SERGE – Serious Game for the Education of Risk Management in Software Project Management

Giusy Annunziata
gannunziata@unisa.it
University of Salerno
Salerno, Italy

Stefano Lambiase
slambiase@unisa.it
University of Salerno
Salerno, Italy

Fabio Palomba
fpalomba@unisa.it
University of Salerno
Salerno, Italy

Filomena Ferrucci
fferrucci@unisa.it
University of Salerno
Salerno, Italy

ABSTRACT

Software Project Management is the systematic and disciplined approach for planning, executing, monitoring, controlling, and closing software development projects. Risk management is a key knowledge area that aims to react to the unpredictable events that often affect software projects. Teaching risk management is vital to equip individuals and organizations with the adapted skills to prevent and monitor challenges and potential issues. In this paper, we propose a serious game named **Serge**, conceived to involve students in learning risk management and improve their skills through gamification and simulation of a real-world application context. The features for the design of Serge were identified through a literature review. An iterative Game Design Phase was employed to build, test, and refine the design of Serge. Finally, the proposed approach was assessed by conducting a controlled experiment to compare risk management skills acquired through a traditional lecture and using Serge. The results show that adopting a serious game as Serge, able to involve the students actively, can improve the acquisition of risk management skills.

CCS CONCEPTS

• **Software and its engineering → Risk management**; *Collaboration in software development*; • **Applied computing → Education**.

KEYWORDS

Risk Management; Software Project Management; Software Engineering Education; Serious Games.

ACM Reference Format:
Giusy Annunziata, Stefano Lambiase, Fabio Palomba, and Filomena Ferrucci. 2023. SERGE – Serious Game for the Education of Risk Management in Software Project Management. In *Proceedings of 46th International Conference on Software Engineering (ICSE 2024).* ACM, New York, NY, USA, 10 pages. https://doi.org/XXXXXXX.XXXXXXX

ICSE 2024, April 2024, Lisbon, Portugal
© 2023 Association for Computing Machinery.
ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Software Project Management is a set of knowledge areas outlined in the *Project Management Body of Knowledge (**PMBOK**)* applied to ensure that a software project meets the constraints of time, cost, and quality, and customer requirements [15]. The misapplication of this knowledge may lead to project failure and, for this reason, the importance of managerial figures cannot be ignored [5].

A comprehensive understanding of project management processes, tools, and best practices enhances the ability to deliver software projects successfully and contributes to improved resource distribution, cost control, and quality assurance. However, despite managers' expertise, threats and uncertainties can derail project timelines, budgets, and outcomes. Risk management is the area that focuses on identifying, assessing, and mitigating potential risks—events that could positively or negatively impact the project—that may emerge [4]. Given the intrinsic complexity and unpredictability of software development and management, the education and training of individuals to acquire suitable risk management skills is very important. Moreover, adequate approaches should be adopted to successfully prepare project managers to manage real-world challenges. In particular, the teaching of Software Project Management (SPM) knowledge areas has been supported by organizations such as the Association for Computing Machinery (ACM) and IEEE-Computer Society in their joint task force curricula for undergraduate computing courses [25].

Nowadays, teaching techniques based on the use of *Serious Games* have become popular in education due to their ability to engage learners, promote active participation, and enhance the application of knowledge [18]. It is a proven and effective learning method incorporating gamification and simulation [6, 20, 21]. The term gamification refers to the intention to use design elements of games—such as strategic and mechanical design—in activities that do not belong to the world of games [18]. Simulation refers to reproducing real-world contexts to exploit them for learning [18].

Marcelino and Domingues [21] investigated the various PMBOK areas already exploiting serious games as a teaching support method. The key results highlight that most serious games proposed in the literature target time management. At the same time, the authors noticed a noticeable lack of instruments that might be employed to teach other software project management areas. Among them, risk management represents one of the least considered.

The study's results indicate a significant interest in adopting serious games as educational approach for teaching knowledge areas in the PMBOK. However, it is evident that, despite their potential, serious games still need to cover certain aspects of software project development. Among them, one of the most impactful is Risk Management. This observation emphasizes the need for continued innovation in serious game to ensure comprehensive coverage of the challenges presented in software project management.

To fill the gap highlighted in state of the art, we propose **SERGE** (**S**erious Game for the **E**ducation of **R**isk Mana**ge**ment); It is dedicated to software project management students, aiming to educate future generations of project managers by stimulating risk management learning activities through a board game.

SERGE was developed following a three-step process, described in Section 3,s Figure 1. In Step 1 (Section 4), we conducted an in-depth requirement elicitation phase, drawing insights from related work through a literature review. In Step 2 (Section 5), the game's design was drafted and rigorously tested using an iterative improvement approach, using different pilot tests conducted with Ph.D. students. Furthermore, in Step 3 (Section 6), we conducted a controlled experiment involving students enrolled in a software project management course to validate its educational effectiveness. Finally, Section 7 explains the discussion point that emerged, the threats to the validity, and how we mitigated them.

2 BACKGROUND AND RELATED WORK

This section describes the background and related work that is the foundation for our contributions.

2.1 Risk Management

Software project management is the set of knowledge areas for the Project Manager to manage and monitor the development progress of a software project [15]. Inadequate software project management application leads to project failure in 75% of cases [5].

One of the most important knowledge areas for the success of a software project is Risk Management—a set of practices to apply for manage events that can positively or negatively impact the project [4]; It's based on different steps, which aim to identify, analyze, evaluate, and monitor risks [22, 24]. Moreover, one of the most important but least practiced steps is risk retrospective, in which the project manager reports all identified risks and strategies to pass knowledge to future project managers [13].

Risk management is a proactive approach that enhances strategic planning and contributes to achieving long-term objectives while mitigating potential issues [14]. The project managers can minimize potential negative impacts on their projects by implementing robust risk management practices. Furthermore, risk management enables informed decision-making by providing a comprehensive understanding of potential risks and their consequences, allowing for proactive strategies to be put in place [22].

Given the need to keep track of all risks and how they are dealt with, risk management requires a lot of documentation [3]; However, the world of software development is moving toward adopting agile development models, which, for their definition, are against exhaustive documentation [12].

A survey shows that in those models the common avoidance and mitigation strategies adopted is involve customer communication and participation [12]. However, Agile's iterative processes and various continuous improvement practices contribute to risk management, although they are not considered part of it.

