Implementing Escape Room for Teaching Software System Architecture

AUTHORS: John Antony, Amal John, Temitope Adigun, Deepak Eldo, Majid Alsubaie and Richa Aswat*hy*

Abstract- *This report investigates the use of escape rooms as a new teaching method to help students learn software system architecture in college. Typical teaching methods have been quite functional however, lack the imaginative and interactive approach which escape rooms offer. Escape rooms can be used as a teaching tool in this Light object of science (i.e. software system architecture) as will be demonstrated by the analysis carried out below in the background, methods, results, and implications sections.*

Keywords—Escape rooms, educational interventions, interdisciplinary collaboration, empirical studies, professional development, knowledge sharing, software systems, systems architecture, teaching software systems, methods of teaching systems architecture**.**

INTRODUCTION

The architecture of software systems in our consideration when we approach computer science training, including. principal components, design building blocks, and patterns of a software system. Despite the fact that those old teaching methods, such as lecturing, presenting and discussing in a theoretical way, those methods in most cases can't fully catch students and lead to what depth of understanding that students are supposed to grasp on. Although these approaches assist in explaining and comprehending theoretical principles, they may not serve the students, who need to be able to work with hands on and problem solving, to navigate the real life challenges in software development.

The phenomenon represents an issue of conventional teaching methods, which mostly does not allow the learner being actively involved. Monotonous and high amount of information of lectures and presentations, less intense participation and critical thinking of students will fail to get it. On top of that, important things such as software systems architecture is much more than just taking in the information; it is interactive and moving hands to try something to know how it really works.

This gap shows that there is need for pedagogy innovation in teaching the system architecture software as the traditional teaching methods do not suit the learner's pace often. Therefore, the first part serves as the prologue for the ensuing investigation of the problems that are therein discussed and how better alternatives can be developed that make learning fun and meaningful.2
Through escape-room adventures – innovative, interactive experiences which have become a popular learning platform in multiple educational spaces such as universities. Escape rooms for the most part have been centered around getting teams to interweave and work together by solving puzzles, discovering clues, and finishing a specific endeavour in an allotted time frame. In order to foster teamwork, problem solving, decision-making, and communication – all of them are necessary for software system architecture – simulators are intended.

The section elaborates on software system architecture whose implementation using escape rooms promises to be a great revival tactic for high student engagement and mastery of concepts. Through making students facing these difficult simulate tasks that echo real problems which we normally get out of school, escape rooms bring one a unique chance to practice the theory they learned in context led to a deeper understanding and better practicing skills.

Additionally, escape rooms can form a versatile and interactive learning atmosphere that adjusts to the different styles and preferences of various students so that a group of

diverse students can learn in one room as well. This introduction hence paves the way for the subsequent investigation which is intended to provide an insight into the different methods of implementing and integrating escape room into the curriculum for software systems architecture teaching.

In the light of that, an introduction should play a role as a midst to come to the problem statement – the limitations of traditional methods of teaching software system architecture – and the solution – the involvement of escape rooms as an innovative educational method. It gives evident-base for the rest of the paper that gives background, reason, and objective for the study.

Research Approach and Contribution

The purpose of this research report is to explore the feasibility of implementing escape room methodologies for teaching software system architecture and other usages of escape room in higher education. This review culminates in a proposal to integrate escape room concepts into the teaching system software. Since the modern education has many traditional way of lectures, presentations and workshops in learning. Escape rooms have proven to be engaging and effective learning activities when conducted face to face. We have developed a unique research approach which helps in high level and dynamic learning.

2.0 Methods

The methodology for this systematic review involves a comprehensive search strategy, rigorous selection criteria, and systematic data extraction and synthesis.

2.1 Search Strategy

A systematic search is conducted using relevant keywords related to teaching software system architectures using various academic databases such as Google Scholar, IEEE Xplore, ACM Digital Library, and Scopus. Keywords such as "software architecture education," "teaching methods," and "software engineering curriculum" is employed to ensure a thorough search.

2.2 Selection Criteria

Papers and articles is included if they focus on teaching software system architectures in academic settings, present empirical evidence or insights on pedagogical approaches, and are published in peer-reviewed journals or conferences. Only English-language publications from the past decade has been considered to ensure relevance and currency.

2.3 Data Extraction and Synthesis

Relevant data, including teaching methods, learning outcomes, challenges, and best practices, has been systematically extracted from selected studies. A standardized data extraction form has been used to ensure consistency. The synthesized findings has analyzed thematically to identify patterns, trends, and discrepancies across the literature.

2.4 Quality Assessment

Each selected study has been critically appraised for methodological rigor and relevance to the review objectives. Quality assessment tools

such as the Joanna Briggs Institute Critical Appraisal Checklist for Systematic Reviews and Research Syntheses has been utilized to assess the quality of included studies.

2.5 Reporting

The systematic review findings has been reported following the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) guidelines. A transparent and reproducible approach has adopted to enhance the validity and reliability of the review process.

This systematic review aims to provide a comprehensive and evidence-based analysis of teaching software system architectures, informing pedagogical practices and guiding future research in this area by adhering to these methodological steps.

Research Questions

Based on the objectives described in the introduction part we have developed research questions. These questions form the basis for the literature review:

- **RQ1.** How can current research on software architecture and escape room implemented and optimized?
- **RQ2.** What is the current status of learning software architecture using escape room?
- **RQ3.** What can be learned from current research results in implementing escape room for teaching software architecture for future investigations?

A. Comprehensive Systematic Review on Teaching Software System Architectures

A.1 Introduction

Teaching software system architectures is crucial to computer science education, helping students comprehend the underlying ideas and design patterns that drive sophisticated software systems. As the need for qualified software engineers continues to climb, the ability to successfully educate system designs has become more vital (Félix García-Carballeira, 2020). This systematic study seeks to give a complete overview of the present status of teaching software system architectures, emphasizing the available methodologies and the issues educators encounter in this subject. The progress of technology has led to a profusion of software systems with varied degrees of complexity, ranging from basic applications to large-scale distributed systems (Thornton, 2013). Understanding how to design, evaluate, and implement these systems is vital for software engineers, making software system architectures a cornerstone of computer science education. However, teaching system architectures offer particular problems due to the subject's abstract nature and the quickly changing technology context (Sezgin Kaçar, 2013).

By integrating current research and assessing best practices in teaching software system architectures, this study tries to uncover practical ways for educators and curriculum creators. Furthermore, by emphasizing the problems and limits of present techniques, this review intends to promote future research and innovation in this field. Ultimately, the results of this research have the potential to influence pedagogical methods and enhance the quality of software engineering education, therefore contributing to the formation of a highly qualified workforce capable of fulfilling the needs of the digital era.

II.1 Context and Significance

In technology, the necessity of educating software system designs cannot be understated. As software systems grow more sophisticated and pervasive, the ability to develop and construct scalable, dependable, and maintainable software solutions is vital. Software system architectures give the conceptual foundation for understanding how software components interact and cooperate to meet system needs (Juan, 2012).

Moreover, the environment in which software system designs are taught is significant. With the fast evolution of technologies such as cloud computing, micro services, and the Internet of Things (IoT), educators need help keeping their curricula relevant and up-to-date (HalitOztekin, 2010). Students must not only master core ideas but also be prepared with the skills and information required to adapt to evolving technology and market trends.

Furthermore, the context goes beyond academics to business, where the need for software engineers competent in system architectural concepts is considerable. Employers desire people who can construct robust and scalable software systems, underlining the necessity of excellent instruction in this subject. This systematic study gives insights into teaching software system designs' present methods and problems. By evaluating current literature and combining best practices, this study attempts to enlighten educators, curriculum creators, and industry stakeholders on successful ways to impart critical system design knowledge to the next generation of software engineers.

II.2 Rationale for the Review

This systematic review is motivated by addressing the gaps and challenges in teaching software system architectures. Despite the critical importance of this subject in software engineering education, there needs to be more comprehensive analyses of current teaching practices and their effectiveness. By systematically reviewing the literature, this study aims to fill this gap and provide evidence-based insights that can inform pedagogical approaches and curriculum development in software system architectures. Ultimately, the goal is to enhance the quality of education and better prepare students for careers in software engineering.

