Challenges to Use Role Playing in Software Engineering Education: A Rapid Review

Mauricio Hidalgo[1,3(✉)] , Hernán Astudillo[2,3] , and Laura M. Castro[4]

[1] Escuela de Informática y Telecomunicaciones - Universidad Diego Portales, Santiago, Chile
mauricio.hidalgo@mail.udp.cl, mauricio.hidalgob@usm.cl
[2] ITiSB - Universidad Andrés Bello, Viña del Mar, Chile
hernan@acm.org
[3] Universidad Técnica Federico Santa María, Santiago, Chile
[4] Universidade da Coruña, A Coruña, Spain
lcastro@udc.es

Abstract. Role playing is a teaching method widely used to enhance students learning and engagement, by allowing them to adopt specific roles and interact with others in simulating real-world scenarios, thus applying their theoretical knowledge in a practical context. In Software Engineering Education (SEE), role playing may help students to develop key skills (like teamwork, problem-solving, and critical thinking), to understand the complexities and challenges of software development, and to appreciate the importance of collaboration and effective communication. To use role playing effectively, SEE teachers need to understand the challenges that arising from using it. This paper presents the design, execution and results of a *rapid literature review* to identify these challenges. Several well known digital libraries (Web of Science, Scopus, and IEEE Xplore) yield 44 articles, which after inclusion/exclusion filters left a total of 23 articles. Key findings are that: (1) most role playing is used to teach skills linked to software development and teamwork/"soft skills", and secondarily to software design, quality assurance, and process management; rather than project management or quality assurance; (2) challenges and generic considerations for implementing role playing were identified; and (3) challenges for applying role playing in SEE were identified by virtue of the SE specialty classification. In summary, role playing is a mature teaching technique used to in other fields, and has made limited inroads in SEE, mainly in disciplines dear to agile development (like development and teamwork).

Keywords: Software engineering education · Role playing · Rapid review

1   Introduction

Software Engineering is a constantly evolving discipline that requires technical and soft skills. In the educational context, students prefer hands-on work to

theory [3]. In turn, SE teachers have used several approaches to address the socio-technical aspects, such as role-playing for improving the teaching-learning experience [4]. One popular method for improving the teaching-learning experience is role-playing [4]. For both pedagogical and ethical reasons, teachers must consider how formatting and preparation can impact learning outcomes [5]. This rapid review explores the use of role-playing in software engineering education to identify the challenges and considerations to take into account when implementing role-playing in the context of software engineering education.

The remainder of this paper is structured as follows: Sect. 2 introduces the theoretical background and the key concepts of SEE and Role Playing (RP); Sect. 3 relates some previous experiences of the use of RPin Engineering and SEE; Sect. 4 describes Rapid Review (RR) and the research protocol employed to guide our work; Sect. 5 presents the review's key findings of the study considering the generic and specific (for SEE) challenges and recommendations; Sect. 6 presents a results summary; Sect. 7 presents the possible threats to the study validity and the measures taken to mitigate them; and Sect. 8 summarises and concludes the paper.

2   Background Concepts

2.1   Software Engineering Education

Software Engineering Education (SEE) can be understood as the process of teaching students the principles, concepts and skills necessary to design, develop, test and maintain quality software. In this context, software developers education must adapt in the same way as the discipline itself, and not only with the recent addition of online training for computing skills [33]. While it is important to acknowledge and address the practical aspects of the technology, tools, and methodologies, students and teachers should also be equipped with skills that enable them to comprehend and master the ever-changing landscape of software development. Considering the above, [34] defines a (non-exhaustive) list of skills and knowledge that the software engineer must master like the theoretical foundations, design methods, technology and tools of the discipline. In addition, they have to be able to keep their knowledge up-to-date with respect to the new approaches and technologies, interact with other people, manage the development process, understand, model, formalize, analyse and recognize problems, reuse or adapt known solutions, and coordinate the work of different people. This list points to, in one hand, acquiring the theoretical foundations of a discipline (commonly accomplished through traditional schooling). However, managing a process includes the complexities of interpersonal interactions (with human stakeholders or teammates), and often demands hands-on experience and presents distinct challenges that vary based on the specific context. Hence, it is important that SEE incorporates techniques to allow students to develop relational skills through the practice of roles (like Role Playing), tasks, collaborative work and other areas that allow the development in a human scale environment.

