









Automated Support for Risk Management in Scrum Agile Projects

Samuel de Souza Lopes¹  , Rogéria Cristiane Gratão de Souza² ,
Allan de Godoi Contessoto² , André Luiz de Oliveira³ ,
and Rosana Teresinha Vaccare Braga¹ 

¹ Institute of Mathematical and Computer Sciences,
University of São Paulo, São Carlos, SP, Brazil
samuel.lopes@usp.br, rtvb@icmc.usp.br

² Department of Computer Science and Statistics, São Paulo State University,
São José do Rio Preto, SP, Brazil
{rogeria.souza,allan.contessoto}@unesp.br

³ Department of Computer Science, Federal University of Juiz de Fora,
Juiz de Fora, MG, Brazil
andre.oliveira@ice.ufjf.br

Abstract. The importance of software in the modern world entails the need to develop technologies that make the software development process more agile. Agile software development approaches have been proposed to deal with constant changes in project requirements. In Scrum, the Product Owner manages such changes so that the developed software brings significant value to the customers. However, there are potential risks involved in Product Owner responsibilities that, if not properly managed, can lead to project failure and significant financial losses. In this paper, we introduce RIMPRO-AST, an automated tool support to manage risks involving the product owner. The automation of risk management processes is crucial since the lack of computational support imposes barriers to the success of risk management activities in agile projects. RIMPRO-AST supports the process defined in RIMPRO risk management framework to guide Scrum teams to manage risks involving Product Owner roles. The results obtained through the evaluation of RIMPRO-AST with potential users indicate its effectiveness in speeding up and controlling risk management activities. Therefore, our study demonstrates that RIMPRO-AST can be used to minimize the threats and their risks and maximize the opportunities that might arise throughout Scrum software development projects.

Keywords: Agile approach · Scrum (software development) · Product owner · Project management · Risk management

1 Introduction

Agile methods, such as Scrum, as well as agile practices, like “release early” and “release often”, are well established in software development and address the

limitations of waterfall models [11,28]. Scrum is an evolutionary, corrective, and self-adaptive method for managing software development processes. Although Scrum has been created in 1993, it started to gain notoriety from the beginning of the 21st century due to the changes that the Agile Manifesto brought to the Software Engineering field, with the introduction of novel methods to manage and develop software [5,9,13]. However, the benefits of agile methods concerning evolution and self-adaptiveness have been faced with skepticism due to their emphasis on contrary ideas from traditional software engineering, such as scarce software documentation and prioritization of project changes.

The uncertainty and active participation of stakeholders in software projects contributed to the adoption of Scrum and other agile methods, but risk management has been neglected or partially supported in agile methods. Although risk management is gaining importance among organizations, risks may arise and should be managed throughout the life cycle of the project [9,14,30]. In Scrum, the Product Owner has a major role in defining the requirements to address the customer needs and leading the project. For Product Owner decision-related risks to be properly managed, it is necessary to incorporate traditional risk management approaches within agile methods [15,30].

In a previous work from the authors [18], a framework for risk management related to Product Owner has been proposed, named Risk Management Product Owner (RIMPRO). In this paper, we propose RIMPRO Automated Support Tool (RIMPRO-AST), built upon RIMPRO. The idea is to provide an automated support for risk management involving the Product Owner in agile projects. Automated tools offer support to professionals in performing risk management processes, such as those foreseen by RIMPRO [12,16].

The remainder of this paper is organized as follows. Section 2 introduces the basic concepts needed for the reader to understand the contributions of this work. Section 3 provides an overview of RIMPRO [18]. Section 4 introduces the RIMPRO-AST tooling support for RIMPRO framework. Section 5 describes the evaluation of RIMPRO-AST with users, and it presents the results. Section 6 discusses the related works. Finally, Sect. 7 highlights the conclusions and future work.

2 Background

In this section, we introduce the concepts of agile methods and Scrum (Sect. 2.1), and risk management in agile projects (Sect. 2.2), to provide the basis for the reader to understand this work.

2.1 Agile Methods and Scrum

Agile methods are a way of developing software that complies with a set of principles defined in the Agile Manifesto [5]. Those principles emphasize that the skills of the development team should be recognized and exploited so that members develop their ways of doing the job. The priority is to deliver software to the customer incrementally. Project documentation is minimized due to the

use of informal communication among team members. For those increments to be developed quickly, customers need to be involved in the process to provide quick feedback on the evolution of the software, as well as to inform which requirements should be prioritized by subsequent increments [27].

Several agile methods, e.g., Scrum, have been recognized among software development organizations [23]. Scrum is an agile method for project management with an emphasis on software development projects [28]. The underlying philosophy of Scrum recognizes that the customers often change their opinion about the product they want and that the development challenges are unpredictable by their nature [7]. Since the problem being solved cannot be fully understood from the beginning, Scrum emphasizes maximizing the ability of the development team to quickly deliver in response to emerging customer requirements. Scrum focuses on incremental software development. The set of all software requirements is called *Product Backlog*, and the set of implemented requirements in each *Sprint*, an iteration in which an increment is delivered to the customers, is called *Sprint Backlog*. The *Sprint Backlog* is defined during the *Sprint Planning* meeting, where the Product Owner describes the highest priority features for the Scrum Team. Scrum has an adaptive and self-corrective approach to review the increments implemented in each Sprint and to check possible improvements in the processes used to manage the project in the *Sprint Review*, and *Sprint Retrospective* meetings, respectively [27]. Although Scrum is one of the most adopted agile methods in the industry, it does not provide support for formal risk management that encompasses planning, analysis, risk response plan processes [30].