II.3 Objectives of the Review

The primary objective of this systematic review is to comprehensively examine the existing literature on teaching software system architectures. Specifically, the review aims to:

1. Identify the range of teaching methods and approaches in software system architecture education.
2. Evaluate the effectiveness of different pedagogical strategies in facilitating student learning and comprehension.
3. Explore the challenges and barriers faced by educators in teaching software system architectures.
4. Synthesize best practices and recommendations for improving the quality of software architecture education.
5. Provide insights and recommendations for future research directions in this field.

Through these objectives, the review seeks to contribute to advancing pedagogical practices and enhancing students' preparation for software engineering careers.

II.4 Structure of the Review

This systematic review will follow a structured approach to synthesize relevant literature on teaching software system architectures. The review will begin with an introduction providing background information and the rationale for the study. Subsequently, it will include sections on context and significance, rationale for the review, objectives, and methodology. The main body of the review will comprise sections analyzing the findings of the selected studies, organized thematically based on key topics and research questions. Finally, the review will summarize key findings, implications for practice, and suggestions for future research in software system architecture education.

Included Studies

The systematic search across multiple databases yielded a selection of nine studies that met the predefined inclusion criteria:

1. Software System Architecture in Contemporary Computing

The article provides a comprehensive overview of the critical role that software system architecture plays in various domains, emphasizing its significance in ensuring the success, scalability, and adaptability of software projects. Audrito et al. (2022) highlight the importance of scalable architectural patterns, such as microservices, in managing the complexity of contemporary systems. Avhankar et al. (2022) stress the need for flexible and efficient architectures in wireless sensor networks to optimize resource utilization and support diverse localization approaches. Amirkhani and Barshooi (2022) discuss how well-designed architectures facilitate consensus among distributed agents in multi-agent systems, enhancing decision-making and system performance.

Moreover, the article explores the relevance of software architecture design in enabling specialized functionality, improving resource utilization, and ensuring data integration in IoT applications, as evidenced by the work of Sha et al. (2022) and Tianxing and Hong (2021). Boubiche et al. (2021) address the critical role of architectural design in enhancing cyber security in wireless sensor networks, while Zamry et al. (2021) present an energy-efficient clustering method to prolong the lifespan of WSNs. Additionally, the article discusses the importance of software system architecture in intelligent settings, such as smart homes or workplaces, where it facilitates the integration of heterogeneous devices and services. Shimoda et al. (2020) present RW-QAnswer, an assistance system that leverages semantic technology to enhance human-computer interaction and user experience in intelligent environments.

In summary, the article underscores the crucial role of software system architecture in ensuring software-intensive systems' performance, dependability, and security across diverse domains. Embracing scalable, adaptable, and secure architectural patterns is essential for addressing contemporary computing systems' increasing challenges and needs.

1. Teaching Systems Architecture

Teaching systems architecture encompasses theoretical foundations, practical application, and contextual understanding. Garlan and Shaw (1993) and Firesmith (2010) stress the importance of imparting theoretical knowledge, including architectural styles and design patterns. Cavalcante et al. (2023) emphasize the value of hands-on projects in fostering practical skills. Gacek et al. (1995) highlight the significance of understanding the broader system environment. Finally, Konersmann and Goedicke (2020) underscore the development of critical thinking and communication skills. The article provides a comprehensive framework for effective systems architecture education, integrating theoretical concepts, practical experience, and essential skills development.

3. Teaching Software System Architectural Models

The article presents a detailed review of significant architectural methods in software system design, as defined by renowned experts. Monolithic architecture, introduced by Garlan and Shaw (1993), provides simplicity but may need more scalability and agility in complex systems. Microservices design, addressed by Cavalcante et al. (2015), enables agility and scalability via separate service components but demands careful service coordination and operational complexity management. Service-oriented architecture (SOA), defined by Gacek et al. (1995), stresses loose coupling and reusability but confronts issues in interoperability and governance. Event-driven architecture, addressed by Cavalcante et al. (2016), offers real-time responsiveness but may create complications in event processing and administration. Fi resmith (2010) described that layered architecture promotes modularity and abstraction but may suffer from tight coupling and cross-cutting issues. Component-based design, emphasized by Guessi et al. (2015), supports reusability and flexibility but demands attention to component compatibility and composition concerns. The essay gives excellent insights into distinct architectural paradigms and their consequences for software system design.

4. A Collaborative Approach to Teaching Software Architecture

The article established that the collaborative approach to teaching software architecture is a promising methodology that fosters active engagement and knowledge exchange among students (Alexander, 1977). By encouraging collaboration, educators can create dynamic learning environments where students can leverage each other's strengths and insights to deepen their understanding of architectural principles and practices (Joop, 2019). Collaborative learning activities such as group projects, peer reviews, and problem-solving sessions promote teamwork, communication, and critical thinking skills essential for success in software engineering. Additionally, collaborative approaches enable students to explore diverse perspectives and approaches to software architecture, enhancing their ability to tackle complex design challenges (Arie Van Deursen, 2020). Adopting a collaborative approach to teaching software architecture can enrich the learning experience, empower students to become more effective problem solvers, and prepare them for successful careers.

5. Teaching Computer System Design and Architecture course — An experience

The experience of teaching a Computer System Design and Architecture course is a multi-faceted journey that encompasses various aspects of system architecture, performance, usability, and learning outcomes. Th is journal looked into fundamental concepts such as instruction set architecture, understanding of system components, and pipelining techniques, aiming to provide students with a comprehensive understanding of computer system design principles.

Drawing from the journal (Felder & Brent, 2003) and outcomes-based education studies (Jansen & Christie), educators can design curriculum reforms that align with industry standards and promote meaningful learning experiences. Corporating lifelong learning principles (Mocker & Spear) and assessment tools for outcome-based engineering courses (Deng et al.) further enhances the course's effectiveness in preparing students for the dynamic field of computer system design.

Overall, the teaching experience underscores the importance of teamwork, standards-based reforms, and a holistic approach to curriculum development in imparting essential computer system design and architecture knowledge and skills.

6. A Better Way to Teach Software Architecture

The article tackles a critical gap in undergraduate computer science and software engineering programs: the poor teaching of software architecture. Despite the complexity and variety of software architecture methods in business, bringing this knowledge into the classroom has proved problematic. The writers see this as a "chicken and egg" dilemma, where the absence of effective teaching materials limits architecture teaching, prolonging the deficit cycle. The study suggests a solution by providing criteria for teaching technical and non-technical software architecture elements and recommending relevant teaching techniques. The authors want to better educate students for employment as software architects by bridging the gap between industry practices and academics.

Drawing from a rich set of references, including works on microservice architecture (Christensen), software architecture principles (Bass et al.), architectural patterns (Buschmann et al.), and software design education (Cai et al., Capilla et al., Cervantes et al.), the paper offers a comprehensive framework for improving software architecture education.

In conclusion, "A Better Way to Teach Software Architecture" is a relevant and significant addition to the area, providing educators with the tools and direction required to strengthen the teaching of software architecture and better prepare students for success in the software engineering industry.

7. An Architecture-Based Software Reliability Modeling Tool and Its Support for Teaching

This work provides a unique architecture-based software dependability modeling tool for instructional purposes. Th e application aids the transmission of learning materials to students by allowing them to perform relative analyses on various architectural designs and calculate more precise dependability metrics. So, firmware dependability is a critical quality trait, and strengthening it early in the software life cycle minimizes testing and maintenance work later. The architecture-based method described in this application enables students to study various architectural styles and exercise several design possibilities. Equipped with a graphical user interface (GUI) for architecture-to-state modeling, the program contains four architectural styles and enables the incorporation of new styles by students.

References to fundamental publications in software dependability modeling and architecture, such as Cheung (1980), Shaw et al. (1995), Wang & Chen (1999, 2005), and so on, add credence to the paper's methodology. By reducing the learning curve, expanding comprehension of design influences on dependability, and raising student interest in quality features, this tool has tremendous promise for improving software architecture teaching. OvErall's "An Architecture-Based Software Reliability Modeling Tool and Its Support for Teaching" is an excellent addition to software engineering education.