2.2 Role Playing

Role playing educational activities have their origin in Role-Playing Games (RPGs) and, in view of that origin, they share a core base formed by rules for the game mechanics, stories (modules) that give meaning and context to the actions of characters, and means of social interaction through which a story is co-created [6]. The integration of character development with mastery design within a social context gives the potential to develop learning skills such as communication, problem-solving, and leadership, while presenting the learner with narrative agency within the curriculum [19]. Considering the above, role playing (sometimes referred to as role-play simulation in educational settings) is an experiential learning method in which learners are involved in a proposed scenario by representing an interacting part in it [20]. The two baseline concepts for role playing are:

– **Roleplayer or Interactor:** The person who develops or improvises a role as part of a scenario. Roles can be performed by individual students, in pairs, or in groups which can play out a more complex scenario [21].
– **Scenarios:** The scenarios are situations (not necessary a simulationin which two or more people act out in specific roles. The scenario is outlined by the teacher or professional, and while it must allow improvisation, it represents a safe and supportive environment where students will develop their own meaningful first-person experience [20].

In short, role playing exercises give students the opportunity to assume the role of a person or act out a given situation [22].

3 Related Work

The use of role playing as a pedagogical strategy in Higher Education has been the subject of interest and study in the academic community, and its applications are very diverse given the versatility of the roles and scenarios. In Engineering, a pilot study generating classroom STEM RP simulations to meet course learning objectives was conducted to gain deeper insight into the barriers to both adoption and sustained utilization of RP course content from several areas of engineering problem-solving [7]. In a closer environment to our speciality, [8] explores the potential of role engineering education and project development for engineering students using role playing in a case study for Human-Automation Systems, and [9] created and implemented a role-play case study in an undergraduate computing data mining course to improve ethical understanding of algorithms among computing students. In the SEE field two of the first role playing uses (not directly mentioned) were reported in 1987 [10,11], and expose an experience of teaching a senior-year course on software maintenance, centred around a maintenance project by the use of clients and groups roles and a maintenance project as scenario [10]; and a comprehensive description of one way of organizing and presenting a project course where components are well-trained

advanced students, a minimal product, an elaborate engineering process, good tools, and task assignments for external participants, deliverables, and collateral duties [11]. Subsequently, new studies emerged in SEE to demonstrate that empirical study projects give students experience in research [12]; to allow acquiring competences by using Case Method and Role Play as instruments in several lectures of Software Engineering for teaching Finite State Machines [13]; by the proposing of an active learning approach that uses interactive role-play simulations in a virtual 3D environment for understanding Object-Oriented Software [14] and programming language concepts and skills by a Web-based Multiplayer Online Role Playing Game [15]; by the developing a two-and-a-half-day role-playing workshop for engineers that focuses on teaching the importance of RE, the background, rationale, and purpose of the requirements, as well as the actual requirements [16]; to present an assessment approach for teamwork performance in software engineering education for encourages and supports student active and collaborative learning by using an approach specially assessing teamwork performance of a team and each team member of the team [17]; and for giving the opportunity for students to appreciate the value of software design principles or even to learn how to apply principles in practice [18].

In summary, related work indicates that Role Playing is an intriguing approach in Software Engineering Education, fostering both relational and technical skills. This RR seeks to offer a more comprehensive insight into this aspect.

4   Research Method: Rapid Review

This section summarizes the Rapid Review (RR) research protocol employed as the research method to guide our work. Rapid Reviews (RR) are practice-oriented secondary studies, and their main goal is to provide evidence to support decision-making towards the solution, or at least attenuation, of issues practitioners face in practice [1]. Rapid Reviews were first mentioned in the literature in 1997, when [32] described the rapid health technology assessment program in the south and west regions of England, but did not provide a formal definition [31]. To characterise them, a rapid review is an accelerated way of conducting a systematic review of the scientific literature [31]. Unlike a traditional systematic review, which involves a long and detailed process, a rapid review seeks to perform a rapid evaluation of the existing evidence in a shorter period of time. The main objective of a rapid review is to provide a fast-delivered, summarized synthesis of the available evidence to inform decision-making in a specific context [30]. Considering the above, we will guide our research under this approach following the guidelines indicated in [2,30] for a RR process. Therefore, our RR protocol has the following assumptions:

- **Assumption 1.** The SEE teachers and researches have had time to adopt Role-Playing in the past 10 years.
- **Assumption 2.** The researches making empirical experiments with Role-Playing have documented their findings both positively and negatively.

– **Assumption 3.** There is no evidence on the challenges comprising the implementation of Role Playing in SEE beyond the "generic guidelines" (for any speciality) that can be found in the formal and grey literature.

4.1 Definition of Goal and Research Question

This Rapid Review (RR) is undertaken with the objective of providing a comprehensive perspective on the challenges associated with the utilization of Role-Playing in Software Engineering Education (SEE). To achieve this objective, we formulated the following research question:

– **Research Question (R.Q.).** "What's are the **challenges** to implementing role playing in software engineering education?" ***Rationale:*** Considering the diverse contexts within Software Engineering, compiling a list of challenges (and considerations) will empower educators and researchers to implement the Role Playing technique while tailoring it to the unique aspects of their courses or empirical experiments.