2.2 Risk Management in Agile Projects

According to Pritchard [22], risk management is a method for identifying and controlling areas or events that have the potential to cause unwanted changes. The Guide to the Project Management Body of Knowledge (PMBoK), in its sixth edition [21], defines risk management as a set of processes encompassing planning, identification, analysis, planning of responses, and control of project risks. In PMBoK Guide, risk management is a knowledge area composed by seven processes: **Plan Risk Management** – define how to conduct risk management activities for a project; **Identify Risks** – identify individual project risks as well as the sources of overall project risk, and documenting their characteristics; **Perform Qualitative Risk Analysis** – prioritize individual project risks for further analysis or action by assessing their probability of occurrence and impact, among other characteristics; **Perform Quantitative Risk Analysis** – numerically analyze the combined effect of identified individual project risks and other sources of uncertainty on the overall project objectives; **Plan Risk Responses** – develop options by selecting strategies and agreeing on actions to address overall project risk exposure, as well as to treat individual project risks; **Implement Risk Responses** – implement agreed-upon risk response plans; and **Monitor Risks** – monitor the implementation of agreed-upon risk response plans, tracking identified risks, identifying and analyzing new risks, and

evaluating risk process effectiveness throughout the project. Due to the lack of standardization of the term “risk”, the definition provided by the PMBoK is used throughout this paper. In this definition, the risk is an “*event or an uncertain condition that, if occurs, can result in positive (opportunities) or negative impacts (threats) in one or more project objectives, such as scope, time, cost, and quality*” [21].

In agile project management, several projects have uncertainties and risks due to their susceptibility to changes. To ensure that risks are well understood and treated, projects managed through adaptive approaches make use of frequent reviews of work products and multi-functional project teams to accelerate communication and knowledge sharing. Such risks can be managed through traditional risk management processes, as long as they are adapted to the context of agile development [1, 2]. Eventually, several risks remain unknown since they are ignored throughout the project life-cycle. Thus, it is necessary to introduce risk management processes within agile development [2]. In this context, the Project Management Institute (PMI), together with the Agile Alliance, developed the Agile Practice Guide. This guide provides tools, situational guidance, and an overview of the available agile approaches to obtain better results throughout the project. The Agile Practice Guide assists traditional project teams aiming to apply agile development concepts to their projects. Although the support provided by Agile Practices Guide to traditional teams adopting agile practices, it does not support changes or modifications to PMBoK processes or knowledge areas, such as risk management [1], thus, justifying the relevance of the research presented in this paper.

3 RiSk Management PRoduct Owner (RIMPRO)

In this section, we describe the RIMPRO framework, which introduces risk management within Scrum agile software development processes. This section provides a summary of RIMPRO, whose details are available in [18].

RIMPRO is a risk management framework that introduces activities to manage risks related to the Product Owner roles and decisions in agile projects into the Project Management Body of Knowledge (PMBoK). RIMPRO guides traditional teams that intend to adopt agile practices in their projects, which is a common practice in the current software projects so that the teams can combine Scrum principles with a structured risk management process [1].

The Product Owner plays an essential role throughout the project life-cycle, with the responsibility of managing requirements, as well as ensuring that the software brings significant value to customers. Due to her/his importance to the project, the identification and analysis of the risks associated with Product Owner decisions become necessary. The previous knowledge of the risks related to PO decisions may contribute to the success of the project [28]. Project information such as budget, schedule, and stakeholders, as illustrated in Fig. 1, are inputs to the execution of RIMPRO risk management processes:

- **Risk Management Planning (Sect. 3.1):** It defines **how** project risk management processes will be conducted;
- **Risk Identification (Sect. 3.2):** It **identifies** the project risks as well as its characteristics based on the analysis of the documentation;
- **Risk Analysis (Sect. 3.3):** here, the team members **prioritize** the risks identified and documented for additional actions to be undertaken throughout the Sprint;
- **Risk Response Planning (Sect. 3.4):** **Development** and **selection** of strategies, and **agreement** on the actions to be taken to maximize opportunities and minimize the threats to the project objectives;
- **Risk Response Implementation (Sect. 3.5):** **Implementation** of the agreed risk response plan to ensure that the risk management planning structured in the previous process will be executed; and
- **Risk Monitoring (Sect. 3.6):** **Monitoring** the execution of risk response plans to the prioritized risks, track the identified risks, identify and analyze newer risks, and evaluate the effectiveness of the risk management processes through the project.

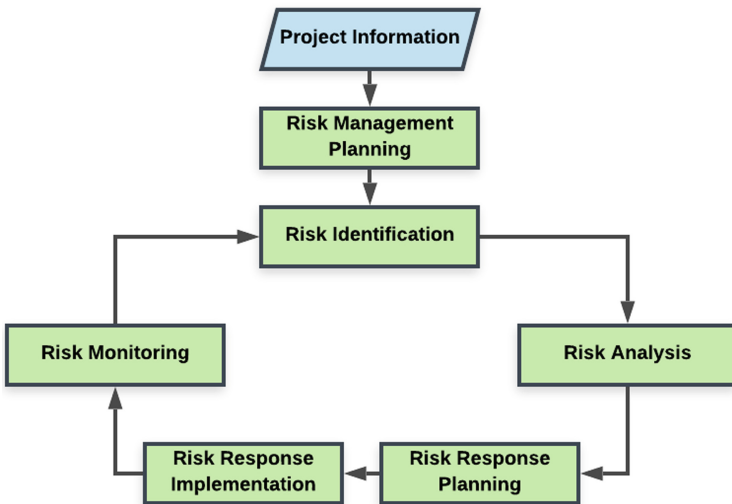


Fig. 1. Relationship between RIMPRO processes. Extracted from [18].