Risk management could be considered one of the most sensitive areas in Software Project Management because a single mistake during its application could lead to a project failure; the importance of applying it cannot be understated [13]. For this reason, the education behind this area is complex; it needs simulation and practice, not only theory concepts to be handed down.

2.2 Gamification in Software Engineer and Software Project Management

Nowadays, gamification in the educational context is becoming increasingly popular. Various types of games are developed to support learning; they are utilized in diverse contexts, including Software Engineering and Software Project Management.

Petri et al. [25] investigated the benefits of using a digital game versus a non-digital one. They compared different games, conducting controlled experiments with students to reach their effectiveness. The results show that the learning degree is the same; however, non-digital games provide greater involvement and participation, while digital games are more automatic and easy to understand.

Gresse-Von-Wangenheim et al. provide **SCRUMIA** [11], a game designed for learning the Scrum framework and is based on managing and implementing a project through tasks. A fantasy scenario provides guidelines for each player to know what to do to complete the product implementation.

Another example of a serious game in the Software Engineering context is provided by Liu et al. BARA is a serious game that aims to improve skills in requirement elicitation [19]. Players can interact with game characters, which provide them pieces of information to elicitate the requirements for their project.

Among the serious games closest to Software Project Management, in particular Risk Management, there is the work of Taran [29] provides a game that uses the traditional board game element; it is composed of 5 phases, each for a specific stage of the waterfall software development model. The randomness provided by the dice roll simulates the unpredictability of risks. The complexity of game rules limits the game to possible expansions.

Another relevant game is *Riskware—A game for teaching Software Project Risk Management* [16]: The game is similar to the first; it consists of a dice roll to simulate risks and buy resources to advance. However, it differs from traditional board games because each player can make their board according to the project steps to satisfy. However, this feature is also the cause of the main problems and complexities of the game.

Another game analyzed is *ARMI 2.0—An Online Risk Management Simualtion* [28]: It is a digital game consisting of 5 rounds, one for each stage of the waterfall development model. At the beginning of each round, it is conducted a brainstorming phase about the possible risks; according to them, each player bought specific resources to prevent and mitigate risks.

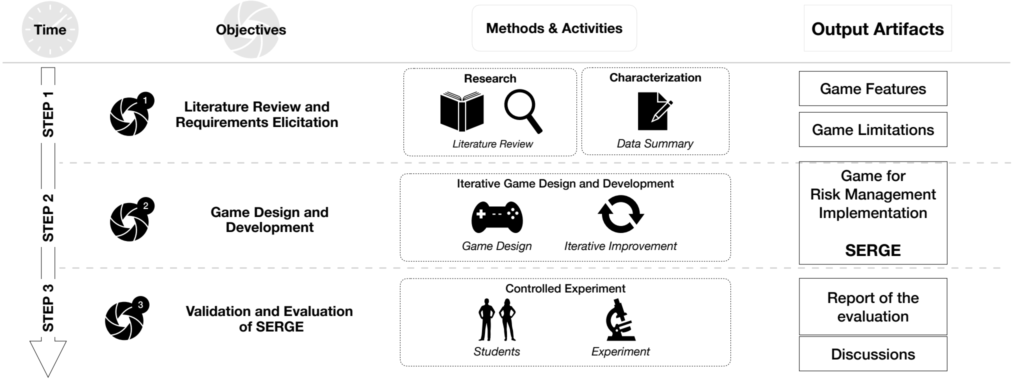


Figure 1: Overview of the methods used for our study.

3 RESEARCH STUDY DESIGN

We describe the approaches to achieve the objective of the study.

> ◎ **Main Objective**
>
> *Propose and evaluate a new approach to support the teaching of Risk Management for Software Project Management through a serious game. The learning objectives we aim to achieve are to improve skills related to (i) risk analysis, (ii) risk prevention, and (iii) risk contingency strategies.*

To achieve our objective, we divided it into three sub-objectives, summarized in Figure 1 and described in the following sub-objectives:

RO$_1$—Literature Review and Requirements Elicitation
Requirements elicitation through educative methods in the literature.

Starting from previous work on education, we conducted a requirements elicitation phase through a literature survey to obtain a list of features that SERGE had to fulfill.

RO$_2$—Game Design and Development
*Game Design and Development*Iterative phase between creation and validation of Serious Game's design and content.

We derived the design of SERGE by extracting features from the literature. Subsequently, we conducted an iterative process involving testing, refinement, and improvement to optimize the game's structure and effectiveness.

RO$_3$—Validation and Evaluation of SERGE
*Game Design and Development*Evaluate the efficacy of SERGE through controlled experiments with university students.

We conducted a controlled experiment to validate and assess SERGE. We administered a risk management skills questionnaire to a group of 10 university students both before and after using SERGE. After, we compared the responses of a control group of 10 university students with those of students who had utilized SERGE in their training. We also assess the results using the statistic test T-test.

The final result of those 3 research objectives is **SERGE**, a serious game to support the teaching of Risk Management designed with extracted features to fill the gap of already validated serious games. SERGE is validated through a controlled experiment with University students of a software project management course.

4 LITERATURE REVIEW AND REQUIREMENTS ELICITATION

To satisfy the RO$_1$, we identify information on current alternative teaching methods applicable in software project management. It aims to provide our Serious Game with collective knowledge and successful practices of existing games, incorporate proven features, and optimize its potential for effectiveness.

We conducted a requirements elicitation phase involving a literature survey of the existing serious games in software project management, so our keywords are "*Serious Game*" and "*Software Project Management*". Starting with the most relevant results, shown in Section 2, we conducted a *snowballing phase* to expand our research and identify serious games for analysis, understanding their structure and game characteristics. Below are the serious games analyzed and the features extracted from them. Table 1 summarizes the results and the extracted features.

4.1 Game Definition

One of our starting points is "*An analysis of how well serious games cover the PMBOK*" [21], in which Marcelino and Domingues surveyed the various serious games created in the context of software project management, to understand which of the knowledge areas were supported by the use of serious games;

Another relevant work identified is "*Gamification in Education*" [18]. They define the main concepts of **Gamification**—the use of elements of the world of games in a real-world context—and **Simulation**—the reproduction of a real scenario in a protected environment [18]. In particular, Gamification provides elements of fun, and interactivity in learning contexts, making it engaging and motivating for students. Simulation provides a safe and controlled environment for students to immerse themselves in real-world scenarios and apply theoretical knowledge to practical situations.