4.2 The Primary Studies Search and Selection Process

This section outlines the process employed to search and select primary studies for this paper. The process, depicted in Fig. 1, culminated in the identification of 23 selected papers for this RR. This selection was achieved through the following sequence of steps:

Search on Scopus, WoS and IEEE Xplore. A search was performed in the WoS and Scopus Databases and the IEEE Xplore digital library considering:

– The keywords: "Role Playing" and "Software Engineering Education"
– Date of searches: Papers published between January 2012 and March 2023.

Automated Duplicates Removal. The papers were filtered using a Python script that allowed us to eliminate documents with the same content.

Application of the Inclusion/exclusion Criteria. For this research we considered the inclusion (In) and exclusion (Ex) criteria. To be included in the RR, the paper must accomplish all of the next conditions:

– (In.) The paper must be in the context of SEE
– (In.) The paper should be explicit in the use of Role Playing
– (In.) The paper must report an empirical evidence-based
– (In.) The paper must be a primary study
– (In.) The paper must be written in English or Spanish

To be excluded in the RR, the paper must satisfy one (or more) of the next conditions:

– (Ex.) The paper publish date is lower than 2012
– (Ex.) The paper include the same author of other most recent paper with similar content
– (Ex.) The papers is not indexed in WoS or Scopus

Manual Papers Inclusion. The paper repository were complemented with suggested sources [30] considering the expertise in software engineering education of the research team. Notwithstanding, the suggested papers also went through the automated duplicate elimination process and the inclusion/exclusion criteria.
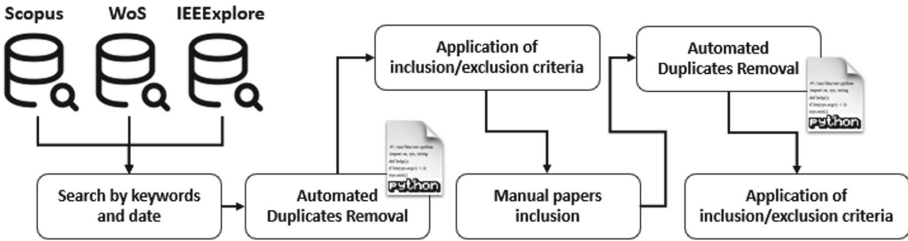

Fig. 1. Search and Selection Process

4.3   Data Extraction Process

Considering the [1,2] guidelines, a Rapid Review usually use a descriptive synthesis method rather than quantitative meta-analysis. For the extraction of data (necessary to answer the research question), a manual review process of the papers was carried out with the objective of generating a classification and a descriptive synthesis of each one of them. The following fields were considered to the data extraction for each paper:

– Description: Contains the description of the analyzed paper.
– Application domain (SE Category): Contains the main category to which the paper contributes in Software Engineering.

Category Classification. Among the 23 selected papers, the primary categorization within the field of Software Engineering yielded the following breakdown: 2 papers focused on Process Management, 2 on Quality Assurance, 2 on Software Architecture, 4 on Requirements Engineering, 5 on Teamwork and Soft Skills, and 8 on Software Development. This categorization information was integrated into the descriptive synthesis for each respective paper.

Descriptive Synthesis. For the descriptive synthesis, we chose to generate a summary that exposes the main ideas and/or proposals of each of the papers. Likewise, the classification of each paper towards a category of Software Engineering was included. The summary of descriptive synthesis is in Table 1.

Table 1. Rapid Review selected papers summary

N°	Paper Name	Short Description	SE Category
01	Role-playing software architecture styles	Propose an innovative use of role-playing as teaching strategy for architecture models of reference (i.e. layered, pipe filter, client-server, etc.)	Software Architecture
02	Applying Role-Playing game in Software Development	Application of Role-playing game where they elicit the software requirements through an interview with the teacher that plays the role of the end user	Software Development
03	Software Engineering Role-Playing Game: An interactive game for Software Engineering Education	Support the learning process through the simulation of a software company environment, challenging the students with the activities and problems from the development process management	Process Management
04	Towards a Role Playing Game for Exploring the Roles in Scrum to Improve Collaboration Problems	The goal of this study is to address such situations using an interactive role playing game-based approach among team members to improve collaboration	Teamwork / Soft Skills
05	Is role playing in Requirements Engineering Education increasing learning outcome?	Investigate whether a higher grade in a role playing project have an effect on students' score in an individual written exam in a Requirements Engineering course	Requirements Engineering
06	Swords and sorcery a structural gamification framework for higher education using role-playing game elements	This is achieved through implementing game elements to the entire second-year cohort (N = 34) of computer game development students, in the unit 'Engineering Software Systems'. The goal is to motivate and engage threat-risk students of the cohort with lower activity, attendance and involvement in the unit	Software Development
07	ATAM-RPG: A role-playing game to teach architecture trade-off analysis method (ATAM)	Describes ATAM-RPG, a role-playing game to support the teaching of ATAM by simulating stakeholder's interaction and trade-offs	Software Architecture
08	A Role-Playing Game for a Software Engineering Lab: Developing a Product Line	Report an experience of ours, showing how in the context of a software engineering course at University of Bologna our students tackled the task of developing a software product line consisting of four products which were variants of a basic shared asset	Software Development
09	A Capstone Course on Agile Software Development Using Scrum	Discusses the achievement of teaching goals and provides empirical evaluation of students' progress in estimation and planning skills	Process Management
10	Applying a maturity model during a software engineering course - How planning and task-solving processes influence the course performance	Report on a study and teaching approach where, in three successive semesters and at two different institutions, we started rating the process maturity of students solving tasks in our software engineering courses and transparently related the maturity levels to the task performances	Software Development