For the correct application of RIMPRO, all stakeholders must participate in the proposed processes since their knowledge must be gathered throughout the execution of risk management processes [20,26]. Moreover, given that risks involving the Product Owner can arise throughout the project life-cycle, the processes foreseen by RIMPRO should be iteratively executed on all Sprints. We emphasize that all the documentation provided by the framework must be created and reviewed during the Sprint.

3.1 Risk Management Planning

During the Risk Management Planning, we define how the project risk management should be conducted. This process must be performed at the beginning of the project, before the first Sprint Planning Meeting, since risks may arise while the Product Owner performs functions throughout the entire project. At the beginning of the project, key definitions are established, such as who is the individual responsible for project risk management. This individual, named “Risk Master”, must ensure that all Scrum Team members are performing the risk management processes foreseen by RIMPRO, as well as managing the planning documents. As this framework focuses on risk management involving the Product Owner, the Risk Master should be represented by the Product Owner itself for two main reasons: i) the Product Owner is the most important member of the Scrum Team for risk management [29]; and ii) the risks can be related to the client and the Product Owner is the most suitable member to treat them, since (s)he has direct contact with the customer [27]. In addition to the Risk Master, other assignments must be done such as defining the roles and responsibilities of the Scrum Team members, deadlines to establish how often the risk management processes will be carried out throughout the project life-cycle, the maximum amount or volume of risks that stakeholders are willing to tolerate (*stakeholder risk appetite*), and budget.

At the end of the process, all agreed definitions should be included in the Risk Management Plan, which describes how risk management processes are structured and executed. To not degenerate Sprint’s goal, changes to the Risk Management Plan must be requested through the Sprint Retrospective, as this meeting makes adjustments to Scrum Team to improve its work [27].

3.2 Risk Identification

Here, the risks are identified, and their characteristics are documented. All stakeholders, including customers, should be encouraged to suggest new risks at any time throughout the project due to the susceptibility of the project to uncertainties [1]. To support this activity, the risks are documented in the Project Risk Backlog, which contains among other information, the probability of occurrence, and the impact of each identified risk. As the focus is to manage the risks involving Product Owner roles, each risk should be classified through the following taxonomy presented in [18], which is specific to Product Owner-related risks:

- **Requirements:** Risks that may arise when the Product Owner does not correctly perform his/her duties in the requirements engineering stage;
- **Software Quality:** Risks related to the lack of quality (clarity, conciseness, completeness, testability) of the software developed;
- **Migration to Scrum:** Risks related to the inherent insertion of characteristics of agile approaches in the context of teams that employ traditional software engineering techniques;
- **Not Defined:** Risks that do not fit into any of the categories above.

3.3 Risk Analysis

The risks are qualitatively analyzed and the Project Risk Backlog is updated for additional action. As the purpose of this process is to prioritize the risks that will be monitored during the Sprint, risk analysis should be performed throughout the Sprint Planning since the Sprint goals are defined in this meeting [28]. The risks are analyzed using the Risk Planning Poker [18] technique, an adaptation of Planning Poker to risk management. Risk analysis is performed anonymously among Scrum Team members based on the Delphi [8] technique, used to reach a consensus among experts while preserving their anonymity.

After Risk Planning Poker, the Risk Master creates the Sprint Risk Backlog with the list of all monitored risks of a particular Sprint. Each Sprint has a list of risks that can affect the success of the iteration. Since the Scrum Team has few members, lean documentation, and a limited budget, the Sprint Risk Backlog should contain few risks to avoid a significant increase in additional project work [28]. To facilitate the monitoring of the Sprint Risk Backlog, a probability and impact matrix should be used [21]. Therefore, risks are normalized to values in the range of 0 to 1 using min-max normalization [10]. Such scaling is adopted by (1), where $risk_normalized_i$ is the value of $risk_i$ normalized to a value contained in the range [0, 1], $risk_i$ is the probability of occurrence of the i^{th} risk calculated on Risk Planning Poker, and max and min are the values of the highest and lowest card in the deck used during the analysis of $risk_i$, respectively. After normalization, the Risk Master should define probability ranges for the categories (e.g., very low, low, moderate, high, and very high) and exhibit them in a probability and impact matrix as shown in Fig. 6 [21]. The responses to the risks that make up the Sprint Risk Backlog are defined in the subsequent process.

$$risk_normalized_i = \frac{risk_i - min}{max - min} \quad (1)$$

3.4 Risk Response Planning

Here, the team members develop strategies and actions to maximize opportunities and minimize the threats to the project objectives [1]. This process is performed after Risk Analysis once the risks of Sprint Risk Backlog have already been defined, but their respective answers have not been elaborated yet.

Risk responses should be developed in collaboration with all stakeholders, including customers with knowledge in the application domain and managers. Planned responses should be proper to the relevance of the risk, cost-effective to meet the challenges, realistic within the project context, agreed between all stakeholders, and have a designated stakeholder. In general, it is necessary to select the most suitable response to risk among the diverse possible available options. The Risk Master should mediate this process before the beginning of the Sprint since the responses to certain risks may vary throughout the project, i.e., risk responses for a given Sprint may not be appropriate for subsequent Sprints [1].

For each risk from the Sprint Risk Backlog, the strategy or mix of response strategies of greater efficiency should be selected, including major and secondary strategies as needed. If the major strategies do not take effect, the possibility of applying the secondary strategies should be evaluated. Another point to emphasize is the secondary risks. For these risks, a surplus may be allocated for time or cost contingencies, as well as the identification of the conditions that trigger the use of these surpluses [1]. At the end of the process, the Risk Master should update the Sprint Risk Backlog response lists, and start the subsequent process.