8. Pr Priscilla – Proposal of System Architecture for Programming Learning and Teaching Environment

The study offers a revolutionary system architecture developed for teaching and learning programming, stressing interaction and current educational methodologies such as gamification, microlearning, and automated assessment. By splitting students' essential knowledge and abilities into discrete domains linked to complicated programming issues, the proposed approach seeks to deliver a systematic and thorough learning experience. Key modules and ideas underpinning the system's architecture are clarified, displaying the integration of interactive components, learning analytics, and customization capabilities. The system contains a graphical user interface (GUI) for architecture-to-state modeling, allowing students to visualize software architecture and experiment with various design choices.

References to relevant literature on programming education and

automated assessment enhance the paper's theoretical framework. Wo rks by Guzman et al. (2013), Robins et al. (2003), and Lee et al. (2011) give insights on interactivity in education, learning programming, and the link between perplexity and accomplishment in beginner programmers.

Furthermore, references to research on automated assessment tools and microlearning-based training applications underlinethe practical relevance and viability of the proposed approach. No table studies by Krusche and Seitz (2018), Rozkosny (2018), and Zufic and Jurcan (2015) add to the conversation on automated assessment, microlearning, and learning management systems. Generally, "Priscilla – Proposal of System Architecture for Programming Learning and Teaching Environment" proposes a potential foundation for increasing programming education via creative system architecture and pedagogical techniques.

9. Software Architecture for Integration of Institutional and Social Learning Environments

AlMegren and Yassin (2013) explore the challenges faced by learners in Saudi Arabia regarding learning object repositories (LORs) in Software Architecture. They emphasize learners' difficulties in efficiently accessing and using these libraries. The authors stress the need for tailored learning resources, which are often divided into smaller components called learning objects (LOs), to meet the specific requirements of each student. These results align with Chun's (2004) emphasis on the significance of flexible teaching and learning methods, especially in the context of e-learning platforms.

Implementing service-oriented architecture (SOA) in educational systems, as Blinco et al. (2009) suggested, is essential for attaining flexibility and enhanced performance. Ca squero et al. (2010) provide an integrated eLearning 2.0 architecture, emphasizing the need for smooth communication across various components of the e-learning environment. Co nde et al. (2014) provide a service-based paradigm to integrate personal and institutional learning settings to improve the learning experience. Do Dero and Ghiglione (2008) highlight the significance of organized lesson sequences and interactive learning environments in the model of activities for learning content generation? Ju NCO's methodology illustrates how combining instructional and problem-based learning activities offers students a variety of learning possibilities. The references emphasize the significance of tackling the complex issues in e-learning settings, especially in promoting individualized and interactive learning experiences for learners.

Key Findings and Themes

In evaluating the works linked to teaching software system architectures, numerous significant discoveries and themes emerge, offering light on different elements of this complex area. These results and topics span instructional techniques, technology breakthroughs, integration issues, and the shifting role of educators. Let us look into each of these areas:

A. Pedagogical Approaches

1. Problem-Based Learning (PBL): Many papers show the usefulness of problem-based learning in teaching software system designs. As stressed by Chun (2004), the technique emphasizes active learning by involving students in addressing real-world issues. Similarly, the approach described by Junco et al. stresses the relevance of problem-solving activities inside a social learning environment (SLE), enabling students to participate and share ideas.

2. Instructional Design: The papers stress the necessity of organized lesson sequences and instructional design ideas. Do Dero and Ghiglione (2008) examine the design of learning objects (LOs) and the development of interactive learning environments, ensuring that instructional materials appeal to varied learning styles?

B. Technological Advancements

1. Service-Oriented Architecture (SOA): Blinco et al. (2009) argue for the implementation of SOA in educational systems, noting its advantages in attaining flexibility and increased performance. The incorporation of SOA provides smooth communication between diverse components of e-learning environments, as Casquero et al. (2010) exemplified.

2. Web Services Integration: The papers focus on integrating web services to promote interoperability across learning management systems (LMS), external tools, and social learning environments. Do Dero and Ghiglione (2008) consider using REST-based web services for accessing learning design services? In contrast, Conde et al. (2014) suggest a service-based architecture for merging personal and institutional learning environments.

C. Integration Challenges

1. Complexity of Learning Environments: Henderson et al. (2006) emphasize students' perspectives of the psycho-social clinical learning environment, highlighting the need for seamless integration between personal and institutional learning environments. Gu duric et al. (2013) explore the integration of QDITA and LAMS, underlining the complexities of integrating multiple learning tools and platforms.

2. Legacy Systems Integration: The papers highlight the issues provided by legacy systems and the necessity for compatibility and scalability in integrating multiple educational technologies. Co-Peto (2012) analyzes the OSID-osid package description, highlighting the necessity of standardized interfaces for interoperability.

D. Evolving Role of Educators

1. Facilitators of Learning: The articles illustrate the increasing role of educators as facilitators of learning in technology-enabled contexts. Am es (1992) analyzes the aims, structures, and student motivation in classrooms, highlighting the role of educators in creating student involvement and motivation.

2. Personalization of Learning: Wilson et al. (2012) study how educators might employ web-deployed widgets and collaborative capabilities to tailor student learning experiences. They advocate for solutions that integrate personal and institutional learning settings, allowing educators to personalize learning experiences to student requirements.

The publications evaluated give valuable insights into teaching software system designs, showing the relevance of pedagogical innovation, technological integration, and the growing role of educators in defining technology-enabled learning environments. These results and themes underline the varied nature of teaching software system architectures and the necessity for collaborative efforts to solve the problems and possibilities in this area.

B. Comprehensive Systematic Review on Benefits of Escape Room in Higher Education

INTRODUCTION

Escape Rooms have emerged as innovative educational tools that engage learners in immersive and interactive experiences, fostering collaboration, problem-solving, and critical thinking skills (Anderson, 2021). Originating from the entertainment industry, Escape Rooms have gained traction in various educational settings, including healthcare, nursing, pharmacy, and general education. This introduction provides a comprehensive overview of Escape Rooms in education, drawing upon a wide range of literature to explore their evolution, applications, and educational impact.

The utilization of Escape Rooms in education has been documented across numerous disciplines, reflecting their versatility and effectiveness as pedagogical tools. Research by Nicholson (2015) highlighted the potential of Escape Rooms to enhance learning experiences and promote active engagement among participants. Subsequent studies have further examined the educational benefits of Escape Rooms, including their ability to improve teamwork (Wu et al., 2018; Fitzpatrick et al., 2021), leadership skills (Valdes et al., 2021), and communication (Fitzpatrick et al., 2021). In healthcare education, Escape Rooms have been increasingly integrated into curricula to simulate clinical scenarios and reinforce theoretical knowledge. For example, Akatsu et al. (2022) implemented Escape Rooms in medical education, demonstrating their efficacy in teaching clinical skills and promoting active learning. Similarly, Anderson et al. (2021) discussed the potential of Escape Rooms to enhance healthcare simulation experiences, emphasizing their role in facilitating hands-on learning and teamwork.

Moreover, Escape Rooms have been utilized to teach specific subject areas, such as pharmacology (Hermanns et al., 2017), toxicology (Korenoski et al., 2021), and emergency medicine (Cantwell et al., 2022). These studies have underscored the effectiveness of Escape Rooms in reinforcing key concepts, improving retention rates, and fostering critical thinking abilities among learners.

In addition to their traditional format, digital and virtual Escape Rooms have emerged as innovative alternatives, offering greater accessibility and flexibility in educational settings (Kubin, 2020; Koo and Li, 2016).

These digital platforms enable educators to create immersive learning experiences that transcend physical boundaries, accommodating diverse learning preferences and enhancing student engagement. Generally, the body of literature on Escape Rooms in education reflects their growing significance as effective pedagogical tools. This systematic review aims to synthesize existing research, providing insights into the educational benefits, design considerations, and future directions of Escape Rooms in various educational contexts. Through a comprehensive analysis of the literature, this review seeks to inform educators, policymakers, and researchers about the potential applications and implications of Escape Rooms for teaching and learning.

II.1 Context and Significance

Educational escape rooms require time for design, create, conduct and evaluate. Therefore high return on investment is expected for teaching and learning. Many studies shows positive learning gains are acquired using escape room method. The impact on learning was consistent across diverse fields and educational levels. Active and collaborative learning is happening in this technique.