Table 1. (*continued*)

N°	Paper Name	Short Description	SE Category
11	A pattern-based approach for improving model quality	Constructed a catalog of anti-patterns of correctness and quality problems in class diagrams, where an anti-pattern analyzes a typical constraint interaction that causes a correctness or a quality problem and suggests possible repairs to improve modelers' capabilities	Quality Assurance
12	Impact of Practical Skills on Academic Performance: A Data-Driven Analysis	Explores the correlation between practical (e.g., programming, logical implementations, etc.) skills and overall academic performance and analyzing the statistical features of students	Teamwork / Soft Skills
13	A Quest to Engage Computer Science Students: Using Dungeons Dragons for Developing Soft Skills	Explains a didactic approach for engaging students into active learning for developing soft skills and to make them aware of the importance of this topic in relation to their professional future	Teamwork / Soft Skills
14	Verification and Validation Model for Short Serious Game Production	Presents a model based upon requirements traceability and inspections to verify and validate the correct implementation of requirements and the successful production of quality short serious games	Quality Assurance
15	Cultural and linguistic problems in GSD: a simulator to train engineers in these issues	Work focuses on the development of a virtual training environment that can simulate global software development scenarios involving virtual agents (VAs) from different cultures	Software Development
16	Applying virtual reality to teach the software development process to novice software engineers	In this study, a virtual reality-based software development framework (VR-SODEF) is proposed, which provides an interactive virtual reality experience for learning software development tasks, from requirements analysis to software testing	Software Development
17	Role-Play as an Educational Tool in Auditing Skills	The primary objective of this research is to develop auditing skills: communication, problem solving, and teamwork by using three different role-play activities with 62 students in a bachelors of accounting degree at Walailak University	Teamwork / Soft Skills
18	Design of a course oriented to the comprehension of Agile Methods based on Teamwork, Role-play, and Class Project with a Real Client	Presents the methodological strategy developed in an undergraduate elective course, whose main objective was the comprehension of the philosophy of Agile methods in engineering students	Teamwork / Soft Skills
19	Evaluating Role Playing Efficiency to Teach Requirements Engineering	The objective of this study is to assess the use of role playing and role reversal in an intensive requirements engineering course and reports the experience teaching requirements engineering courses using role playing for four sessions of two hours each	Requirements Engineering

<div align="right">(continued)</div>

N°	Paper Name	Short Description	SE Category
20	Leveraging Role Play to Explore Software and Game Development Process	This paper used role play techniques to engage the students more fully in the processes of development by creating a scenario in which the students have suddenly become the owners of an independent game studio	Software Development
21	Requirements Engineering Education using Expert System and Role-Play Training	This paper propose a method for use in requirements engineering education in a university, and based on 3 principles. (1) A type of expert system, (2) Group-work role-play training and (3) A software agent system	Requirements Engineering
22	Role Playing as a Method of Learning in Knowledge Engineering	This paper showns a Role Playing based procedure in wich Students are trained to interview experts to extract knowledge for intelligent systems and, at the same time, to argue their opinion	Requirements Engineering
23	Use of Role-play and Gamification in a Software Project Course	This paper discusses the use of team role-play activities to simulate the experience of working in a professional, game development studio as a means of enhancing an advanced undergraduate game design course	Software Development

5   Challenges to Role Playing Activity in SEE

In this section, we will examine the challenges outlined in the papers, acknowledging that they may fall into either a "generic" category or a "specific" category pertaining to one of the classifications within Software Engineering.

5.1   Generic Challenges

In several of the studied papers we noticed that Role Playing follows four generic steps to design an RP implementation [5]: (1) Preparation and explanation by the teacher, (2) Student preparation for the activity, (3) Develop of the Role Playing Activity and (4) Debrief and reflection of the experience. Notwithstanding this, there are challenges to be faced in each implementation and after a rigorous inspection of the 23 selected papers in light of the R.Q., several challenges were identified:

- **Time:** Designing, planning, executing, and assessing role playing activities requires a significant investment of time and dedication.
- **Adjustment period:** Students who are new to these types of activities will need an adjustment period to acquire the necessary skills for active participation. Otherwise, they may become distracted from the learning process.
- **Classroom setup:** Setting up the classroom for role-playing can be challenging, particularly when working within space constraints. Additionally, considering accessibility issues for both students and instructors can add to the complexity of the setup process.