3.5 Risk Response Implementation

The risk response plans that compose the Sprint Risk Backlog are implemented by the team members to ensure that risk responses are carried out as planned. Attention to this process will ensure that the responses (measures) to the agreed risks are implemented. Tools and techniques can be used for implementing the risk response plans associated with the Sprint Risk Backlog, such as [1]:

- **Expert Opinion:** Third part expert opinion with specialized knowledge should be considered by the Scrum Team members to validate or modify responses to risks, and if necessary, to decide how to implement them most efficiently;
- **Interpersonal and Team Skills:** Among the interpersonal and team skills that can be used in this process, the main one is influence. Some risk response actions may be owned by people outside the Scrum Team or who have other conflicting demands. It is necessary, at certain points of the project that the Risk Master takes influence to encourage the appointed risk owners to take the necessary measures when appropriate;
- **Project Management Information System:** An information system is recommended to support project management, including schedule, resource, and software cost to ensure that the agreed risk response plans and their associated activities are integrated within other project activities.

If any response is modified throughout the process, the Sprint Risk Backlog should be updated [1].

3.6 Risk Monitoring

Here, the team members monitor the implementation of the risk response plans contained in the Sprint Risk Backlog, and the risks that may affect the Sprint, and assess the effectiveness of risk management processes throughout the Sprint. This process is performed throughout the Sprint since risks may arise throughout the whole project life-cycle [21].

The step of evaluating the effectiveness of the risk management processes proposed by RIMPRO is carried out during the Sprint Retrospective meeting, since this meeting aims to verify the successful measures (actions), what can be improved and what actions will be undertaken to improve several aspects that

may limit the speed of the project, such as deficiencies in the risk management processes. Such an evaluation must be performed in the presence of all Scrum Team members at the Sprint Retrospective meeting, as it is the moment when the whole team must present the lessons learned from each Sprint for taking the benefits for future projects and subsequent Sprints of the current project. Therefore, plans to improve risk management processes can be established and further applied to Sprints and subsequent projects [21,27].

To ensure that the stakeholders are aware of the current risks, the Sprints should be continuously monitored. Risk Monitoring uses project information to determine whether the responses to the implemented risks are effective, the current project risks have been changed, the status of individual risks identified in Sprint has been changed, among others [1].

Risk reviews are scheduled regularly and it should examine and document the effectiveness of the risk responses made in the Sprint Risk Backlog. Risk reviews can also result in the identification of newer risks, including secondary risks arising from responses to agreed risks, reassessment of current risks, closing out risks that are out of date, identification of problems that arise as a result of risks that have occurred, and identification of lessons learned for implementation in subsequent Sprints or similar projects in the future. The Sprint risk review should be conducted as part of a regular project status meeting, such as the Daily Scrum [1,28].

4 RIMPRO Automated Support Tool (RIMPRO-AST)

The automation of RIMPRO risk management framework was implemented as RIMPRO-AST, an integrated module to the System to Aid Project Management (SAPM), previously developed by the Software Engineering Research Group from São Paulo State University (UNESP). SAPM is an automated web-based tool to support the execution of project management activities in conformance to the PMBoK guide best practices [17].

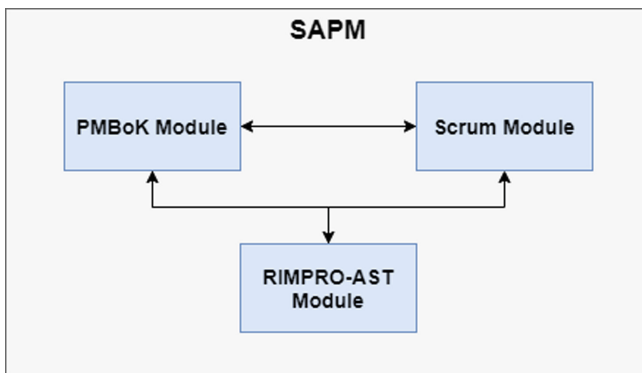


Fig. 2. SAPM architecture. Elaborated by the authors.

Figure 2 shows a general overview of SAPM tool architecture. RIMPRO-AST is attached to the existing Scrum module into SAPM [19] so that all the created projects and their respective Sprints are integrated into RIMPRO-AST. Project Sprints are linked to RIMPRO-AST, allowing the users to allocate risks to each Sprint via Sprint Risk Backlog.

The RIMPRO-AST capabilities and the screenshots of their respective graphical user interfaces are described in the following subsections. The side and top menus were removed to improve the legibility of RIMPRO-AST capabilities illustrated in the screenshots.

4.1 Notes Board

This capability presents information about the risks from the Project Risk Backlog and the Sprint Risk Backlog to users. Moreover, if the probability of occurrence of a given project risk increases, a warning is inserted in the board as illustrated in Fig. 3.

Notes Board



Note	Date	Time	Actions
'Very large project team' risk has been inserted in 'Activity 1' sprint	26/10/2021	20:21	View Risk 
'Changes during the sprint' risk has been inserted in 'Activity 1' sprint	26/10/2021	20:21	View Risk 
'Failure to prioritize requirements' risk has been inserted in 'Activity 2' sprint	26/09/2018	08:38	View Risk 

Fig. 3. Notes board interface. Elaborated by the authors.

4.2 Risk Management Plan

It allows users to define the general aspects of RIMPRO-AST risk management so that each project Sprint has a unique version of the Plan. This capability also allows users to visualize the Risk Management Plan in a Portable Document Format (PDF) file.