The combination of gamification and simulation gives rise to *Serious Games*, a powerful educational tool that uses the best of both worlds. It allows students to apply their knowledge in a protective environment, making instruction more effective and engaging. The definition of Serious Games meets our needs to realize an educational method to improve risk management skills, in which participants can act in the role of Project Managers and test their skills, understand their mistakes, and improve their knowledge.

4.2 Features from Serious Games in Software Project Management

Starting from the reviewed papers, described in section 2, we identified features from existing serious games and considered them possible features for SERGE. Moreover, we focused on the weaknesses of those works to propose a Serious Game that would fill the identified gaps. This ensured that our serious game was based on established knowledge and supported us in creating an effective and engaging educational tool for teaching risk management.

It is also important to note that the games examined are based on the traditional waterfall software development model. However, the software development world is currently moving into using the Agile software development model [12]. To meet the needs of companies, SERGE will be based on the Agile development model. The analysis identified features from existing serious games and experiments already conducted on them. Moreover, those games' limitations represent an important starting point for our work. We report all the information about our analysis in Table 1.

> **Summary of Requirements Elicitation**
>
> The requirement elicitation phase led us to define features and objectives that SERGE need to satisfy; SERGE will:
> - combines gamification and simulation to provide an engaging environment where practice the skills learned.
> - support risk management with an Agile development model in a software development context.
> - be non-digital to encourage knowledge sharing.
> - provides a scenario to simulate the development process.
> - have simple rules and game mechanics.
> - provides suitable game materials to support risk management learning.

5 GAME DESIGN: SERGE – SERIOUS GAME FOR THE EDUCATION OF RISK MANAGEMENT

This section satisfies our RO$_2$; in order to make clear all the process, first we describe the game, and after we explain the creation and validation process of SERGE's design through an Iterative Improvement approach [26].

5.1 Description of the Game

Game Scenario and Material. The scenario behind SERGE involves players utilizing the agile development model to create a platform for the Office of Services at University of Salerno. We selected a topic familiar to the students to ensure their primary focus remains on risk management concepts rather than on software development. SERGE is provided by decks of different cards, shown and described in the table 2.

Further materials are provided to participants, which include:

- **User Game** Sheet in which the players can write the game progress, such as resources acquired and risks;
- **Resources** A list of resources that can be used to mitigate risks; including technologies and team skills; on User Game Sheet.
- **Notes** A sheet to use to draft strategies;
- **Token** Used as coins in the game; they are divided in: High (value of €250), Medium (value of €200), and Low (value of €150).

All materials are illustrated in detail and available in the online appendix [1]. The information behind the scenario and the material needed for learning, i.e., risk and event cards, was based on books to teaching software project management [10, 15, 27].

Game Phases. The game's prerequisites include theoretical knowledge of aspects of (1) Risk Management, (2) Scrum Framework, and (3) Planning Poker Approach; however, the latter is explained in detail in the game leaflet in the online appendix materials [1].

Collaboration and exchange of ideas are one of the game's primary goals, so to make the game effective, a minimum of 4 players is recommended. On the other hand, a maximum limit of 12 participants has been set to not prolong the conversations and not to create chaos during the game. The game can be played either individually or in pairs.

The game is divided into 3 different sequential phases, each with a different mode. Each stage is designed to meet one of the learning objectives discussed in Section 5.2.

Each player will play the role of a Scrum master who must take in information and develop a strategy for risk prevention and mitigation. Players will choose a volunteer to play the lead player, who, in addition to taking part in the game, will also be in charge of distributing the materials at each turn and ensuring that everyone understands the game and what to do at the various phases. Before starting to play the game, it is recommended that such a player read the leaflet, make sure that everyone understands the game, and distribute to each of them (1) the decks of probability and impact cards, (2) a User Profile Card, (3) Tokens, (4) User Game Sheet, and (5) a notes sheet. Cards and tokens are reported in Table 2.

1. Enhancing analysis skills through communication and confrontation - Phase One of the game: ***Risk Analysis****.* In this phase, through a Planning Poker approach, players will give their opinion on the impact and probability of risks, giving reasons for the choice. Each player will receive a deck of probability cards and a deck of impact cards. For each risk present in the game, the lead player will read the id and description of the risk; according to it, each player will hypothesize its impact and probability on the project and choose it with the impact and probability cards. An example of cards is shown in Figure 2. All players simultaneously unveil the value of the cards. The value with a major presence will be assigned to the risk. The lead player will write it on the card and place it on the board in the box related to the probability chosen, as illustrated in Figure 4. This phase is focused on active collaboration, constant discussion, and knowledge sharing, providing each player with an additional point of view.

Table 1: Results of Requirement Elicitation

Paper	Problems or Idea	Features extracted for SERGE
Scrumia—An educational game for teaching SCRUM in computing course [11]	Use of a fantasy scenario to describe the user story and the tasks to do to complete the project	• Using a scenario to provide context and background to participants.
BARA: A Dynamic State-based Serious Game for Teaching Requirements Elicitation [19]	Participants interact with practitioners to conduct the requirement elicitation phase	• Simulate and act a role. • Using a scenario to provide project information to participants.
Using Games in Software Engineering Education to Teach Risk Management [29]	Excessive complexity of the game and rules	• Simple rules and game mechanics. • Support materials to remind participants of the rules of the game.
Riskware—A game for teaching Software Project Risk Management [16]	Game strategies are publicly displayed, and players tend to copy each other. Too much freedom provides confusion. The game provides insufficient money to buy resources	• Each player develops their strategy privately. • The phases of the game are distinct and well-defined. • Providing fake currency for procurement.
ARMI 2.0—An Online Risk Management Simulation [28]	Appreciated the use of brainstorming	• Use of Planning Poker to compare ideas. • Use of Scrum Sprints to create distinct phases.
Games for Teaching Software Project Management: An Analysis of the Benefits of Digital and Non-Digital Games [25]	Digital games have better player guidance and support, while non-digital games are more engaging than digital	• Making a non-digital game to increase cooperativeness and participation. • Provide supporting materials to succumb to the gap of non-digital games.