– **SE subfield:** Given the breadth of Software Engineering, it is important to focus the Role Playing on a particular subfield. In this way, concrete results can be observed under defined scenarios.
– **Scenarios:** Considering the scenarios, one challenge is that improvisation can be allowed in a safe and supportive environment in which learners will develop their own expertise.

Notwithstanding the above, there are interesting points that remain unaddressed. These are:

– **Suitability for every student:** role play may not be suitable for every student, especially those who have special educational needs (e.g. ASD, Asperger Syndrome) and people who struggle with active participation or expressing themselves in class, because these activities can potentially generate anxiety and might hinder their overall performance.
– **Anti-bias:** depending on the context, certain role playing activities can be emotionally charged and confrontational; e.g. societal biases could cause problems for vulnerable students.
– **Professional profile:** in cases of higher courses, it was not possible to observe a profiling of students considering their technical skills or learning styles; hence, the results may not necessarily be the best for some particular students. For example, if a student whose *forte* is programming is assigned a management role, they may learn from the experience by virtue of not exploiting their core strength.

In any case, the aforementioned aspects correspond to the human aspect that is inherent to role playing (and to education in general).

5.2   SEE Category Challenges

In this section we will review, separately, the challenges extracted for each of the categories identified in the papers that make up the RR.

Requirements Engineering. The papers [24, 35, 37, 38] researched on the application of role playing in Requirements Engineering Education and the following challenges and recommendations were identified:

– The assigned role must be clear (client, developer, etc.) to develop and understood the activities to teach the elicitation, analysis, specification and requirements validation.
– The scope of each role must be specified for the prioritization of requirements.
– If an actor in the developer team role needs clarification from the customer, such interaction must be allowed. Therefore, roles should change as little as possible during the ongoing Role Play.
– The use of standards for requirements and modeling tools can be included in the activities of each role of the development team.

No evidence of "requirements' change" situations was found in the papers reviewed.

Process Management. The use of role-playing simulations improved understanding of project management challenges and promoted hands-on learning [25,45]. Considering it, the following challenges and considerations were identified:

– The project (scenario) must be clear and include explicit limitations.
– Considering the limitations, the student must take choices rather than following instructions. E.g.: Choose between developers considering their programming language knowledge or select the life cycle.
– It's important to add some reality. Include "challenges" like team members may briefly leave the project because they are sick or on holiday [25].

Quality Assurance. Likewise, the quality assurance observed in [46,47] shows how role playing can focus on meeting quality standards. The following challenges and considerations were identified:

– Being a transversal process to the SW development stages, it must be clear which of these stages will be considered for the scenario.
– There must be roles to measure and examine if the work complies with the requirements and acceptance criteria and roles to ensure the execution of the processes.
– If the scenario contemplates the complete development cycle, students should be prevented from focusing directly on development and neglecting the analysis and design phases.

Software Architecture. In order to use role-playing games in this context, one must be clear about the "subcategory" that one intends to teach. In this case, the following challenges could be identified:

– Define a clear scenario to teach tradeoffs. The pros and cons of each component in the scenario should be included in each of the choices.
– To teach about architectural models, consider each architectural model to teach as an scenario [20]. Define a role to each component in the system including their functionalities, limitations and interactions with other components and parts of the system.

Teamwork/Soft Skills. In order to focus the Role playing on the development of Teamwork/Soft skills in SEE it is mandatory that the scenario and roles are linked to another category (e.g. development). Then, the challenges and recommendations found for this point in SEE are:

– Ensure to develop an scenario to learning the relationships between technology and society
– Must define clearly each role and their responsibility in the development team or the stakeholder.

Software Development. The use of role-playing is focus for the development process itself. Considering it, the following challenges and considerations were identified:

- Must specify any domain as a part of the scenario.
- Must define a clear and unique flow for each scenario.
- Must define clear requirements for the developers role and remove ambiguities.
- In domain implementation, ensure each team implements a different part of the common assets.

6   Results Summary

The review presented yields several interesting findings:

- Most studies found on *role playing in SEE focus on software development and teamwork/soft skills*; this tips the balance to the operational over the tactical and strategic aspects of the software development process.
- *Roles are usually attributed to humans*, but a role may also be associated (with the appropriate approach) to a software component.
- *Time and adjustment periods* are mentioned as part of the limitations of each study, and should be considered as a full part of the role playing.
- Scenarios should be defined according to the subfield to be developed and its constraints; likewise, the scenarios can be face-to-face, virtual, or mixed.
- The generic findings apply for any category in Software Engineering, but the specifically may apply to the main category in first instance.