4.3 Project Risk Backlog

This capability allows users to visualize the list of all project risks that have not been allocated yet to a particular Sprint in a web-based interface. Alternatively, it also allows users to generate a report with all project risks in the .pdf format. Figure 4 shows the Project Risk Backlog web-based interface where the risks highlighted in red represent threats, and the risks highlighted in green represent opportunities.

Project Risk Backlog

Manage Project Risk Backlog risks

Insert risk +







Created in	Name	Classification	Probability	Impact	Actions
21/09/2018 at 11:02	Lack of communication	Requirements	0.9	0.7	  
21/09/2018 at 11:02	Antecipated sprint	Quality	0.4	0.9	  

Fig. 4. Project risk backlog interface. Elaborated by the authors.

4.4 Sprint Risk Backlog

This capability lists to the user all project risks allocated to a given Sprint. Since each Sprint has a unique Sprint Risk Backlog, this view (see Fig. 5) allows the user to monitor project risks throughout the Sprint. This feature also allows the user to register a risk into a given Sprint Risk Backlog. To assign a project risk to a specific Sprint the user should: view the risks in the Project Risk Backlog, and select the desired Sprint.

Sprint Risk Backlog

Manages project sprint risks

Select one sprint

View the risks of this sprint

'Activity 2' sprint risk




Created in	Name	Classification	Probability	Impact	Actions
26/09/2018 at 08:38	Failure to prioritize requirements	Requirements	0.5	0.7	  

Fig. 5. Sprint risk backlog interface. Elaborated by the authors.

4.5 Probability and Impact Matrix

It is a graphical visualization of combinations among probability and impact that result in a probabilistic risk rating into low, moderate, higher priority categories. Each Sprint has a unique matrix to represent the probabilities and impact of each risk. Figure 6 shows an example of Probability and Impact Matrix, in which the low, moderate, and high priorities are represented by shades of green, yellow, and red respectively. The numbers represent the number of risks in each interval of probability/impact. For example, the number 1 in Fig. 6 indicates that one risk (in this case, *Failure to prioritize requirements*) is in the red zone.

Probability and Impact Matrix - 'Activity 2' Sprint

Probability	Threats					Opportunities					Probability
0.9 - Very High	0	0	0	0	0	0	0	0	0	0	0.9 - Very High
0.7 - High	0	0	0	0	0	0	0	0	0	0	0.7 - High
0.5 - Regular	0	0	0	0	1	0	0	0	0	0	0.5 - Regular
0.3 - Low	0	0	0	0	0	0	0	0	0	0	0.3 - Low
0.1 - Very low	0	0	0	0	0	0	0	0	0	0	0.1 - Very low
	0.05 Very Low	0.10 Low	0.20 Regular	0.40 High	0.80 Very High	0.80 Very High	0.40 Alto	0.20 Regular	0.05 Very Low	0.10 Low	
	Negative Impact					Positive Impact					

Fig. 6. Probability and impact matrix interface. Elaborated by the authors.

4.6 Risk Search

It allows users to search for risks documented in previous projects using RIMPRO-AST, in which the user was a member of the Scrum Team. Moreover, the user can export the risks found into the Project Risk Backlog to the current project. Figure 7 shows an example of how the risk search is performed in RIMPRO-AST: the exemplified search took an empty string as input and produced as output a risk set containing all the risks found in other projects. To restrict the search, only type and submit the string that we want to look for.

Search risks

Perform the risk research from other projects

Enter the name of the risk:

Search

Risks found

Name	Classification	Probability	Impact	Action
Very large project team	Not defined	0.4	0.7	👁
Changes during the sprint	Migration to Scrum	0.4	0.7	👁

Fig. 7. Risk search interface. Elaborated by the authors.

5 RIMPRO-AST Evaluation

We conducted a survey involving 31 participants, including Information Technology professionals, Computer Science undergraduate, and graduate students, to evaluate usability aspects and the effectiveness of RIMPRO-AST in supporting users to perform Risk Management tasks in agile projects. Figure 8 shows the distribution of the participants in each area of activity. The participants contributed to detecting possible improvements to RIMPRO framework and RIMPRO-AST tool to better suit them to the needs of agile project teams, since they have also answered a qualitative questionnaire with more general questions.

Through face-to-face presentations, one of the authors introduced the RIMPRO risk management framework to the groups of participants (Information Technology professionals and students) to clarify the objectives of the evaluation and to solve possible doubts about RIMPRO and the capabilities of RIMPRO-AST tool. After that, RIMPRO-AST was introduced through a simulated Sprint, in which the participants simulated the identification and analysis of risks, as well as their allocation to Sprints. At the end of the presentations, the participants were invited to use RIMPRO-AST for 15 days to evaluate the tool remotely. Moreover, an evaluation form and guide containing information on how to use RIMPRO-AST module were sent to all the participants via e-mail.

● IT Professional ● Graduate Student ● Undergraduate Student

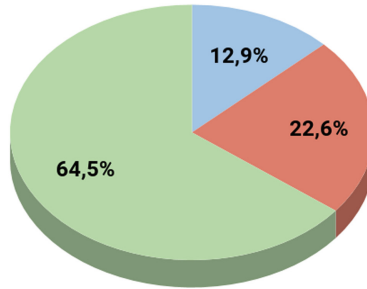


Fig. 8. Distribution of evaluation participants. Elaborated by the authors.