II.2 Rationale for the Review

We are incorporating the idea of implementing the teaching of software architecture using escape. There are a lot of benefits for escape room in higher education. Escape room sessions help in developing various skills. Communication skills, leadership skills, social skills and problem solving skills are enhanced. According to WHO, psychosocial skills are ability to respond effectively to the demands and challenges in daily life. These skills include self-awareness, empathy, communication and problem solving. This is developed by escape room in higher education. Escape room in higher education helps in encouraging group cohesion.

Student and teacher relationship in improved in escape room technique. It develops a bond outside the classroom also. Personal level strengths and weakness can be understood by this mechanism. It also develops a taste for learning and enthusiasm in gaining knowledge. Real world situations can be solved by practicing this sort of educational methods. Also helps in thinking outside the box. Challenges are viewed as opportunities to learn and grow.

II.3 Objectives of the Review

This review is based on the objective of sorting out benefits of escape room in higher education. This helps in implementing software architecture teaching using escape room. Each and every benefits of escape room can be incorporated to learn the software architecture. An escape room experience is an excellent tool for engaging students, developing soft skills like teamwork, leadership, communication and putting hard skills into practice. These are more important with the rise of generative AI tools.

While everyone wishes to leave traditional class room but escape room makes everyone engaging. Software architecture learning is made by escape room mechanism. Escape rooms foster critical thinking abilities, encourage collaborative learning and confidence. This is a dynamic platform for applying theoretical information into practical.

II.4 Structure of the Review

This systematic review will follow a structured approach to synthesize relevant literature on benefit of escape room in higher education. The review will begin with an introduction providing background information and the rationale for the study. Subsequently, it will include sections on context and significance, rationale for the review, objectives, and methodology. The main body of the review will comprise sections analyzing the findings of the selected studies, organized thematically based on key topics and research questions. Finally, the review will summarize key findings, implications for practice, and suggestions for future research in escape room for higher education..

An Architecture-Based Software Reliability Modeling Tool and Its Support for Teaching

This work provides a unique architecture-based software dependability modeling tool for instructional purposes. Th e application aids the transmission of learning materials to students by allowing them to perform relative analyses on various architectural designs and calculate more precise dependability metrics. So, firmware dependability is a critical quality trait, and strengthening it early in the software life cycle minimizes testing and maintenance work later. The architecture-based method described in this application enables students to study various architectural styles and exercise several design possibilities. Equipped with a graphical user interface (GUI) for architecture-to-state modeling, the program contains four architectural styles and enables the incorporation of new styles by students.

References to fundamental publications in software dependability modeling and architecture, such as Cheung (1980), Shaw et al. (1995), Wang & Chen (1999, 2005), and so on, add credence to the paper's methodology. By reducing the learning curve, expanding comprehension of design influences on dependability, and raising student interest in quality features, this tool has tremendous promise for improving software architecture teaching. OvErall's "An Architecture-Based Software Reliability Modeling Tool and Its Support for Teaching" is an excellent addition to software engineering education.

Addressing the past issues with Escape room

Escape rooms have been around for quite a while providing participants with the opportunity to put their problem-solving skills to test and unravel riddles within a fixed time limit so that they can claim victory and escape definitively. Escape rooms do not only involve entertainment but innovation is also a potential tool; for instance, they can act as educational learning tools for complicated subjects like architecture of software systems(Bilbao-Quintana et al., 2021).Through blending up of the mechanism of immersive learning, problem-solving, and the teamwork, escape-rooms provide a lively lab where students can familiarize themselves with various abstract concepts and hence bring them to real-world life. By far the strongest point we can find in using escape rooms for training software system architecture easily is the possibility of transferring the most practical circumstances to a classroom. Software architecture is an industry comprising of a number of aspects represented by design, development, and implementation. Lectures that are based on traditional practices or that cover the theoretical side in details are sometimes not able to capture this area as much as it deserves. While doing escape room, one can tellingly feel these problems that architects in their careers frequently come by. The representatives must cooperate, think hard and apply architectural rules, and thus through the set of practical problems they get acquainted with finding the solutions(Fraguas-Sánchez et al., 2022).

Collaborative Approach for Escape Room Model adopting the approach of collaboration in developing exit rooms is indispensable when it comes to delivering a multi-dimensional experience that would keep participants occupied with innate abilities and passions. Bringing together people with different roles such as faculty members, students and professionals from all divisions of the escape room can get a variety of viewpoints and expertise.

Collaboration give rise to creativity and innovation because people with diverse ideas and visions contribute to the entire feeding of ideas. Furthermore, such cooperation is accompanied by the use of multiple approaches and issues that eventually reflect the escape room design itself, contributing to a better and a more meaningful game play. Apart from that, a joint effort endorses an atmosphere of participation among the participants since they are involved in the full process of refining the escape room idea into a reality. In the long run, teamwork as a guideline that allows us to make realistic and interactive situations where players can show their independent thinking skills, teamwork and learning strategies.

Escape Room Design – An Experience

Creating an escape room is a multifaceted task involving the development of a solid background before your start being creative and focused on detail. It starts with brainstorming to incorporate the stimuli that are conceived to be applicable to the story, puzzles or other aspects of the experience. The designers together come up with puzzles and challenges containing elements that keep the players interested and the objectives incorporated, so that the participants have total fun as well as an intellectual challenge (Veldkamp et al., 2020). The set design and decoration that show off in creating an immersive environment can be called the heartbeat of the game, as every detail is carefully chosen by the designers to exemplify the set for the sake of the escape room. Designers conduct these iterations by trying to figure out the puzzle concepts, create challenges, and test them to determine if they satisfy the complexity and engagement level of the players. Interactions and evaluations are the pointers where the designers will depend on the participants' input to produce a rewarding and easily recalled educational experience (Dugnol-Menéndez et al., 2021).

Benefits of Escape room

Let us also not forget that escape rooms are great for experiential learning, which enables students to experience different concepts rather

than going through the process of abstract memorization. The facilitator should move away from dry imparting of information, but towards the participants being active in the learning process, ie. solving problems, integrating information, and providing solutions. This is reminiscent of the way software architecture involves thinking. Through such an interactive learning process students are looking deeper into the meaning and essence of architectural decisions and, simultaneously, such having a hands-on approach makes them more prone to successful learning.

Moreover, escape rooms enhance these qualities and especially collaboration, a trait of good software developers. To this end, in a regular escape room experience, players have to learn to communicate well, distribute tasks among each other and utilize whatever talent they might have available for their overall success. Just as in software-driven tasks such projects, architects have to work together with developers, stakeholders, and other staff in the team to bring these projects to a successful completion.

As an add on, Escape rooms which fosters creativity and innovative skilled problem solving also takes place. Unlike software architecture the problems that architecture has to deal at times requires out-of-the-box thinking for an optimum solution to these problems. While in the escape room, the participants are constantly advised to think outside the boxed, embrace different manners, and evaluate their designs per the feedback they receive-a behavior that is characteristic of the code innovation of architecture. Stimulating an environment that celebrates imagination and mental flight, the escape rooms allow students to experiment with novel approaches, and thus the required flexibility to navigate the rapidly-changing software industries easily is developed.

Moreover, objections such as including the learning objective on software architecture can be addressed by including the content in escape rooms. Take it as for instance, if a designer wants student`s be acquainted with architectural patterns, design principles or system scalability they can create puzzles and problems that fit course content and after that repeated key concepts within engaging and memorable context. Also, similar to debriefing after the theater experience students are given a chance to after theater visit and to articulate the learning gained through the escape room activity.