Furthermore, it's noteworthy that specific critical aspects were conspicuously absent in all of the papers under review:

- There is no mention of cases where students have *special educational needs* (e.g.: A.S.D., Asperger Syndrome).
- No evidence was found of *how roles are assigned* to students; whether the study had a single general role for them or several, the assignment rationale was not made explicit.
- No evidence was found either of explicit addressing of **societal or other biases** that may occur.
- No cases could be found where the requirements were changing.

7   Threats to Validity

Several possible threats to the validity of our study were identified and mitigated:

- **Documents search:** the search was conducted across WoS, Scopus, and IEEE Xplore databases, primarily due to the accessibility constraints imposed by the nature of RR studies. This choice serves to reduce the potential risk of including papers whose source cannot be reliably verified.

– **Language:** all documents that were not in English or Spanish were excluded. This measure was taken to minimize the risk of misinterpreting documents written in languages not understood by the researchers.
– **Quality Appraisal:** although this step was omitted, it's important to note that our search and selection process focused exclusively on studies published in conferences and journals known for their stringent review processes.
– **Results:** the interpretation of the findings is constrained by the inherent limitations of the RR process.

8   Conclusions and Future Work

This paper presents a Rapid Literature Review (RR) focusing on the utilization of Role Playing in Software Engineering Education (SEE). The RR encompassed an examination of the WoS, Scopus, and IEEE Xplore databases, resulting in a selection of 23 papers. The majority of these papers revolve around software development and the enhancement of teamwork and soft skills, with a considerable emphasis on Requirements Engineering. This suggests that Role Playing, regardless of its potential in other areas, primarily contributes to skill development for product creation and team formation, targeting the operational aspects of the software development process. Moreover, while most papers delve into roles associated with "people," one paper, [20], stands out for its unique perspective, illustrating the assignment of roles to components. Conversely, the discussion concerning the design of scenarios is notably limited or entirely absent. This could potentially be attributed to the assumption that scenarios are inherently intertwined with the essence of Role Playing, and their formalization presents a challenge, given the various constraints such as space, modality, and time, among others, that may impinge upon them. Despite this observation, there is a discernible surge in interest surrounding the application of Role Playing in SEE. It becomes apparent that the challenges in implementing this approach center around operational planning considerations and the requisite focus on specific circumstances pertinent to the target group, i.e., the students who will engage with the technique. Moreover, our study, aligned with the research question, has unveiled a range of challenges and considerations relevant to the broad application of Role Playing, both in generic terms and within specific Software Engineering contexts. This compilation of insights serves as a valuable resource for educators aiming to integrate this technique into their SE courses and for researchers exploring the use of Role Playing in SEE. Finally, and as future work, an intriguing opportunity lies in the absence of student profiling based on their individual abilities or learning styles. This void presents the potential for the incorporation of tools like "Kolb's Learning Model," which can facilitate the development of Role Playing in a manner tailored to enhancing each student's strengths through the allocation of roles that align with their aptitudes.

References

1. Cartaxo, B., Pinto, G., Soares, S.: Rapid reviews in software engineering. In: Contemporary Empirical Methods in Software Engineering, pp. 357–384. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-32489-6_13

2. King, V.J., Stevens, A., Nussbaumer-Streit, B., et al.: Paper 2: performing rapid reviews. Syst. Rev. **11**, 151 (2022). https://doi.org/10.1186/s13643-022-02011-5

3. Erdogmus, H., Péraire, C.: Flipping a graduate-level software engineering foundations course. 2017 IEEE/ACM 39th International Conference on Software Engineering: Software Engineering Education and Training Track (ICSE-SEET), Buenos Aires, Argentina, pp. 23–32 (2017). https://doi.org/10.1109/ICSE-SEET.2017.20

4. Henry, T., LaFrance, J.: Integrating role-play into software engineering courses. J. Comput. Sci. Coll. **22**, 32–38 (2006)

5. Stevens, R.: Role-play and student engagement: reflections from the classroom. Teach. High. Educ. **20**, 1–12 (2015). https://doi.org/10.1080/13562517.2015.1020778

6. Cheville, R.A.: Linking capabilities to functionings: adapting narrative forms from role-playing games to education. High. Educ. **71**(6), 805–818 (2016)

7. White, A.R.: Pilot study and survey to increase adoption and sustained utilization of simulations using role-play course content. In: IEEE Frontiers in Education Conference (FIE), vol. 2022, pp. 1–15. Uppsala, Sweden (2022)

8. Ponsa, P., Vilanova, R., Amante, B.: The use of role playing in engineering curricula: a case study in human-automation systems. In: IEEE EDUCON 2010 Conference, pp. 1335–1341. Madrid, Spain (2010). https://doi.org/10.1109/EDUCON.2010.5492373

