBABILITY BY DIFFERENT CRITERIA

 n, Kanban, large companies, medium companies (probability in percent (%), probability rank from low to high)

Criteria: Differ ent projects So

Chieffand Directory projects, Schun, Randall, Range Companies, mediani companies (producinty in percents (77), producinty tank non low to mgn)											
N	Risk category	Different projects		Kanban		Scrum		La	rge companies	Medium companies	
INO		%	Rank	%	Rank	%	Rank	%	Rank	%	Rank
1	Organisational risks	77	High	63	Medium	70	High	63	Medium	100	High
2	Technological risks	77	High	75	High	70	High	88	High	50	Medium
3	Security risks	77	High	88	High	90	High	75	High	100	High
4	Human risks	77	High	63	Medium	90	High	63	Medium	100	High
5	Risks of changing claims	69	Medium	75	High	80	High	63	Medium	100	High
6	Budgetary / financial risks	62	Medium	38	Medium	70	High	63	Medium	50	Medium
7	Execution risks	62	Medium	50	Medium	50	Medium	63	Medium	50	Medium
8	Market / Business risks	54	Medium	25	Low	60	Medium	50	Medium	50	Medium
9	Legal risks	46	Medium	38	Medium	60	Medium	38	Medium	75	High
10	Environmental risks	8	Low	13	Low	10	Low	13	Low	0	Low

TABLE II COMPARISATION OF RISK CATEGORIES BY DIFFERENT CRITERIA Criteria: Different projects, Scrum, Kanban, large companies, medium companies (probability in percents (%), probability rank from low to high)

Here, too, environmental risks are the least likely to occur. To make the study more successful, a comparison of risk categories is made between the *Scrum* and *Kanban* methods. The probabilities of occurrence provided in Table II can be used to analyse the most common risk categories specific to the *Kanban* method. The biggest risk is to face the security risks that come first for both methods. They are followed by changes in requirements and technological risks with a probability of 75 percent, which are also in the most dangerous category for the *Scrum* method. However, the *Kanban* method has two risk categories that fall into the lowest probability of occurrence, leaving exactly five risk categories in the middle category.

# *E. Probability of Occurrence of Risk Categories Depending on the Size of Companies*

First, the risk categories of large companies are summarised and analysed, which can be seen in Table II. According to Table II, there are two categories with a high probability of occurrence – technological and security risks. Almost all of the other ten categories, with the exception of environmental risks, have a medium probability distribution, which can also have consequences for the operation of projects.

Next, the risk categories of medium companies are summarised and analysed. These results can be used to compare the risk categories of medium and large companies. Mediumsized companies predominate in the three categories with a high probability of entry. Both risk categories and risk examples are more likely to occur in medium-sized companies, which means that large companies are less likely to be exposed to specific risks or categories. It can be concluded that medium-sized companies need to be more careful and more prepared against various risks. The only drawback to this conclusion is that the number of respondents working at medium-sized companies is twice as small as in large companies.

# VI. CONCLUSION AND FUTURE RESEARCH

At the beginning of the article, the main goal has been defined, i.e., to identify and classify risks in capability software development methods and the related projects based on the analysis of agile software and the obtained survey data. Theoretical analysis of the topics related to the aim of the research and practical implementation of the set tasks have enabled the authors to achieve the expected result and obtain useful information about various software development companies.

In order to achieve the aim of the research, it was necessary to start with the analysis of capability development and their methods. Theoretical risk accounting and analysis of software development capabilities were performed using information sources. Then, the practical part of the research started by conducting surveys of 13 companies in Latvia about the risks in project implementation. The main achievement of the research is the creation of risk classification tables with different categories, which provide information about possible communication problems and difficulties in the implementation of projects by software development companies.

Very useful and high-quality data were obtained from the surveyed respondents, which could be further analysed from other points of view, obtaining new risk classification results.