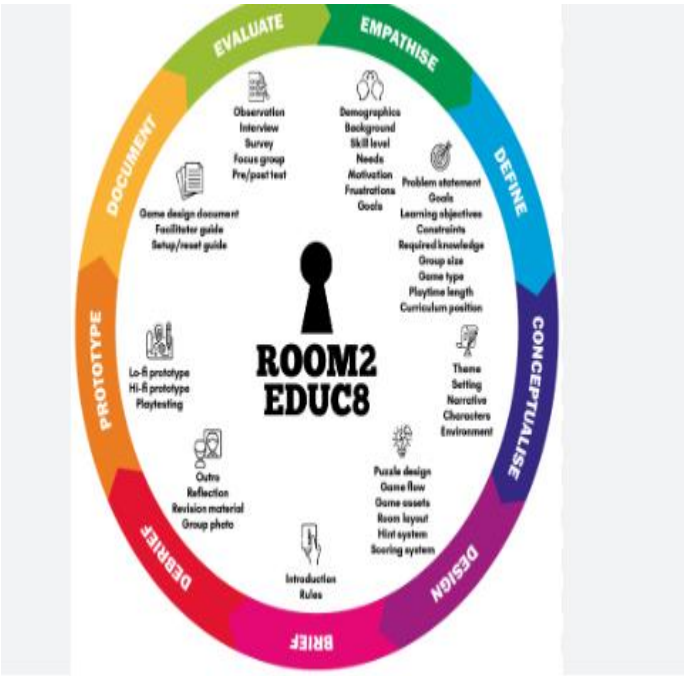

Figure 1 Framework for Escape room (Fotaris & Mastoras, 2022)

In summary, escape rooms are a great alternative for teaching software system architecture because of the way the experience engages rather than merely informs the students. It challenges students to solve multilayered tasks and work together as a team to complete important objectives. With an environment in between theoretical and practical that resembles domain testing applied architectural principles, what students learn in the context of escape rooms is the step ahead in the complexities of real-world software. Educators are continuously on a quest to find innovative teaching approaches, which in the context of this case study about software architecture, the use of escape rooms will certainly bring about a significant transformation in teaching and learning.

3.2 Educational Impacts of Escape Rooms in Enhancing Learning Outcomes

Escape rooms have demonstrated various educational impacts in enhancing learning outcomes across diverse disciplines. Research findings suggest that the use of escape rooms fosters active engagement, improves critical thinking skills, enhances content retention, promotes teamwork, and increases motivation among participants (Anguas-Gracia et al., 2021; Friedrich et al., 2019).

Active engagement is a key benefit of escape rooms, as they provide hands-on, experiential learning experiences that encourage participants to actively participate in solving problems and completing tasks (Farus-Brown et al., 2020). This active engagement facilitates deeper learning and understanding of the subject matter, as students are actively involved in applying theoretical knowledge to real-world scenarios (Akatsu et al., 2022).

Escape rooms also promote critical thinking skills by requiring participants to analyze information, identify patterns, make connections, and generate creative solutions to complex problems

within a limited time frame (Antón-Solanas et al., 2022). By engaging in critical thinking processes, students develop higher-order thinking skills that are essential for academic success and professional competency (Clauson et al., 2019).

Furthermore, escape rooms facilitate content retention by providing memorable and immersive learning experiences that appeal to multiple senses (Aubeux et al., 2020). The thematic immersion and interactive nature of escape rooms create lasting impressions that enhance information retention and recall among participants (Cerenzio and Ocheretyaner, 2021).

Teamwork is another significant educational outcome of escape room experiences, as participants must collaborate, communicate effectively, delegate tasks, and leverage each other's strengths to solve puzzles and escape successfully (Fusco et al., 2021). This collaborative learning environment mirrors real-world professional settings where teamwork and interdisciplinary collaboration are essential for success (Gordon et al., 2019).

Moreover, escape rooms increase motivation and engagement among participants by providing a fun and challenging learning experience that encourages active participation and rewards successful problem-solving efforts (Guckian et al., 2020). The intrinsic motivation generated by escape room activities can lead to enhanced learning outcomes and a positive attitude toward learning (Hursman et al., 2022).

Therefore, escape rooms have a profound impact on enhancing learning outcomes by promoting active engagement, critical thinking skills, content retention, teamwork, and motivation among participants, as supported by evidence from research studies across various educational settings.

3.3 Student Engagement

Student engagement is a crucial aspect of educational effectiveness, and escape rooms have been shown to significantly enhance engagement levels among participants. These immersive learning experiences captivate students' interest and motivation by providing them with challenging, interactive, and enjoyable activities (Anderson et al., 2021).

Escape rooms offer a departure from traditional passive learning methods by placing students in the center of the learning experience, where they must actively participate in problem-solving and decision-making tasks (Barrickman et al., 2022). This active engagement fosters a sense of ownership and responsibility for learning outcomes, leading to increased motivation and investment in the educational process (Cunha et al., 2023).

Furthermore, the collaborative nature of escape rooms encourages peer interaction and communication, promoting social engagement and teamwork skills (Dittman et al., 2021). Students work together to solve puzzles, share ideas, and support each other, leading to a sense of camaraderie and community within the learning environment (Eukel and Morrell, 2021).

Escape rooms also leverage elements of gamification, such as competition, time pressure, and rewards, to incentivize student participation and enhance engagement (Faysal et al., 2022). The game-like structure of escape rooms appeals to students' intrinsic motivation and desire for challenge and achievement, driving them to actively seek solutions and overcome obstacles (Ferns et al., 2022).

Moreover, the thematic immersion and storytelling aspects of escape rooms create an engaging and memorable learning experience that resonates with students (Ferrer-Sargues et al., 2021). By immersing participants in a fictional narrative or scenario, escape rooms stimulate their curiosity, imagination, and emotional investment, leading to deeper levels of engagement and involvement (Foltz-Ramos et al., 2021).

Additionally, escape rooms accommodate diverse learning styles and preferences, allowing students to engage with course material in ways that are personally meaningful and relevant to them (Frenzel et al., 2020). Whether through visual, auditory, kinesthetic, or interpersonal modes of learning, escape rooms offer multiple entry points for student engagement and participation.

3.4 Interprofessional Education (IPE)

Escape rooms offer a unique platform for promoting collaboration and communication among students from diverse professions and sectors, making them valuable tools for interprofessional education (IPE) initiatives across various disciplines.

Escape rooms have been extensively utilized in healthcare to facilitate teamwork and communication among healthcare professionals, including physicians, nurses, pharmacists, and physical therapists (Fusco et al., 2021). These experiences encourage participants to leverage their respective expertise and perspectives to solve complex challenges, fostering a deeper understanding of each other's roles and responsibilities within the healthcare team (Hebert, 2023).

Beyond healthcare, escape rooms have also found applications in engineering, architecture, agriculture, and other fields where interdisciplinary collaboration is essential for problem-solving and innovation. For example, in engineering education, escape rooms provide opportunities for students from different engineering disciplines to collaborate on design projects, simulations, or problem-based scenarios (Kinio et al., 2019). By working together to overcome engineering challenges within a time-constrained environment, students develop essential teamwork, communication, and problem-solving skills that are applicable across various engineering domains.

Similarly, in architecture and design, escape rooms can serve as immersive learning environments where students collaborate to address architectural challenges, spatial constraints, and design considerations (Kubin, 2020). By engaging in hands-on design tasks and creative problem-solving activities, students gain practical experience in interdisciplinary teamwork and learn to integrate diverse perspectives into their design processes.

In agriculture, escape rooms can facilitate collaboration among agronomists, environmental scientists, economists, and policymakers to

address complex issues related to food security, sustainability, and agricultural development (Farus-Brown et al., 2020). Through interactive simulations, case studies, or role-playing exercises, students explore interdisciplinary approaches to solving real-world agricultural problems, such as crop management, soil conservation, and rural development.

Lastly, several studies reviewed evaluated the impact of escape rooms on learning outcomes and found positive results across various educational settings. For example, Farus-Brown et al. (2020) explored the use of escape rooms in undergraduate nursing education and reported improvements in student engagement, critical thinking, and knowledge retention. Similarly, Frenzel et al. (2020) conducted a study on the effects of escape rooms on pharmacy students' knowledge and perceptions, highlighting gains in content mastery and confidence.

3.5 Game Design and Implementation

Game design and implementation in educational escape rooms involve careful consideration of various elements to create engaging and effective learning experiences. Drawing from the reviewed literature, we can analyze several key aspects of escape room design:

3.5.1 Scenario Development

Effective escape rooms begin with a well-crafted scenario that sets the stage for the immersive experience. Authors such as Akatsu et al. (2022) emphasize the importance of integrating educational content seamlessly into the narrative to ensure relevance and engagement. For example, in healthcare-focused escape rooms, scenarios may simulate clinical settings or patient encounters, providing learners with realistic challenges that require critical thinking and problem-solving skills (Barrickman et al., 2022).

3.5.2 Puzzle Design

Puzzles serve as the core gameplay mechanic in escape rooms, requiring participants to solve clues, decipher codes, and unlock hidden mechanisms to progress through the game. Aubeux et al. (2020) highlight the role of puzzle diversity in maintaining player engagement and catering to different learning styles. Puzzles can range from logic puzzles and riddles to hands-on tasks and interactive challenges, allowing for a multifaceted learning experience (Caldas et al., 2019).