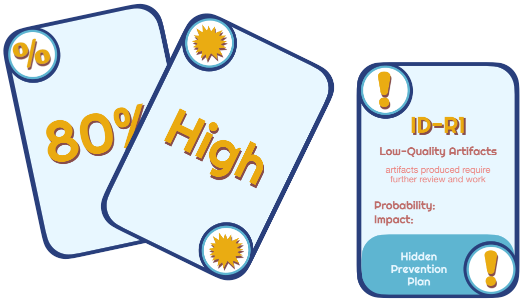
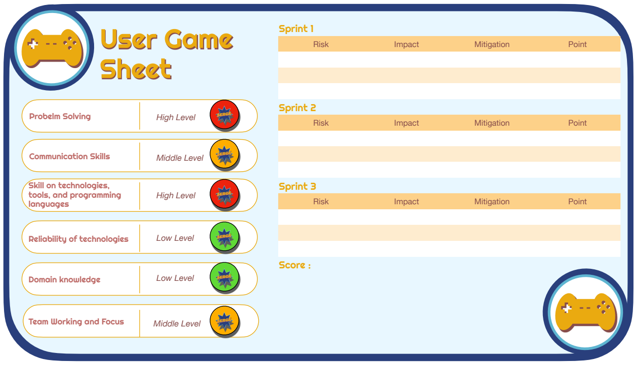


Figure 2: Planning Poker Card and Risk Card used in the First Phase



User Game Sheet

Probelm Solving	High Level
Communication Skills	Middle Level
Skill on technologies, tools, and programming languages	High Level
Reliability of technologies	Low Level
Domain knowledge	Low Level
Team Working and Focus	Middle Level

Sprint 1

Risk	Impact	Mitigation	Point

Sprint 2

Risk	Impact	Mitigation	Point

Sprint 3

Risk	Impact	Mitigation	Point

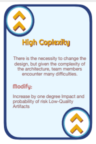
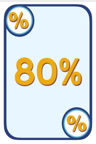
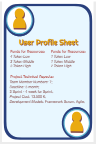
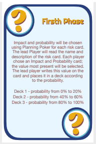
Score :

Figure 3: User Game Sheet

2. Developing organizational skills to implement a strategy to deal with risks - Phase Two of the game: ***Procurement****.* In the second phase, each player must implement a strategy to prevent risk and buy the most appropriate resources. All resources are reported in the online appendix [1]; an example is reported in Figure 3. At the beginning of the game, the lead player provided to each player tokens, a User Game sheet on which they signed the list of resources available, and a User Profile card with the number and distribution of tokens that each player can spend to buy resources. Each resource must be acquired with a token that could be high, medium, or low, as shown in Table 2. E.g. to prevent risks such as *low-quality artifacts*, a player decides to buy a high level of the resource "*Skill on technologies, tools, and programming languages*", spending a high token (red). In this way, each player will choose the appropriate value to assign to each resource according to the effectiveness of that resource to prevent or mitigate the risks analyzed in the first phase. Figure 3 shows the User Game Sheet; on the left of the sheet, there is the resource table with the token spent to buy them, and on the right, the table with the different game rounds, i.e., Sprint.

3. Improving skills by addressing risks in a simulation - Phase Three of the game: ***Risk Management****.* In this phase, players will simulate the management of a software project. The randomness provided by the dice roll simulates the events that trigger the occurrence of risks. This phase is divided into 3 rounds, each representing a *Sprint*. In each round, starting from the younger player, each player rolls dice; according to the outcome, they can catch a risk or an event card (even value for the event card and odd value for the risk card). If the result is an event card, it describes an event scenario that modifies the probability and impact of one or more risk cards, in worse or better. The lead player should update the information about probability and impact on the cards. If the result is a risk card, the player rolls the dice again; the second result indicates the probability deck from which the risk card is caught. E.g., if the result is 3, the player will have to draw a risk card that, on the board, is located in the risk card box with a probability 40% or 60%. The board is shown in Figure 4. Catching a risk card means that players find that risk in their project to prevent or mitigate it. To prevent the risks, the player should have all the resources in the card; otherwise, they can mitigate it by spending a token. For example, if a player encounters the risk of "*low-quality artifacts*", the prevention plan indicates having "*Skill on technologies, tools, and programming languages*" of medium level. If the player has acquired that resource at a medium or high level, they prevent

Table 2: Cards of SERGE

Card	Name	Description
	Card	the back of the cards are depicted as shown in the figure
ID-R1 Low-Quality Artifacts Hidden Prevention Plan	**Risk Card**	Risk cards have (1) id, (2) name, (3) description, (4) prevention plan (list of resources to have to prevent that risk), and (5) contingency plan (impact of the risk)
	**Event Card**	This card explains an event which makes re-evaluate the impact and probability of one or more risk cards
	Impact Card	Used in Planning Poker to indicate the impact (High, Medium, or Low) of a risk to occur
	**Probability Card**	Used in Planning Poker to indicate in percentages (from 0% to 100% on a scale of 20%) the probability of a risk to occur
	**User Profile Sheet**	User card with information recap
	Tokens	Coins of the game, divided in High (red), Middle (yellow), and Low (green)
	**Info Card**	Set of guide cards, which are provided to each player to remember quickly the rules and stages of the game

the risk, gaining 3 points. Otherwise, players read the impact of the card; if the impact is medium, the player will have to spend a medium token to mitigate that risk, gaining 1 point. The risks encountered, tokens spent, and points obtained can be signed in the User Game sheet, Figure 3. At the end of all sprints, the player with the most points will be the winner. This phase highlights the game's competitive sides, showing whether the strategic choices made are effective in encountering less risk than opponents.

4. Acquiring skills that are also applicable in the working world. The game aims to ensure that the concepts learned are also applicable in a work setting and not just in academia; for this reason, the structure of the serious game is based on the Agile development model, which is the most used nowadays in the working world. In particular, SERGE adopted the Scrum Sprints to divide the game's development process into different rounds. Moreover, the phase of knowledge sharing about the risks during the first phase of the game exploits the Planning Poker approach; It is commonly used to measure and estimate the User Stories [23, 30].