5.1 Evaluation Results

Throughout the evaluation period, participants filled the evaluation form anonymously. The questions were divided into two categories: general questions, and questions related to the specific capabilities of RIMPRO-AST. The general questions were assessed using the Likert Scale [6], while the questions related to specific functions of the tool were assessed using scores ranging from zero to ten points.

RIMPRO-AST General Evaluation. The Likert scale adopted in this work comprises the following items: *Strongly Disagree*, *Partially Agree*, *Neither Agree nor Disagree*, *Partially Disagree*, and *Strongly Agree*. The following statements were provided for the analysis of the participants:

- **Statement 1 (S1):** I am satisfied with the ease of use of RIMPRO-AST;
- **Statement 2 (S2):** I am satisfied with the quality of the RIMPRO-AST interface.

RIMPRO-AST Specific Functionalities Evaluation. The following RIMPRO-AST functionalities were evaluated by the participants:

- **Functionality 1 (F1):** Support for information on changes made to risks (Notes Board);
- **Functionality 2 (F2):** Support for the risk management plan;
- **Functionality 3 (F3):** Support for risk identification through the Project Risk Backlog;
- **Functionality 4 (F4):** Support for risk organization through Sprint Risk Backlogs;
- **Functionality 5 (F5):** Support for the visualization of risks through the Probability and Impact Matrix;
- **Functionality 6 (F6):** Support for the reuse of risks from other projects through the Risk Search.

Open Questions. We also provided two open questions to the participants expressing their opinions about RIMPRO-AST strengths and weaknesses:

- **Question 1:** What are the strengths of RIMPRO-AST?
- **Question 2:** What are the weaknesses of RIMPRO-AST?

5.2 Discussion

From the analysis of the results shown in Fig. 9, it is possible to conclude that the statement related to the ease of use (S1) was most satisfactory, i.e., 60% of the participants strongly agreed and 33% partially agreed. The statement related to the interface quality (S2) received the worst rating, and the participants’ justifications involve the lack of information on how to use the module’s functions, which made it difficult to understand. Analogous to the RIMPRO evaluation results, the participants also concluded that iteratively managing risks is beneficial since the Scrum Team has few members, and the project budget is relatively smaller than traditional projects. Thus, instead of monitoring all risks throughout the project, only risks that can affect Sprint are monitored. This is important when it comes to risks involving the Product Owner because (s)he is present throughout the project and, so, the probability of occurrence, and the impact of the risks involving him/her may vary over the course of the Sprint. Consequently, the risks monitored in a Sprint may not be monitored in subsequent Sprints, and vice versa. The participants considered it crucial to provide a prior list of risks involving the Product Owner’s roles to guide the execution of RIMPRO-AST throughout the project since it provides lessons learned from previous projects and assists the Scrum Team in discussing new risks that can emerge [18].

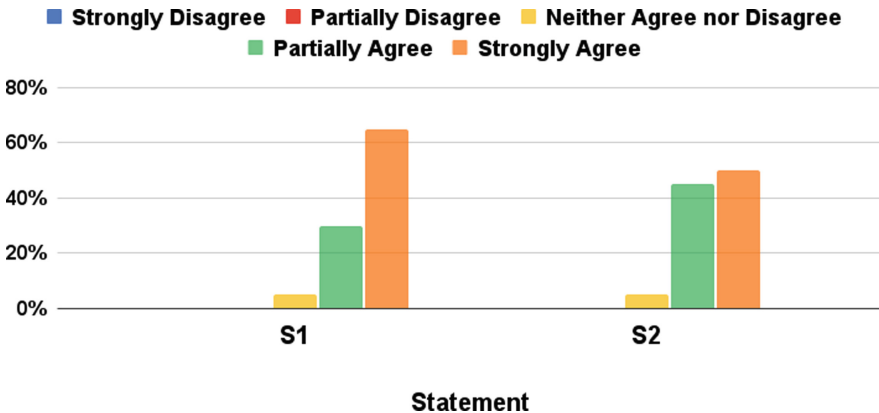


Fig. 9. General evaluation histogram of RIMPRO-AST. Elaborated by the authors.

Table 1 and Fig. 10 present the evaluation results of RIMPRO-AST specific functions. From Table 1, it is possible to verify, based on mode, that the most

frequently assigned score by the participants was ten points, and, according to the median, half of all scores given by the participants in all functionalities evaluated was greater than nine points. In a complementary way, from the analysis of Fig. 10 it is possible to conclude that all functions were well evaluated, and the support for the organization of risks through the Sprint Risk Backlogs (F4) received more negative reviews due to the participants' difficulties when inserting a risk in a given Sprint Risk Backlog. However, it is important to highlight that, even for functionality F4, there was an average score equal to 8.8446 with a standard deviation equal to 1.061, which characterizes a satisfactory evaluation.

Table 1. Statistical results related to specific functionalities of RIMPRO-AST. Elaborated by the authors.