3.5.3 Facilitation Techniques

Facilitators play a crucial role in guiding participants through the escape room experience and providing support when needed. Hermanns et al. (2017) emphasize the importance of trained facilitators who can balance assisting with allowing participants to work through challenges independently. Facilitators may offer hints or clues to keep the game progressing smoothly while ensuring that learners remain actively engaged and immersed in the experience.

3.5.4 Technology Integration

Some escape rooms incorporate technology, such as digital platforms or augmented reality, to enhance immersion and interactivity. Antón-Solanas et al. (2022) discuss the use of digital escape rooms in nursing education, where technology enables the creation of dynamic scenarios and interactive puzzles that can be accessed remotely. By leveraging technology, educators can expand the scope of escape room experiences and reach a wider audience of learners.

3.5.5 Assessment and Feedback

Educational escape rooms can also serve as assessment tools to evaluate learners' knowledge, skills, and teamwork abilities. Clauson et al. (2019) describe using escape rooms to assess students' readiness for advanced practice experiences in pharmacy education, highlighting the potential for performance-based assessments within the escape room context. Additionally, providing feedback during and after the experience allows learners to reflect on their performance and identify areas for improvement (Eukel and Morrell, 2021).

3.6 Engagement and Motivation

Escape rooms are known for their immersive and interactive nature, which can enhance student engagement and motivation. Gómez-Urquiza et al. (2022) conducted a qualitative observational study on nursing students' experience with an escape room for training clinical skills, highlighting the high levels of engagement and enthusiasm among participants. By providing hands-on learning experiences in a fun and challenging environment, escape rooms can stimulate intrinsic motivation and foster a deeper understanding of course content.

In addition, educational escape rooms offer flexibility in terms of design and implementation, allowing educators to tailor the experience to specific learning objectives and student needs. Dittman et al. (2021) proposed a flexible and customizable virtual escape room approach for inter professional learners, emphasizing the importance of adaptability in catering to diverse learner preferences and abilities. This adaptability enables educators to create personalized learning experiences that accommodate different learning styles and preferences.

II. RESULT

A literature review pinpoints that lecture, presentation, and case studies are the most popular education methods for software system architecture classes. These two approaches could prove to be useful in providing the abstract concepts, but at the same time they may fall short in connecting students with the material and may lead to them not fully comprehending its content. Besides that, the failing of these traditional approaches in satisfying the diverse set of needs in the modern learning environment is also slowly being realized by many.

It followed on the literature review that showed the increasing use of escape-room-based learning platform in a higher education environment as a way to improve student engagement, teamwork, problem solving, and critical thinking skills. Escape rooms foster a hands-on and realistic learning modality in which children work in teams to overcome obstacles in a hypothetical scenario that calls for knowledge acquired in the classroom.

Escape room are the modern form of education. They are used to strengthen teams. Teams are made stronger by escape room methods. In learning concepts easily one can try escape room. In an educational setting escape room can have multiple formats. We can add a lot of activities within the classroom for high end and dynamic learning.

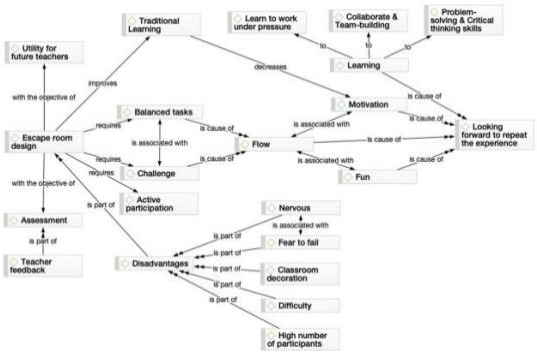

Figure1 Network diagram showing students experience regarding the educational escape room

Surveys' results, teachers, the interface skills, and experts' opinion series have helped us to make resolutions on teaching methods and phenomenon of escape rooms in software system architecture. The survey's results showed that despite the fact conventional methods are still popular; increase of interest in alternative methods, particularly escape rooms to maximize student engagement and improve result has been noticed.

Across students are excited about the use of escape rooms as an alternative and interesting way of studying software system, underlying the positive role of escape rooms, such as increased motivation, teamwork and problem-solving skills. Educators acknowledged that built-in escape rooms can be used as interactive and immersion teaching aids and make the students to take greater participation.

Educators, students, and industry professionals' statements in the interviews made it clearer to what the benefits or disadvantages of using escape rooms as a digital tool to teach Software System Architecture were Teachers underlined the necessity of tasks, created in escape rooms that correlate with learning targets and curricular tasks as well as allowing the access and inclusion for all students.

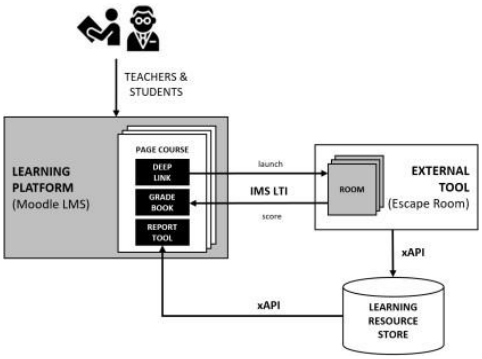

Figure 2 Integration of learning platforms with external tools

Students contributed to a discussion about the pros of escape rooms, as a fun idea of this team play gave them the feeling of a group-achievement and personal satisfaction when they were resolving the puzzles and completing the tasks. The experts from the industries reflected upon the fact that escape rooms would not only make it easy for participants to be equipped with practical skills but they would also develop the collaboration capabilities which are greatly needed in the work places for success.

Thematically designed observational investigations in classrooms helped to conduct a comprehensive study about the use of escape rooms in teaching software system architecture. Scientists picked the most active mission-driven students while the students were solving the problems, acted as a team and improved their problem-solving skills during the escape room activities. Students displayed remarkable interest for the engaging learning experiences provided by escape rooms which made them Sample Sentence: By utilizing a variety of sources, including interviews, data analysis, and surveys, we can gain valuable insights into the impact of gender inequality on women's participation in politics.

In a summary, educators have been still prevailing with the traditional methods of teaching, however, there is raising the interest on non-traditional methods such as escape rooms for the better engagement and learning outcomes. What makes escape rooms unique is that engage people by providing a host of learning experiences that involve better motivation, teamwork, and problem-solving disciplines and hence, are a promising learning tool for the subject of software system architecture.

III. DISCUSSSION

It will be the first step in the discussion process by the analysis of the results and information collected from the literature review, surveys, interviews and observational studies. It brings out the underlying themes and structure which are consistent in various data sources, highlighting both the pros and cons of conventional methods while still pointing out the numerous benefits of escape room teaching against the background of system architecture.

The expanding significance of research results on the effectiveness of traditional pedagogy calls for innovative methods aimed at overcoming the underlying limitations and increasing the level of students' engagement and learning results. Escape room education contributes to the idea that it helps students to solve problems and think critically, enhancing their knowledge through exploration and collaboration in interactive of immersive forms.

Significance of Escape Rooms in Teaching Software System Architecture

The debate use of escape rooms in teaching software systems architecture is grounded on the research and this is reveled through the cutting-edge insights that are garnered from the case study. It emphasizes the advantages and merits of escape rooms experiences such as excitement, engagement, and technical skill development which are key factors for adequately equipping students to face real life challenges in rapidly evolving and dynamic technology realm.

The practice areas of escape rooms give students a hands-on experience to visualize what they have learned and make their theoretical understanding more profound. Via their implementation of the real world tricks and team problem solving, escape rooms fill up this gap between theory and practice which are very essential for students to sharpen their skills for future jobs with uncertainties and complexities.

The talk places a strong focus on the possible advantages of using an escape room to teach how the system architecture of software is structured. The talk, however, also deliberates on the challenges and the considerations necessary during the workshop's implementation. For example, these are the logistical challenges that can be countered as space-time, and resources required by such courseware. It is equally important to ensure that the content of these materials is well integrated with learning content and objectives for the sake of aligning the course with learning outcomes.