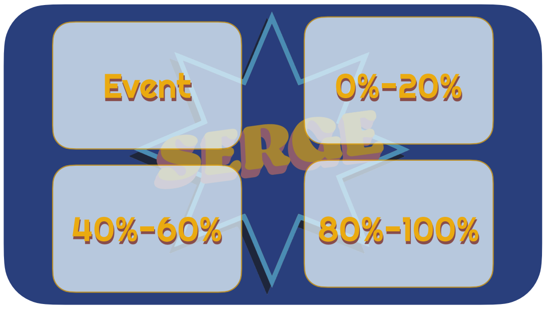


Figure 4: Game board with boxes for the 4 decks

SERGE

SERGE consists of 3 phases:

(1) **Risk Analysis**—Players decide risks' impact and probability using Planning Poker.
(2) **Procurement**—Players acquire the resources to prevent risks.
(3) **Risk Management**— The risks are simulated by dice roll. Each player receives points according to whether he prevents, mitigates, or succumbs to the risk. At the end of all sprints (game phases), the player with the most points will be the winner.

5.2 Design Creation and Choices

This section defines the framework of SERGE and discusses the design choices made to meet our goals.

Creation Steps. Kiryakova et al. [18] in their study, define the basic steps to apply gamification. Starting from them, we defined SERGE's framework:

(1) **Understand the end user who needs to acquire the knowledge**—We identified computer science students as end users of our serious game. Specifically, students dealing with software project management topics.
(2) **Define the learning objectives that the game must meet**—We identified as learning objectives (1) enhancing analysis skills through knowledge sharing, (2) developing organizational skills to plan a strategy to manage risks, (3) improving skills by managing risks in a simulation, and (4) acquiring skills that are also applicable in the working world.
(3) **Creating the educational content and activities**—We created a scenario that simulates the implementation of a software project. Moreover, we provide the risks a project may encounter and the events that may change its impact and probability.

(4) **Adding game-related elements**—We adopt game elements such as (1) the use of tokens as money, (2) the possibility of player enrichment by acquiring resources, (3) the use of points to determine victory, and (4) the use of playing cards and dice.

Communication VS Individuality. One of our main design choices was to consider the trade-off between communication and individuality. Communication and collaboration are necessary to facilitate the exchange of ideas to increase risk knowledge. However, excessive collaboration can lead to losing the main objective of the serious game, i.e., learning. Therefore, we decided to balance them both, creating an initial phase of collaboration where the various players can exchange their knowledge; in a second phase, the participants individually need to plan how to manage risks, respecting cost and resource constraints.

Use of Agile. Most of the serious games investigated and developed to teach software project management using traditional development approaches, such as the Waterfall Development Model. We aimed to meet business demand in the context of software development, which is increasingly moving toward using agile approaches [12]. To satisfy this goal, we employ an agile development strategy inside the framework of our serious game. Moreover, the simulation of Sprint, used in the framework Scrum, helps us maintain a clear status during the game [11].

Game complexity. We designed a game with simple rules and mechanics. This development choice aims to fill the gap of the serious games in literature, which are considered too complex to play. Moreover, we decided to develop the game in a non-digital format to promote player interaction and communication. However, according to Petri et al., one of the most significant advantages of digital games is being guided through the game [25]. To apply these advantages in a non-digital game, SERGE will provide adequate support material with all the game rules.

5.3 Design Testing

We conducted pilot tests to evaluate the correctness of the arguments and game mechanics of SERGE. Ph.D. students in software engineering were involved as pilot test participants, as during their master's degree, they took part in the Software Project Management course and are conducting research studies spanning that topic. The main focus was to assess game mechanics and the correctness and completeness of the topics covered.

First Pilot Test. The first pilot test was conducted at the *SeSa Lab*-Laboratory for Software Engineering of the Department of Computer Science, University of Salerno. It involved 4 Ph.D. students, all male. This pilot test aimed to verify the training objectives and improve the problems presented in the structure of the serious game, so it used a draft version of SERGE, Version 0.1. During the test, the game's creators interacted with participants to explain the game and clarify roles and any doubts about the materials. Among the problems highlighted, we find (i) an inappropriate amount of money, so money were replaced by token (ii) the planning poker cards were too generic for the game, so we realized planning poker card specific for the context and (iii) absence of dynamicity, for which we included events able to change the probability and impact of risks during the game.

Second Pilot Test. The second pilot test was conducted at the SeSa Lab, but differently from the first one, the structure of SERGE was more complete, thus facilitating the game's execution. 6 Ph.D. students—4 males and 2 females—were chosen as participants and divided into three pairs: two female-male and one male-male. The presence of game creators in this plot was minimized also because they provided a leaflet where players could read the game's rules. We obtained excellent feedback, which led to the creation of the final version of SERGE. In particular, one of the aspects most appreciated by all participants was the choice to play in pairs rather than as individual players; it was stated that *"playing in pairs is more productive, as I can discuss with my partner to reach an internal agreement and get more feedback"*. Among the critical issues that emerged during the game, some participants stated *"finding the information in the leaflet is complex, it would be convenient to have boxes or cards to summarizing the most important information"*. Regarding the game material [1], all players preferred to have an ID for each risk card to simplify consultation. In addition, one of them stated *"it would be helpful to have a picture for each card so that they can understand its use from the beginning. Moreover, to simplify the resource acquisition, it would be useful to provide players with a note sheet where they can draft their strategy"*. By the end of the second pilot phase, it took players about 10 minutes to read the leaflet. For the first phase, it took an average of 27 minutes to agree on the impact and probability of each risk card. The second phase of resource acquisition lasted about 10 minutes. The last phase lasted 10-15 minutes for each sprint, for 30-45 minutes. We estimated the game's duration to be about one and a half hours.

> **Summary of pilots test**
>
> According to the feedback obtained by the pilots test, the following changes were made:
> - add events cards to create a dynamic scenario for the risks encountered;
> - use of tokens as coins;
> - use of a customized set of cards for planning poker;
> - divide players into pairs;
> - creation of info cards to summarise game rules;
> - use of logos on cards to make the material more intuitive;
> - provide a note sheet;

6 *SERGE* EVALUATION AND VALIDATION: A CONTROLLED EXPERIMENT

This section meets RO$_3$ through evaluating and validating SERGE, analyzing whether its use will improve risk management knowledge and skills acquired during a traditional lecture. Therefore, we conjure and investigate the following research question:

> **Research Questions:**
>
> *Does the adoption of SERGE as an educational instrument increase students' risk management knowledge and skills?*

To conduct the experiment, we set as our **independent variable** the use of serious games after attending the traditional lecture on risk management. The **dependent variable** analyzed are (1) the answers to the questionnaire and (2) knowledge about risk management provided by the traditional lecture. Moreover, a **cofactor** that could exist for our experiment is related to the participants' prior experience in risk management.