Functionality	Average	Mode	Median	Min-Max	Std Dev
F1	9	9.5	9	5–10	1.1547
F2	9.129	10	9	5–10	1.1178
F3	9.087	10	9	5–10	1.1738
F4	8.8446	10	9	5–10	1.061
F5	9.0322	10	10	5–10	1.378
F6	9.0322	10	10	5–10	1.1968

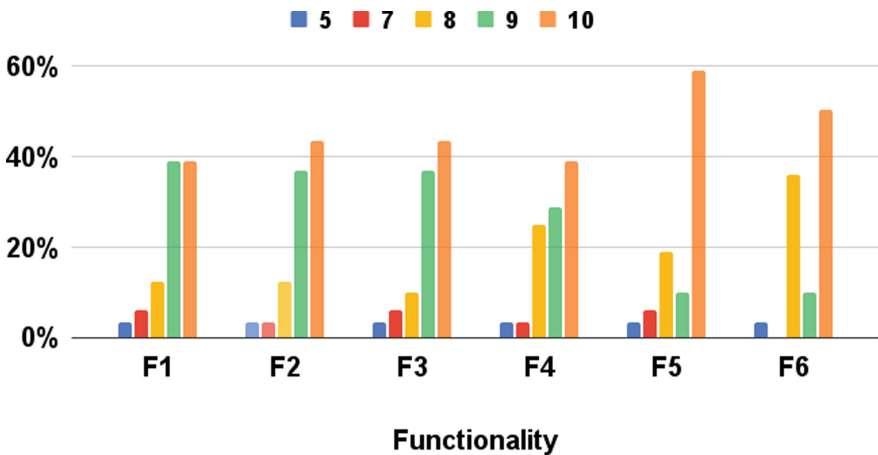


Fig. 10. Evaluation histogram of specific functionalities of RIMPRO-AST. Elaborated by the authors.

5.3 Threats to Validity

We identified the following threats to the validity of this study:

- **Sample Quality:** The sample may not be representative of the population since the most of the participants are not IT professionals, and we were unable to obtain the availability of any Product Owner to participate in the evaluation. The participants' selection was restricted to IT people who have contact with projects that use PMBoK and Scrum to reduce threats to validity. In addition, face-to-face presentations were held with the participants, providing additional background on RIMPRO and RIMPRO-AST.
- **Application:** Throughout the assessment process, we have not had the opportunity to apply RIMPRO in a realistic situation, where IT professionals would use RIMPRO to manage the risks that can arise throughout a concrete project. To have a more realistic scenario, we provided a tutorial section to explain the RIMPRO-AST capabilities to the participants through a simulated Sprint.

6 Related Works

After an analysis of the main tools that provide automated support for the management of Scrum projects available in the literature, we could not find, until the writing of this paper, a tool that could be used specifically in the context of Scrum projects, involving the Product Owner and risk management. The criterion used for searching such tools was the current tools adopted by the market, where we identified the following tools:

- **Scrumwise:** it is a web-based tool (available since 2009) for managing Scrum projects that allows the management of Scrum events, Scrum Team, and artifacts produced throughout the project [25];
- **Jira Software:** Created by Atlassian in 2004, a company that develops systems for project management. Jira is a cross-platform agile project management tool that offers functions similar to those offered by ScrumWise, but supports any agile approach [3];
- **Axosoftware:** AxoSoft was created by Hamid Shojaee, the creator of the world's most famous Scrum video, with a restricted focus on Scrum. Like the previous tools, it also works via a web browser and allows the user to manage all aspects involving the Scrum project [4];
- **ScrumHalf:** Created in 2011, ScrumHalf is a Brazilian web tool for managing Scrum projects. The tool makes it possible for the main actions of the project to be published on the Twitter social network so that only Scrum Team members can see them [24].

To carry out the analysis of such tools, we defined evaluation criteria aiming to analyze the treatment that each tool presents for Scrum, the Product Owner, and risk management. The analysis criteria used were:

- **AC1:** Coverage of all Scrum steps;
- **AC2:** In-depth coverage of all project steps performed by the Product Owner such as requirements management, quality management, and communication with project stakeholders;
- **AC3:** Exclusive focus on Scrum;
- **AC4:** Project risk management support.

From this, the selected tools were analyzed based on the established criteria, and Table 2 presents the results obtained from such analysis. Concerning criteria AC1, AC2, and AC4, all the analyzed tools offer the same functions, differing in some specific functions to manage the project steps performed by the Product Owner, such as managing the Product Backlog, Sprint Backlog, and Increments. However, it is noteworthy that all of them are commercial tools, so the analysis of the tools was performed with versions that have a limited period of use.

Table 2. Analysis of the main tools to support Scrum project management. Elaborated by the authors.

Tool	Analysis Criteria			
	AC1	AC2	AC3	AC4
Scrumwise	✓	✓	✓	✗
Jira Software	✓	✓	✗	✗
Axosoft	✓	✓	✓	✗
ScrumHalf	✓	✓	✓	✗

Concerning the exclusive focus on Scrum, analyzed using the AC3 criterion, only Jira Software does not have a restricted focus on the method. Such a holistic approach may not be beneficial, as future updates may direct the tool to address other agile methods to phase out or discontinue the tool's functions that address Scrum.

Although all the tools support the steps proposed by Scrum, according to the AC1 criterion, the main limitation observed is that none of them support the project risk management, according to the AC4 criterion. Even though it is relevant to use tools to support the risk management activity in Scrum projects, the main agile project management tools do not cover such activity [15]. This finding highlights the importance of this work, which presents an automated framework for risk management in projects developed in conformance with Scrum.

7 Conclusion

Although Scrum is the most current used agile project management method, it does not provide a systematic way to manage risks that may arise throughout

the entire project. In this context, this work contributed to automating the processes defined in RIMPRO risk management framework via RIMPRO-AST tool, considering that the existing agile project management tools do not support risk management. Such automation has been integrated within SAPM Scrum module to facilitate the implementation of RIMPRO in the Scrum Team, and to provide flexibility in decision-making processes by project teams.

For future work, we propose to extract knowledge about the documented risks, for example using Text Mining techniques. Therefore, users would have access to additional information to guide their decisions. Moreover, we propose to improve the RIMPRO-AST graphical user interface to address human-computer interaction usability and accessibility principles, so that users with cognitive, perceptual, and movement limitations can use the tool with efficiency. Finally, we also intend to update/replace the frameworks used in the Scrum module to enable compatibility with current Web browsers, such as Mozilla Firefox, as well as mobile devices, such as smartphones and tablets, which constrain the use of RIMPRO-AST.