Another aspect highlighted in our conversation is the indispensability of ease of use and inclusiveness while creating escape room assignments, to ensure that all the pupils have an equal stab at the task and finish it in a successful way. Strategies for dealing with these problems can be collective partners of teachers, instructional designer and game developers to create and use which are effective for students with different needs.

The discussion culminates with the development of a research implication and future suggestions forting escape rooms into the software system architecture curriculum. It indicates the spotlighted parts where improvisation is needed such as the inventing of new themes for reflective exercises, giving of inclusive study methodologies excluding hearers from participating and undermining the research on lasting consequences of such pedagogical strategies on success and perseverance of college students.

Moreover, the talk focuses on the workers aspects: firstly, cross-departmental cooperation and knowledge exchange among educators, researchers and practitioners so they can move forward the area of game design and implementation. Students are actually learning because they enjoy the thought of being able to escape or navigate through a series of rooms filled with clues, all of which pertain to software system architecture. Therefore, by incorporating escape rooms into the classroom setting, educators can improve the level of learning by ensuring that the students gain a deeper understanding of the subject using a more engaging approach to teaching.

In conclusion, the discussion encapsulates the in-depth analysis, and implications for implementation as well as some future directions for using escape rooms in the configuration of software system architecture in traditional class-room education, which has the ability to transform methods of teaching and learning.

II. LIMITATION

Resource Constraints: Seeking flexibility and affordability escape rooms force organizers to spend additional resources like space, time, materials and expertise. The opportunity to employ these resources could be lacking in some cases, which might very likely lead to the hindrance of educational activity escape rooms.

Logistical Challenges: The coordination with the other services rather than to fit escape room sessions within existing curricula and schedules is a herculean task. While the availability of time slots for students, facilitation of bulk of the participants and provision of necessary logistics (technical help and room setup) may also lead to some logistical constraints.

Accessibility and Inclusivity: Building the escape room enterprise with universal access in mind-for students with speech, language or other unique abilities- is a tricky balancing act between modifying and adjusting design aspects.

Assessment and Evaluation: Ensuring assessment validity and reliability to gauge student learning amounts from escape room experience is multiple of complications. Often the conventional room assessing system is not oriented to evaluate the skills and

competencies gained during the escape room activities. Therefore, the emergence of intelligent evaluation techniques is a must.

Pedagogical Integration: Immersing players and coaches in the escape rooms also calls for educators to have eighth- grade science curriculum and method policy horizontally connected. The perfect escape room activities as complements to traditional teaching methods and meeting a specific learning outcome together with teachers could be challenged by the fact that the process of planning and collaborating among educators is usually very demanding.

Student Engagement and Motivation: The immersion nature of escape rooms has the power to increase the level of motivation and involvement of the students. However, it is important to know that effectiveness of escape rooms may vary using the factors such as interest, studying type and prior knowledge of the learners. To keep the goal of the escape rooms interesting and energetic, continuous support and proficient guidance of the teacher might be required.

The limitation of using Escape Room-based activities in teaching a class need to be remedied through painstaking, interdepartmental ,and transformational processes so as to make them fit the requisites needs and limitations for the classroom settings. Through accepting and overcoming these constraints, educators will be able to tune in the mechanisms of a "game" of escape room for a maximum effect of student learning software system architecture.

IV. IMPLICATIONS

The implications for researchers arising from the exploration of implementing escape rooms for teaching software system architecture are multifaceted and hold significance for advancing both educational practice and academic inquiry are:

Further Research Opportunities: Researchers, in particular, have engaged themselves in excavating various complex aspects of escape room integration in educational setting. This would require examining the strengths and weaknesses of recent findings, devising an appropriate assessment strategy, checking the success of the integration over the long term, and determining the best practices of escape rooms for curriculum.

Interdisciplinary Collaboration: Although escape rooms belongs to no specific correct discipline, it fuses different areas, creating a framework for research collaboration between educators, computer scientists, psychologists and the game design community. Teamwork can take a variety of forms which include information sharing and brainstorming; all of these factors can undoubtedly lead to the creation and harnessing fainter disciplinary approach. Innovative Pedagogical Approaches: Academic formers can basically be responsible for the innovation and incorporation of new teaching methods by studying the endless possibilities of escape rooms and other robust learning environments. Through investigating new teaching methods researchers can provide advisors for teaching and development practitioners whose activities aimed at the cultural development of instruction and learning at the university.

Dissemination and Knowledge Sharing: Researchers have the duty of not only making the discoveries known to the wider academic community, but also to disseminating the outcomes. This can be undertaken by furnishing research articles, attending conferences, organizing workshops and sharing valuable documents online, and in repositories.
Ethical Considerations and Equity: Researchers when it comes to ethical issues, should also review all implementation related issue, privacy and consent.

In general, the researchers possess a unique chance to move forward educational practice and science through research of a new educational escape rooms worldwide prevalent technique for teaching software system architecture. Through interdisciplinary collaboration, experimental work, professional development provision and ethical contemplation, researchers will be at the fore front of the coming revolution which will facilitate the transformation of higher education teaching and learning.

V. REFERENCES

1. *AlMegren, A., & Yassin, S. Z. (2013). I am learning object repositories in e-learning: Challenges for learners in Saudi Arabia—European Journal of Open, Distance, and E-Learning, 16(1), 115.*
2. *Ames, C. (1992). Classrooms: Goals, structures, and student motivation. Journal of Educational Psychology, 84(3), 261.3.*
3. *Blinco, K., Grisby, T., Laird, A., O'Neill, O., Srikanth, V., & Smythe, C. (2009). Adopting service-oriented architecture (SOA) for enterprise systems in education: Recommended practices. Retrieved April 25, 2015.*
4. *Casquero, O., Portillo, J., Ovelar, R., Benito, M., & Romo, J. (2010). IP Network: An integrated eLearning 2.0 architecture from a university's perspective. Interactive Learning Environments, 18(3), 293-308*
5. *.Chun, A. H. W. (2004). The agile teaching/learning methodology and its e-learning platform. In Advances in Web-Based Learning – ICWL 2004 (pp. 11-18). Springer, Berlin, Heidelberg.*
6. *Conde, M. Á., García-Peñalvo, F., Alier, M., Mayol, E., & Fernández-Llamas, C. (2014). Implement and design a service-based framework to integrate personal and institutional learning environments. Science of Computer Programming, 88, 41-53.*
7. *Coppeto, T. (2012). OSID-osid package description. Retrieved from http://osid.org/specifications/osid/package.html on April 25, 2015.*
8. *Cvetanovic, S., &Raspopovic, M. (2012). Design of learning object ontology for the database course. In Proceedings of the*

Third International Conference on e-Learning (pp. 108-113).