Context of the study and Participants. To validate SERGE, we relied on the *"Software Project Management"* (SPM) course held during the second year of the Master's Degree in Computer Science at the University of Salerno. The course aims to acquire the appropriate skills to conduct software project management. During the course, students play the role of project management, managing the entire development process of a software product. To each manager are assigned a team of seven bachelor's degree students, which are attending the *"Software Engineering"* (SE) course. The bachelor's degree students—i.e., *Team Members* (TM)—must develop the software product proposed by the Master's students—i.e., *Project Managers* (PM). To pass the exam, PMs must estimate costs, time, and resources for the project development. Since it is an academic project, the budget is composed of hours/work, and the resources are the bachelor's degree students. Moreover, the PM should define tasks and deadlines and review the work and progress of the project.

The experiment was conducted in the first semester of the academic year 2022/2023. In this academic year there was 30 students attended the SPM course; among them, we chose a sample of 20 students (40% female and 60% male) and divided into project managers pairs (3 female-female pairs and 7 male-male pairs). 5 were selected as Group A (2 female-female pairs and 3 male-male pairs). Other 5 pairs (one female-female and the other 4 male-male pairs) were chosen to represent Group B (i.e., the Control group). Participants were chosen *no-probabilistically*, specifically in a **Convenience Sampling** approach [7]. Before selecting the participants, we asked them if they had any prior experience with risk management to ensure that all the participants had the same knowledge. The experiment was conducted after the participants attended the risk management lecture in the SPM course.

6.1 Experiment Design

We applied the *Completely Randomised Design* [8] to conduct the experiment. We identified the risk management skills as a factor and as our treatments (*Treatment 1:*) Fill out the knowledge and skills questionnaire after the risk management lecture; and (*Treatment 2:*) use of SERGE after the risk management lecture and fill out the knowledge and skills questionnaire. The material provided to the participants were (1) the link to the questionnaire on Google Forms, (2) the game material [1], and (3) additional sheets and pens for notes. The entire material was produced in their mother tongue to keep participants focused on the learning factor and ensure that all players understood the game and the questionnaire as well as possible. We divided our experiment into steps: *Step A* and *Step B*. To answer our research question, we compared the results of both steps; they provided information about participants' risk management skills and knowledge, so they are not sequential. The entire experiment design is summarised in table 3.

Table 3: Design of the Experiment

	Group	Treatment 1	Treatment 2
Step A	Group A_{BS}	Questionnaire S_{BS} — SeSa Lab	
	Group A_{AS}		SERGE & Questionnaire S_{AS}— SeSa Lab
Step B	Group B	Questionnaire S_B — SeSa Lab	

Step A. This step was conducted at the Software Engineer (SeSa Lab) at the University of Salerno. We aimed to immediately compare the outcomes obtained before and after using SERGE, applying both treatments to the same Group. To identify the answer of the same group, but at different times, we called it Group A_{BS} (before using SERGE) and Group A_{AS} (after using SERGE).

Participants answered to the first questionnaire; their answer were labeled with S_{BS} After, they were divided into pairs to take part in SERGE; to maintain their anonymity, each was assigned an ID (according to the couples, they were A11, A12, A21, A22, A31, A32, A41, A42, A51, and A52). Once that SERGE had been completed, the participants were again given the same questionnaire. The answers to that questionnaire are those labeled S_{AS}.

After using SERGE, the participants were interviewed to collect their opinions and experiences among the use and design of SERGE. These interviews took place separately for each participant immediately after using the game. Some of the questions were:

- *Do you find the Serious Game SERGE complex to use?*
- *Would a digital version be better than the version you used?*
- *What aspects did you enjoy most about using SERGE?*
- *Which aspects would you change in SERGE, and why?*

Step B. This step aims to confirm the results obtained from the first phase, with a control group labeled as Group B. This step was conducted at the SeSa Lab at the University of Salerno; where participants of Group B were subjected to treatment 1; since they only had to complete the questionnaire, providing them with an identifier was unnecessary.

6.2 Analysis of the Results

We compare the results obtained from different surveys to answer our research question. The results show increased risk management knowledge using SERGE. This data seems to indicate the educational value of the serious game, demonstrating its effectiveness as a teaching support tool.

Specifically, Figure 5 shows the number of correct answers, i.e., Score, obtained by players in the three questionnaires; The Score distribution testifies to the improvement of risk management skills for the participants of the Serious Game; particularly, 2 players obtained all answers correctly, and two others obtained a score very close to it. Moreover, the distribution range of the results is changed; before taking part in SERGE, players obtained a score between 4 and 11. After using SERGE, players' scores are between 6 and 20, meaning all answers were correct.

Analyzing in detail the number of correct answers for each question in the three different questionnaires, we showed an improvement of 85% for the questionnaire of Group A_{AS}, and only 15% of

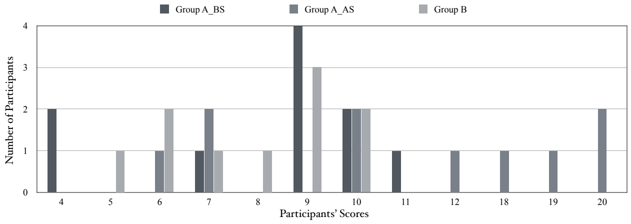


Figure 5: Plot of Distribution of Scores and Number of Participants.

them remained unchanged. The only exception is answer 12, in which the control group got more correct answers than Group A_{AS}.

All participants said they enjoyed the game, particularly when collaborating and communicating in the knowledge-sharing phase, stated "*I enjoyed the phase where each player shared their opinions on risks. Knowing other people's viewpoints helped me enlarge how I approach things. It was enlightening.*". Regarding the involvement of the game and the proposed mode, they stated *"Of course, a digital version could simplify many phases, but in this way, you'll lose the game's beauty. There will be no more communication and involvement, the interaction between the various players will be lost"*. Moreover, participants also appreciated the game materials, especially the info cards, stated *"The presence of the info cards made it much easier to use the game. They were an excellent idea"*.

To ensure the validity of SERGE, we also applied the T-test to the results obtained for our experiment. The T-test is a statistical test used to determine if there is a significant difference between the means of two groups [17]. To reject our null hypothesis, which posits that the results of the two groups are equal, we aimed to obtain a *p-value* less than 0.05. The result of the T-test shows a *p-value* of **0.017**, so we are confident about the results' validity.