References

1. Alliance, A.: Agile Practice Guide, vol. 1. Bukupedia (2017)
2. Andrat, H., Jaswal, S.: An alternative approach for risk assessment in scrum. In: 2015 International Conference on Computing and Network Communications (CoCoNet), pp. 535–539. IEEE (2015)
3. Atlassian: Jira software - the #1 software development tool used by agile teams (2021). <https://www.atlassian.com/software/jira/>
4. Axosoft: Agile project management software (2021). <https://www.axosoft.com/>
5. Beck, K., et al.: Manifesto for agile software development (2001). <http://www.agilemanifesto.org/>
6. Boone, H.N., Boone, D.A.: Analyzing Likert data. *J. Extension* **50**(2), 1–5 (2012)
7. Campbell, J., Kurkovsky, S., Liew, C.W., Taflivovich, A.: Scrum and agile methods in software engineering courses. In: Proceedings of the 47th ACM Technical Symposium on Computing Science Education, p. 319–320. SIGCSE 2016, Association for Computing Machinery, New York, NY (2016). <https://doi.org/10.1145/2839509.2844664>
8. Dalkey, N., Helmer, O.: An experimental application of the Delphi method to the use of experts. *Manage. Sci.* **9**(3), 458–467 (1963)
9. Dingsøyr, T., Nerur, S., Baliijepally, V., Moe, N.: A decade of agile methodologies: towards explaining agile software development. *J. Syst. Softw.* **85**(6), 1213–1221 (2012)
10. Faceli, K., Lorena, A.C., Gama, J., Carvalho, A.C.P.d.L.F.d.: Inteligência artificial: uma abordagem de aprendizado de máquina. LTC (2011)
11. Fitzgerald, B., Stol, K.J.: Continuous software engineering: a roadmap and agenda. *J. Syst. Softw.* **123**, 176–189 (2017)
12. Fontoura, L.M., Price, R.T.: Systematic approach to risk management in software projects through process tailoring. In: SEKE, pp. 179–184 (2008)
13. Friess, E.: Scrum language use in a software engineering firm: an exploratory study. *IEEE Trans. Prof. Commun.* **62**(2), 130–147 (2019). <https://doi.org/10.1109/TPC.2019.2911461>

14. de Godoi Contessoto, A., et al.: Improving risk identification process in project management. In: Proceedings of the International Conference on Software Engineering and Knowledge Engineering, SEKE, pp. 555–558 (2016)
15. Gold, B., Vassell, C.: Using risk management to balance agile methods: a study of the scrum process. In: 2015 2nd International Conference on Knowledge-Based Engineering and Innovation (KBEI), pp. 49–54. IEEE (2015)
16. Gregoriades, A., Lesta, V.P., Petrides, P.: Project risk management using event calculus (s). In: SEKE, pp. 335–338 (2011)
17. Lampa, I.L., de Godoi Contessoto, A., Amorim, A.R., Zafalon, G.F.D., Valêncio, C.R., de Souza, R.C.G.: Project scope management: a strategy oriented to the requirements engineering. In: International Conference on Enterprise Information Systems, vol. 2, pp. 370–378. SCITEPRESS (2017)
18. Lopes, S.d.S., de Souza, R.C.G., Contessoto, A.d.G., de Oliveira, A.L., Braga, R.T.V.: A risk management framework for scrum projects. In: 23rd International Conference on Enterprise Information Systems (ICEIS 2021) (2021)
19. Mendonça, L.T., Esteca, A.M.N., De Souza, R.C.G., Santos, A.B., Valêncio, C.R.: A scrum support system integrated to a web project management environment. In: 29th International Conference on Computers and Their Applications, pp. 111–116 (2014)
20. Northrop, L., et al.: A framework for software product line practice, version 5.0. SEI (2007). <http://www.sei.cmu.edu/productlines/index.html>
21. PMI (ed.): A Guide to the Project Management Body of Knowledge (PMBOK Guide). Project Management Institute, 6 edn. (2017)
22. Pritchard, C.L.: Risk Management: Concepts and Guidance. CRC Press (2014)
23. Rola, P., Kuchta, D., Kopczyk, D.: Conceptual model of working space for agile (scrum) project team. *J. Syst. Softw.* **118**, 49–63 (2016)
24. ScrumHalf: Scrumhalf - agilizamos o seu negócio (2021). <https://myscrumhalf.com/>
25. Scrumwise: Scrumwise - features (2021). <https://www.scrumwise.com/features.html>
26. Siqueira, D.L., Fontoura, L.M., Bordini, R.H., de Lima Silva, L.Á.: A knowledge engineering process for the development of argumentation schemes for risk management in software projects. In: Proceedings of the 29th International Conference on Software Engineering and Knowledge Engineering (2017)
27. Sutherland, J., Schwaber, K.: The scrum guide (2020). <https://www.scrum.org/>
28. Sutherland, J., Sutherland, J.: Scrum: the art of doing twice the work in half the time. Currency (2014)
29. Tavares, B.G., da Silva, C.E.S., de Souza, A.D.: Practices to improve risk management in agile projects. *Int. J. Software Eng. Knowl. Eng.* **29**(03), 381–399 (2019)
30. Tavares, B.G., da Silva, C.E.S., de Souza, A.D.: Risk management analysis in scrum software projects. *Int. Trans. Oper. Res.* **26**(5), 1884–1905 (2019)