9. Hingle, A., Rangwala, H., Johri, A., Monea, A.: Using role-plays to improve ethical understanding of algorithms among computing students. In: 2021 IEEE Frontiers in Education Conference (FIE), pp. 1–7. Lincoln, NE, USA, (2021)

10. Andrews, J.H., Lutfiyya, H.L.: Experience report: a software maintenance project course. In: Thirteenth Conference on Software Engineering Education and Training, pp. 132–139. Austin, TX, USA (2000). https://doi.org/10.1109/CSEE.2000.827031

11. Mc Keeman, W.M.: Experience with a software engineering project course. IEEE Trans. Softw. Eng. **SE− 13**(11), 1182–1192 (1987)

12. Host, M.: Introducing empirical software engineering methods in education. In: Proceedings 15th Conference on Software Engineering Education and Training (CSEE&T 2002), pp. 170–179. Covington, KY, USA (2002). https://doi.org/10.1109/CSEE.2002.995209

13. Joseph, S., et al.: Teaching finite state machines with case method and role play. In: 2013 IEEE Global Engineering Education Conference (EDUCON), pp. 1305–1312. Berlin, Germany (2013). https://doi.org/10.1109/EduCon.2013.6530275

14. Jimenez-Diaz, G., Gomez-Albarran, M., Gomez-Martin, M.A., Gonzalez-Calero, P.A.: Understanding object-oriented software through virtual role-play. In: Fifth IEEE International Conference on Advanced Learning Technologies (ICALT 2005), pp. 875–877. Kaohsiung, Taiwan (2005). https://doi.org/10.1109/ICALT.2005.293

15. Chang, M., Kinshuk.: Web-based multiplayer online role playing game (MORPG) for assessing students' Java programming knowledge and skills. In: 2010 Third IEEE International Conference on Digital Game and Intelligent Toy Enhanced Learning, pp. 103–107. Kaohsiung, Taiwan (2010). https://doi.org/10.1109/DIGITEL.2010.20

16. Nakatani, T., Tsumaki, T., Tamai, T.: Requirements engineering education for senior engineers: Course design and its evaluation. In: 2010 5th International Workshop on Requirements Engineering Education and Training, pp. 26–35. Sydney, NSW, Australia (2010). https://doi.org/10.1109/REET.2010.5633111.

17. Chen, J., Qiu, G., Yuan, L., Zhang, L., Lu, G.: Assessing teamwork performance in software engineering education: a case in a software engineering undergraduate course. In: 18th Asia-Pacific Software Engineering Conference, pp. 17–24. Ho Chi Minh City, Vietnam (2011). https://doi.org/10.1109/APSEC.2011.50

18. Jarzabek, S., Eng, P.K.: Teaching an Advanced design, team-oriented software project course. In: 18th Conference on Software Engineering Education & Training (CSEET 2005), pp. 223–230. Ottawa, ON, Canada (2005)

19. Snow, C.: Dragons in the stacks: an introduction to role-playing games and their value to libraries. Collect. Build. **27**(2), 63–70 (2008)

20. Castro, L.M.: Role-playing software architecture styles. In: 20th IEEE International Conference on Software Architecture (ICSA 2023) (2023)

21. Bonwell, C.C., Eison, J.A.: Active Learning: Creating Excitement in the Classroom. The George Washington University, Washington, DC (1991)

22. Northern Illinois University Center for Innovative Teaching and Learning. Role playing. In: Instructional Guide for University Faculty and Teaching Assistants (2012). https://www.niu.edu/citl/resources/guides/instructional-guide

23. Svensson, R.B., Regnell, B.: Is role playing in requirements engineering education increasing learning outcome? Requirements Eng. **22**(4), 475–489 (2016). https://doi.org/10.1007/s00766-016-0248-4

24. Rueda, S., Panach, J.I., Cabotà, J.B., Valverde, F.: Applying role-playing game in software development subjects. In: ICERI2016 Proceedings, pp. 3532–3538 (2016)

25. Benitti, F.B.V.: Software engineering role-playing game: an interactive game for software engineering education. Int. J. Adv. Res. Comput. Sci. **2**(2), 6–9 (2011)

26. Akarsu, Z., Metin, Ö.O., Gungor, D., Yilmaz, M.: Towards a role playing game for exploring the roles in scrum to improve collaboration problems. In: Larrucea, X., Santamaria, I., O'Connor, R.V., Messnarz, R. (eds.) EuroSPI 2018. CCIS, vol. 896, pp. 254–264. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-97925-0_21

27. McConville, J.R., Rauch, S., Helgegren, I., Kain, J.-H.: Using role-playing games to broaden engineering education. Int. J. Sustain. High. Educ. **18**(4), 594–607 (2017). https://doi.org/10.1108/IJSHE-08-2015-0146