In conclusion, the survey results not only seems the ability of gaming to improve knowledge but also could indicate its potential to improve skills in risk management. These results might confirm the importance of integrating gamification and simulation into educational settings, as they provide a dynamic and engaging means for students to acquire and apply knowledge effectively.

7 DISCUSSION AND LIMITATIONS

This section discusses the implementation and design choices, the results obtained, and possible future work. Moreover, we discuss the threats to the study's validity and how we mitigated them.

7.1 Discussion

Digital VS No-Digital Game. Petri et al. [25] in their analysis highlighted the difference between a digital and non-digital game. Digital games automate gameplay making the game easy to play, but limit flexibility and reduce collaboration. Non-digital games are adaptable, promote physical interaction, and enhance social interaction and strategic thinking. Still, it can be hard for players to remember and apply all the rules or complex game mechanisms. The participants themselves found such a non-digital version of SERGE immersive and engaging. However, they felt it necessary to build a digital part to succumb to some complex game mechanics. A good compromise to maintain the strengths of both versions is given to *Metaverse* [2]. The Metaverse wants to bridge this gap by leveraging digital technologies to create interconnected and interactive spaces. It maintains the excitement and expansiveness of digital gaming while promoting authentic social interaction, thus offering a compromise that supports the benefits of digital and non-digital gaming experiences [9].

Outlier Value. SERGE results confirm that adopting a serious game helps improve risk management knowledge and skills. However, there is a presence of an outlier in the results. The answer provided by Group *B* to question 12 was one point higher than the one of Group A_{AS}. We conjure that it happened because the game may have confused the participants. Nevertheless, this hypothesis was disproved by the results of Group *A*. Comparing the results given to response 12 in the two questionnaires of Group A, we found an increase in correct answers after using SERGE. This leads us to hypothesize that this was an isolated case. We aim to conduct further tests to confirm this theory, repeating the experiments.

7.2 Threats to Validity

This section illustrates the threats to the validity of the work and how we mitigated them. We identified and organized the threats using the well-known framework proposed by Wohlin et al. [31].

7.2.1 Threat to Internal Validity. One of the main threats to validity is the possibility that external factors may influence the results obtained, particularly to prevent the "learning factor" that could occur in Group A. We randomly assigned participants to the two groups to balance knowledge to avoid this. Moreover, to ensure that all participants had the same initial skills, we asked if they had prior experience in risk management outside of the traditional lecture provided in the course. Time gaps were introduced between the first questionnaire, the use of Serge, and the final questionnaire. Finally, participants did not have to learn the game's notions to answer the questionnaire. It was designed with scenarios through which the participants had to develop a risk mitigation and prevention strategy and provide the appropriate responses.

7.2.2 Threat to Conclusion Validity. One of the fundamental goals of conducting an experiment is to ensure the replicability of its results. At the same time, one of the biggest threats is not getting the same results by applying the treatment twice. To assess the validity of the results and the replicability of the experiment, we divided it into two phases, conducting a double comparison of the results. First, we compared the results of group A before and after using Serge. Afterward, we compared the results of the control group with one of the participants in Serge. Moreover, other threats to the validity of the experiment may be determined by the environment in which it is conducted. An environment exposed to the presence of other people could influence the participants, who might become distracted, affecting the responses that the participants would provide. The experiment was conducted in a silent and controlled place, the SeSa Lab at the University of Salernoto avoid this risk.

7.2.3 Threat to External Validity. One of the main threats we might have encountered while making the game design was non generalizability related to the end user and learning objectives. To avoid it, we conducted a pilot test to improve game design and assess learning objectives and arguments. To ensure the generalizability of the results, the students subjected to the controlled experiment were chosen using the Completely Randomised Design. Moreover, we focused on implanting a game method through context simulation that supports the player to reason and create strategies with appropriate resources.

8 CONCLUSIONS AND FUTURE WORK

Teaching Risk Management for the agile development model is fundamental to enriching the skills of future software professionals. The main objective of our work is SERGE, a serious game dedicated to supporting risk management teaching and improving the participants' skills and knowledge. It is designed, incorporating features extracted from already validated serious games in literature. Moreover, it also aims to fill the gaps in those games. The content and design behind it are built with an iterative improvement process to ensure optimal quality and effectiveness. Finally, its validation is confirmed through a controlled experiment involving students attending the software project management course at University of Salerno. Our results confirm that employing alternative methods, such as serious games tailored to specific arguments, improves students' knowledge of that topic. In addition, it is possible to learn in an engaging and fun environment by exploiting the strengths of gamification and simulation.

Starting from the results obtained and the discussion points, we conjure the following future works to improve SERGE:

- Enhance the material by providing additional scenarios and contexts. Moreover, players can create and introduce other risks different from those offered by the game; thus increasing knowledge sharing and the educational factor.
- The metaverse blends the benefits of both digital and non-digital games by integrating social interaction from non-digital environments with digital technologies to form interconnected spaces. A future iteration of SERGE in the metaverse could address players' identified shortcomings.

9 DATA AVAILABILITY

The replication package containing the game materials, the questionnaire, and the results are available in the online appendix [1].

REFERENCES

[1] Giusy Annunziata, Stefano Lambiase, Fabio Palomba, and Filomena Ferrucci. 2023. Replication Package – SERGE-Serious Game for the Education of Risk Management in Software Project Management. https://doi.org/10.6084/m9.figshare.24304654

[2] Umberto Bilotti, Dario Di Dario, Fabio Palomba, Carmine Gravino, and Maurizio Sibilio. 2023. Machine Learning for Educational Metaverse: How Far Are We?. In *2023 IEEE International Conference on Consumer Electronics (ICCE)*. IEEE, 01–02.

[3] Luigi Buglione, Christiane Gresse von Wangenheim, Alain Abran, Fergal Mc Caffery, and Jean Carlo Rossa Hauck. 2016. Risk Management: Achieving Higher Maturity &amp; Capability Levels through the LEGO Approach; Risk Management: Achieving Higher Maturity &amp; Capability Levels through the LEGO Approach. *2016 Joint Conference of the International Workshop on Software Measurement and the International Conference on Software Process and Product Measurement (IWSM-MENSURA)* (2016). https://doi.org/10.1109/IWSM-Mensura.2016.33

[4] Marly Monteiro De Carvalho and Roque Rabechini Junior. 2015. Impact of risk management on project performance: the importance of soft skills. *International journal of production research* 53, 2 (2015), 321–340.