Based on the information obtained in the study and the practical part, the following conclusions are drawn:

- based on the results of the survey, it is possible to identify specific examples of risks and risk categories with the highest probability of occurrence in software development projects;
- 2) security risks are always highly probable compared to other risk categories;
- the *Scrum* method is more likely to occur in any of the given risk examples or categories compared to the *Kanban* method;

3088

 large companies are only 2 percent more likely to have probability of occurrence of risk examples than mediumsized ones.

Despite this study has been conducted with the aim of classifying risks based on the survey of IT companies in Latvia, the analytical part is applicable to any other geography and the results obtained in practice can be used by any software development company that applies agile software development and is willing to know about previously unheard risks or their solutions.

Based on the classification of risks, companies can view the data and draw conclusions about their projects. Risk monitoring framework also enables companies to manage risks more efficiently and conveniently.

#### REFERENCES

- D. Duka, "Adoption of Agile Methodology in Software Development," in 2013 36th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), IEEE, 20–24 May 2013, Opatija, Croatia, 2013.
- [2] A. Albadarneh, I. Albadarneh, & A. Qusef, "Risk Management in Agile Software Development: a Comparative Study," in 2015 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT), 3–5 Nov. 2015. Amman, Jordan: IEEE, 2015. https://doi.org/10.1109/AEECT.2015.7360573
- [3] H. Andrat, & S. Jaswal, "An Alternative Approach for Risk Assessment in Scrum," in 2015 International Conference on Computing and Network Communications (CoCoNet), 16–19 Dec. 2015. Trivandrum, India: IEEE, 2016. <u>https://doi.org/10.1109/CoCoNet.2015.7411239</u>
- [4] R. C. Williams, J. A. Walker, & A. J. Dorofee, "Putting Risk Management into Practice," *IEEE Software*, 1997, vol. 14, no. 3, pp. 75–82. <u>https://doi.org/10.1109/52.589240</u>
- [5] K. Ghane, "Quantitative Planning and Risk Management of Agile Software Development," in 2017 IEEE Technology & Engineering Management Conference (TEMSCON), 8–10 June, 2017. Santa Clara, CA, USA. <u>https://doi.org/10.1109/TEMSCON.2017.7998362</u>
- [6] M. Hammad, I. Inayat, & M. Zahid, "Risk Management in Agile Software Development: A Survey," in 2019 International Conference on Frontiers of Information Technology (FIT), 16–18 Dec. 2019. Islamabad, Pakistan: IEEE, 2020. <u>https://doi.org/10.1109/FIT47737.2019.00039</u>
- [7] D. Raphael-Rene, Common Risks in Agile Projects and How to Deal with Them [Online]. 2018 [accessed 23 May 2020]. Available from: https://www.gratasoftware.com/common-risks-agile-projects-deal.
- [8] Project Management Docs [online]. [Viewed 10 April 2020]. Available from: https://www.projectmanagementdocs.com/#axzz60DpjFhs9



**Oksana Nikiforova** received the Doctoral degree in information technologies (system analysis, modeling and design) from Riga Technical University, Latvia, in 2001. She is presently a Professor at the Department of Applied Computer Science, Riga Technical University. Her current research interests include agile software development methodologies and project management methods and tools.

E-mail: <u>oksana.nikiforova@rtu.lv</u> ORCID iD: <u>https://orcid.org/0000-0001-7983-</u>



Kristaps Babris received the Master degree in computer systems from Riga Technical University, Latvia, in 2018. He is presently the second year PhD student at the Department of Applied Computer Science, Riga Technical University. In parallel, he is a CTO at the company which develops Business Intelligence solutions. His current research interests include design and modeling.

E-mail: kristaps.babris@rtu.lv



Jānis Kristapsons is a recent graduate of Riga Technical University with progressive experience in the IT industry. He has a Bachelor degree in computer control and computer science.

E-mail: janis.kristapsons@edu.rtu.lv