28. Montenegro, C.H., Astudillo, H., Álvarez, M.C.G.: ATAM-RPG: a role-playing game to teach architecture trade-off analysis method (ATAM). In: 2017 XLIII Latin American Computer Conference (CLEI), pp. 1–9. Cordoba, Argentina (2017). https://doi.org/10.1109/CLEI.2017.8226416

29. Zuppiroli, S., Ciancarini, P., Gabbrielli, M.: A role-playing game for a software engineering lab: developing a product line. In: 2012 IEEE 25th Conference on Software Engineering Education and Training, pp. 13–22. Nanjing, China (2012)

30. Khangura, S., Konnyu, K., Cushman, R., Grimshaw, J., Moher, D.: Evidence summaries: the evolution of a rapid review approach. Syst. Control Found. Appl. **1**, 10 (2012). https://doi.org/10.1186/2046-4053-1-10

31. Hamel, C., et al.: Defining rapid reviews: a systematic scoping review and thematic analysis of definitions and defining characteristics of rapid reviews. J. Clin. Epidemiol. **129**, 74–85 (2021). ISSN 0895–4356

32. Best, L., Stevens, A., Colin-Jones, D.: Rapid and responsive health technology assessment: the development and evaluation process in the South and West region of England. J. Clin. Effectiv. **2**(2), 51–56 (1997)

33. Shaw, M.: Software engineering education: a roadmap. In: Proceedings of the Conference on the Future of Software Engineering, pp. 371–380 (2000)
34. Ouhbi, S., Pombo, N.: Software engineering education: challenges and perspectives. In: 2020 IEEE Global Engineering Education Conference (EDUCON), pp. 202–209. IEEE (2020)
35. Ouhbi, S.: Evaluating role playing efficiency to teach requirements engineering. In: IEEE Global Engineering Education Conference (EDUCON). Dubai, United Arab Emirates, pp. 1007–1010 (2019). https://doi.org/10.1109/EDUCON.2019.8725045
36. Decker, A., Simkins, D.: Leveraging role play to explore software and game development process. In: IEEE Frontiers in Education Conference (FIE), pp. 1–5. Erie, PA, USA (2016). https://doi.org/10.1109/FIE.2016.7757685
37. Nakamura, T., Kai, U., Tachikawa, Y.: Requirements engineering education using expert system and role-play training. In: 2014 IEEE International Conference on Teaching, Assessment and Learning for Engineering (TALE), pp. 375–382. Wellington, New Zealand (2014). https://doi.org/10.1109/TALE.2014.7062566
38. Kobrinskii, B.A.: Role playing as a method of learning in knowledge engineering. In: 2022 VI International Conference on Information Technologies in Engineering Education (Inforino), Moscow, Russian Federation, pp. 1–5 (2022)
39. Monasor, M.J., Vizcaíno, A., Piattini, M.: Cultural and linguistic problems in GSD: a simulator to train engineers in these issues. J. Softw. Evol. Proc. **24**, 707–717 (2012). https://doi.org/10.1002/smr.562
40. Gulec, U., et al.: Applying virtual reality to teach the software development process to novice software engineers. IET Soft. **15**(6), 464–483 (2021)
41. Maxim, B.R., Brunvand, S., Decker, A.: Use of role-play and gamification in a software project course. In: IEEE Frontiers in Education Conference (FIE), pp. 1–5. Indianapolis, IN, USA (2017). https://doi.org/10.1109/FIE.2017.8190501
42. Srisuwan, P.: Role-play as an educational tool in auditing skills, pp. 613–617 (2018). https://doi.org/10.1109/ICBIR.2018.8391271
43. Veldthuis, M., Koning, M., Stikkolorum, D.: A quest to engage computer science students: using dungeons & dragons for developing soft skills. In: Proceedings of the 10th Computer Science Education Research Conference (CSERC 2021), pp. 5–13. Association for Computing Machinery, New York, NY, USA (2022)
44. Rojas, A.E., Mejía-Moncayo, C.: Design of a course oriented to the comprehension of agile methods based on teamwork, role-play, and class project with a real client. In: 2019 International Conference on Virtual Reality and Visualization (ICVRV), pp. 212–216. Hong Kong, China (2019)
45. Mahnic, V.: A capstone course on agile software development using scrum. IEEE Trans. Educ. **55**(1), 99–106 (2012)
46. Barajas, A., Álvarez, F.J., Muñoz, J., Santaolaya, R., Collazos, C.A., Hurtado, J.A.: Verification and validation model for short serious game production. IEEE Lat. Am. Trans. **14**(4), 2007–2012 (2016)
47. Balaban, M., Maraee, A., Sturm, A., Jelnov, P.: A pattern-based approach for improving model quality. Softw. Syst. Model. **14**(4), 1527–1555 (2013). https://doi.org/10.1007/s10270-013-0390-0