9. *Dodero, J. M., & Ghiglione, E. (2008). ReST-based web access to learning design services. IEEE Transactions on Learning Technologies, 1(3), 190-195.*
10. *García-Peñalvo, F. J., Conde, M. Á., Alier, M., &Casany, M. J. (2011). I am opening learning management systems to personal learning environments. Journal of Universal Computer Science, 17(9), 1222-1240.*
11. *Henderson, A., Twentyman, M., Heel, A., & Lloyd, B. (2006). Students' perception of the psycho-social clinical learning environment: An evaluation of placement models. Nurse Education Today, 26(7nj), 564–571.*
12. *IMS Global Learning Consortium. (2013). IMS Global learning information services specification. Retrieved from*
13. *Joordens, M., Chandrasekaran, S., Stojcevski, A., &Littlefair, G. (2012). The process of design-based learning: A students' perspective. In AAEE 2012: The Profession of Engineering Education, Advancing Teaching, Research, and Careers: Proceedings of the 23rd Annual Conference of the Australasian Association for Engineering Education (pp. 927-934). ESER Group, Swinburne University of Technology.*
14. *Teaching Software Architecture to Undergraduate Students: An Experience Report | IEEE Conference Publication | IEEE Xplore A Collaborative Approach to Teaching Software Architecture | Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*
15. *Djordjevic, J., Nikolic, B., & Milenkovic, A. (2005). Flexible web-based educational system for teaching computer architecture and organization. IEEE Transactions on Education, 48(2), 264-273. DOI: 10.1109/TE.2004.842918*
16. *ANwScCotUhttps://ieeexplore.ieee.org/abstract/document/8658714/?casa_token=vm0ddmrU0CIAAAAA:8vDdDrKbgYxLomKWPARFfk3JIIzyOxkXVyk_FnWOcEmUsXMguV8a68T2_jBwekax7Z8OXzKofdI*
17. *Nayak, A. S., & Vijayalakshmi, M. (2013). Teaching Computer System Design and Architecture Course - An Experience. 2013 IEEE International Conference in MOOC, Innovation and Technology in Education (MITE), Jaipur, India, 21-25. doi: 10.1109/MITE.2013.6756298*
18. *Wang, W., & Scannell, D. (2005). An Architecture-Based Software Reliability Modeling Tool and Its Support for Teaching. Proceedings - Frontiers in Education, 35th Annual Conference, Indianapolis, IN, USA, T4C-T4C. doi: 10.1109/FIE.2005.1611965.*
19. *Skalka, J., & Drlík, M. (2018). Priscilla – Proposal of System Architecture for Programming Learning and Teaching Environment. 2018 IEEE 12th International Conference on Application of Information and Communication Technologies (AICT), Almaty, Kazakhstan, 1-6. doi: 10.1109/ICAICT.2018.8746921.*
20. *A Better Way to Teach Software Architecture | SpringerLink*
21. *Djordjevic, J., Milenkovic, A., & Grbanovic, N. (2000). An Integrated Environment for Teaching Computer Architecture. IEEE Micro, 20(3), 66-74. doi: 10.1109/40.8463.*
22. *Delgado, D. J., Torres-Sáez, R., & Llamosa-Villalba, R. (2014). Develop an Executable Architecture for a System of Systems: A Teaching Management Model. Procedia Computer Science, 36, 80-86. doi: 10.1016/j.procs.2014.09.041.*
23. *Giraldo, F. D., Ochoa, S. F., Herrera, M., Clunie, C., Neyem, A., Zapata, S., Arciniegas, J. L., & Lizano, F. (2011). Applying a Distributed CSCL Activity for Teaching Software Architecture. International Conference on Information Society (i-Society 2011), London, UK, 208-214. doi: 10.1109/i-Society18435.2011.5978540.*
24. *Mustafa, B. (2013). YASS: A System Simulator for Operating System and Computer Architecture Teaching and Learning. European Journal of Science and Mathematics Education, 1(1), 34-42. DOI: 10.1016/j.scico.2016.07.001*
25. *Giraldo, F. D., Ochoa, S. F., Herrera, M., Clunie, C., Neyem, A., Zapata, S., Arciniegas, J. L., & Lizano, F. (2011). Applying a distributed CSCL activity for teaching software architecture. In International Conference on Information Society (i-Society 2011) (pp. 208-214). London, UK. DOI: 10.1109/i-Society18435.2011.5978540*
26. *Gonçalves, A. C., Graciano Neto, V. V., Ferreira, D. J., & Ferreira Silva, U. (2020). Flipped Classroom Applied to Software Architecture Teaching. In 2020 IEEE Frontiers in Education Conference (FIE) (pp. 1-8). Uppsala, Sweden. DOI: 10.1109/FIE44824.2020.9274255*
27. *Raspopović, M., Cvetanović, S., Stanojević, D., & Opačić, M. (2016). Software architecture for integration of institutional and social learning environments. Science of Computer Programming, 129, 92-102. DOI: 10.1016/j.scico.2016.07.001*
28. *Nichols, L., Dewey, K., Emre, M., Chen, S., & Hardekopf, B. (2019). Syntax-based Improvements to Plagiarism Detectors and their Evaluations. In Innovation and Technology in Computer Science Education (ITiCSE '19), July 15–17, 2019, Aberdeen, Scotland, UK. ACM, New York, NY, USA, 7 pages. DOI: 10.1145/3304221.3319789*
29. *Ji, Z., & Song, J. (2015). Improved Teaching Model for Software Architecture Course. In Proceedings of the 2015 International Conference on Education, Management, Information and Medicine (pp. 333-338). Atlantis Press. DOI: 10.2991/emim-15.2015.65*
30. *Teuscher, C., Haenni, J.-O., Gomez, F. J., Restrepo, H. F., & Sanchez, E. (1999). A tool for teaching and research on computer architecture and reconfigurable systems. In Proceedings 25th EUROMICRO Conference. Informatics: Theory and Practice for the New Millennium (pp. 343-350, vol. 1). Milan, Italy. DOI: 10.1109/EURMIC.1999.794490*
31. *Gordillo, A., López-Fernández, D., López-Pernas, S., & Quemada, J. (2020). Evaluating an Educational Escape Room Conducted Remotely for Teaching Software Engineering. IEEE Access, 8, 225032–225051. DOI: 10.1109/ACCESS.2020.3044380*
32. *Manzano-León, A., Rodríguez-Ferrer, J.M., Aguilar-Parra, J.M., Martínez Martínez, A.M., Luque de la Rosa, A., Salguero García, D., & Fernández Campoy, J.M. (2021). Escape*

Rooms as a Learning Strategy for Special Education Master's Degree Students. International Journal of Environmental Research and Public Health, 18(14), 7304. DOI: 10.3390/ijerph18147304

33. Yllana-Prieto, F., Jeong, J.S., & González-Gómez, D. (2021). An Online-Based Edu-Escape Room: A Comparison Study of a Multidimensional Domain of PSTs with Flipped Sustainability-STEM Contents. Sustainability, 13(3), 1032. DOI: 10.3390/su13031032
34. Gordillo, A., López-Fernández, D., López-Pernas, S., & Quemada, J. (2020). Evaluating an Educational Escape Room Conducted Remotely for Teaching Software Engineering. IEEE Access, 8, 225032–225051. DOI: 10.1109/ACCESS.2020.3044380
35. López-Pernas, S., Gordillo, A., Barra, E., & Quemada, J. (2021). Escapp: A Web Platform for Conducting Educational Escape Rooms. IEEE Access, 9, 38062-38077. DOI: 10.1109/ACCESS.2021.3063711
36. López-Belmonte, J., Segura-Robles, A., Fuentes-Cabrera, A., & Parra-González, M.E. (2020). Evaluating Activation and Absence of Negative Effect: Gamification and Escape Rooms for Learning. International Journal of Environmental Research and Public Health, 17(7), 2224. DOI: 10.3390/ijerph17072224
37. López-Pernas, S., Gordillo, A., Barra, E., & Quemada, J. (2021). Comparing Face-to-Face and Remote Educational Escape Rooms for Learning Programming. IEEE Access, 9, 59270-59285. DOI: 10.1109/ACCESS.2021.3073601
38. Estudante, A., & Dietrich, N. (2020). Using Augmented Reality to Stimulate Students and Diffuse Escape Game Activities to Larger Audiences. Journal of Chemical Education, 97(5), 1368–1374. DOI: 10.1021/acs.jchemed.9b00933
39. **Otemaie, K. R., Zanese, P. G., Bosso, N. S., & Grein, E. E.** (2020). Educational Escape Room for Teaching Mathematical Logic in Software Engineering. In Proceedings of SBGames 2020 (pp. 1-4).
40. Manzano-León, A., Rodríguez-Ferrer, J.M., Aguilar-Parra, J.M., Martínez Martínez, A.M., Luque de la Rosa, A., Salguero García, D., & Fernández Campoy, J.M. (2021). Escape Rooms as a Learning Strategy for Special Education Master's Degree Students. International Journal of Environmental Research and Public Health, 18(14), 7304. DOI: 10.3390/ijerph18147304
41. López-Belmonte, J., Segura-Robles, A., Fuentes-Cabrera, A., & Parra-González, M. E. (2020). Evaluating Activation and Absence of Negative Effect: Gamification and Escape Rooms for Learning. International Journal of Environmental Research and Public Health, 17(7), 2224. DOI: 10.3390/ijerph17072224
42. Veldkamp, A., Merx, S., & van Winden, J. (2020). Educational Escape Rooms: Challenges in Aligning Game and Education. Preprints, 2020100344. DOI: 10.20944/preprints202010.0344.v1
43. Marcondes, F. S., Almeida, J. J., & Novais, P. (2013). Applying a Distributed CSCL Activity for Teaching Software Architecture. International Conference on Information Society (i-Society 2011), 208-214. DOI: 10.1109/i-Society18435.2011.5978540