[5] Narciso Cerpa and June M. Verner. 2009. Why did your project fail? *Commun. ACM* 52 (12 2009), 130–134. Issue 12. https://doi.org/10.1145/1610252.1610286

[6] Rafael Oliveira Chaves, Christiane Gresse von Wangenheim, Julio Cezar Costa Furtado, Sandro Ronaldo Bezerra Oliveira, Alex Santos, and Eloi Luiz Favero. 2015. Experimental Evaluation of a Serious Game for Teaching Software Process Modeling. *IEEE Transactions on Education* 58, 4 (2015), 289–296. https://doi.org/10.1109/TE.2015.2411573

[7] Juliet Corbin and Anselm Strauss. 2014. *Basics of qualitative research: Techniques and procedures for developing grounded theory.* Sage publications.

[8] John W Creswell and J David Creswell. 2017. *Research design: Qualitative, quantitative, and mixed methods approaches.* Sage publications.

[9] Dario Di Dario, Umberto Bilott, Maurizio Sibilio, Carmine Gravino, and Fabio Palomba. 2023. Toward a Secure Educational Metaverse: A Tale of Blockchain Design for Educational Environments. In *2023 49th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*. 159–166. https://doi.org/10.1109/SEAA60479.2023.00032

[10] B. Goodrich. 2019. *CAPM Exam Prep Study Guide: Belinda's All-in-One Program for Exam Success.* Goodrich Fremaux Publishing. https://books.google.it/books?id=YvCbvwEACAAJ

[11] Christiane Gresse-Von-Wangenheim, Rafael Savi, and Adriano Ferreti Borgatto. 2013. SCRUMIA - An educational game for teaching SCRUM in computing courses. *Journal of Systems and Software* 86 (10 2013), 2675–2687. Issue 10. https://doi.org/10.1016/j.jss.2013.05.030

[12] Muhammad Hammad, Irum Inayat, and Maryam Zahid. 2019. Risk management in agile software development: A survey. *Proceedings - 2019 International Conference on Frontiers of Information Technology, FIT 2019*, 162–166. https://doi.org/10.1109/FIT47737.2019.00039

[13] Haneen Hijazi, Thair Khdour, and Abdulsalam Alarabeyyat. 2012. A Review of Risk Management in Different Software Development Methodologies. , 975-8887 pages. Issue 7.

[14] Hooman Hooda and Hassan Rashidi. 2019. Classification and Analysis of Risks in Software Engineering. *World Academy of Science, Engineering and Technology* (2019).

[15] Project Management Institute. 2021. *A Guide to the Project Management Body of Knowledge* (7 ed.). 250 pages.

[16] Carlos Mario Zapata Jaramillo, María Clara Gómez Álvarez, and Guillermo González-Calderón. 2013. Riskware: A Game for Teaching Software Project Risk Management.

[17] Tae Kyun Kim. 2015. T test as a parametric statistic. *Korean journal of anesthesiology* 68, 6 (2015), 540–546.

[18] Gabriela Kiryakova, Nadezhda Angelova, and Lina Yordanova. 2014. GAMIFICATION IN EDUCATION.

[19] Yu Liu, Tong Li, Zheqing Huang, and Zhen Yang. 2023. BARA: A Dynamic State-based Serious Game for Teaching Requirements Elicitation. In *2023 IEEE/ACM 45th International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET)*. IEEE, 141–152.

[20] Fernando Rodríguez López, Mario Arias-Oliva, Jorge Pelegrín-Borondo, and Luz María Marín-Vinuesa. 2021. Serious games in management education: An acceptance analysis. *International Journal of Management Education* 19 (11 2021). Issue 3. https://doi.org/10.1016/j.ijme.2021.100517

[21] Etiane Marcelino and Luísa Domingues. 2021. An analysis of how well serious games cover the PMBOK. *Procedia Computer Science* 196, 1013–1020. https://doi.org/10.1016/j.procs.2021.12.104

[22] Jhon Masso, Francisco J. Pino, César Pardo, Félix García, and Mario Piattini. 2020. Risk management in the software life cycle: A systematic literature review. https://doi.org/10.1016/j.csi.2020.103431

[23] Kjetil Moløkken-Østvold, Nils Christian Haugen, and Hans Christian Benestad. 2008. Using planning poker for combining expert estimates in software projects. *Journal of Systems and Software* 81, 12 (2008), 2106–2117.

[24] Maruf Pasha, Ghazia Qaiser, and Urooj Pasha. 2018. A critical analysis of software risk management techniques in large scale systems. *IEEE Access* 6 (2 2018), 12412–12424. https://doi.org/10.1109/ACCESS.2018.2805862

[25] Giani Petri, Alejandro Calderón, Christiane Gresse-Von-Wangenheim, Adriano F Borgatto, and Mercedes Ruiz. 2018. Games for Teaching Software Project Management: An Analysis of the Benefits of Digital and Non-Digital Games. *Journal of Universal Computer Science* 24 (2018), 1424–1451. Issue 10.

[26] J. Schell. 2008. *The Art of Game Design: A book of lenses (1st ed.).* CRC Press. https://doi.org/10.1201/9780080919171

[27] K. Schwalbe. 2010. *Information Technology Project Management, Revised.* Cengage Learning. https://books.google.it/books?id=TYMIAAAAQBAJ

[28] Pontakorn Sonchan and Sakgasit Ramingwong. 2015. ARMI 2.0: An online risk management simulation. *ECTI-CON 2015 - 2015 12th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology.* https://doi.org/10.1109/ECTICon.2015.7207043

[29] Gil Taran. 2007. Using Games in Software Engineering Education to Teach Risk Management.

[30] Adam Trendowicz, Ross Jeffery, Adam Trendowicz, and Ross Jeffery. 2014. Planning Poker. *Software Project Effort Estimation: Foundations and Best Practice Guidelines for Success* (2014), 327–338.

[31] Claes Wohlin, Per Runeson, Martin Höst, Magnus C Ohlsson, Björn Regnell, and Anders Wesslén. 2012. *Experimentation in software engineering.* Springer Science & Business Media.