

# Development of a 3D Immersive Game for Games and Systems Requirements Capture

Thomas Hainey, Gavin Baxter and Raymond Holder

University of the West of Scotland, UK

[thomas.hainey@uws.ac.uk](mailto:thomas.hainey@uws.ac.uk)

[gavin.baxter@uws.ac.uk](mailto:gavin.baxter@uws.ac.uk)

[raymond.holder@uws.ac.uk](mailto:raymond.holder@uws.ac.uk)

**Abstract:** Requirements collection and elicitation is generally recognised as an abstract and complicated area with a reputation of being underdeveloped in computing and games development graduates. It is a necessary component of computing programmes on all sides of the curriculum including software engineering, information systems and games development. One highly problematic factor is the differences between general functional software and a serious game for educational purposes and while both can be classed as software, they have entirely different purposes. Games-based learning applications have been developed for teaching requirements collection where the focus has been on gathering requirements for a software system and this has proven to be effective in some respects. This paper will focus on the development of a 3D immersive serious game for gathering the requirements for a game for entertainment and for learning. The development of the game will be informed by the creation of a serious games' development model taking into account: learning outcomes, pedagogical content integration, general content integration in relation to graphical assets for appropriate realism and immersion, scaffolding and support mechanisms and assessment integration in relation to assessment mechanisms. Case studies will be discussed for the 3D immersive game in relation to fitness for purpose in relation to an entertainment and a serious game and compared and contrasted. The developed game will be evaluated by subject matter experts to ascertain its levels of effectiveness for gathering requirements for an entertainment game and for a serious game.

**Keywords:** Serious game, System requirements, Game requirements, Design, Requirements elicitation

---

## 1. Introduction

Requirements capture, requirements elicitation and requirements collection and analysis are highly important phases in the software development lifecycle and database application lifecycle but is often neglected (Gul et al, 2021; Knauss, Schneider and Stapel, 2008). If the requirements for a system/game are incorrect, then the client/player is dissatisfied, and this incurs greater expense the further along the development process by being more troublesome to correct. Development of functional software, computer games for entertainment and serious games are very different in terms of philosophy and approach. Traditional functional software is designed with the idea of efficiency in mind where the user fulfils their task and minimises their use with the software as much as possible. Games for entertainment have the completely opposite goal in mind where the idea is to elongate the players contact with the software (game) as much as possible and moreover to enhance lengthier playability. As a result, games tend not to have requirements per se but are a more creative, imaginative process involving fabrication. Serious games tend to sit in the middle of functional software and computer games for entertainment as they have a very specific purpose are generally contextualised and have a number of learning outcomes or training outcomes that have to be achieved on completion of the player playing the game. As a result, they have a creative process, however they are more specific in relation to contextualisation and learning requirements. This paper will review some of the previous work on games that have been developed for requirements elicitation and requirements collection and analysis and present a design and implementation of a 3D immersive game for requirements collection that is capable of incorporating a number of generic case studies.

## 2. Previous Work

Requirements collection and analysis is a highly important part of the software development and database application lifecycle. Requirements collection and analysis is probably more associated with the database application lifecycle and is defined by Connolly and Begg (2014) as *"The process of collecting and analysing information about the part of the organization that is to be supported by the database system and using this information to identify the requirements for the new system."* It involves fact finding using various techniques such as interviews, document, and process analysis of an existing system to establish the requirements for a new computerized software system. While games are technically software they differ in terms of overall purpose and creativity with functional software normally resulting from a requirements specification with the requirements being fully or partially known. In any event gathering these requirements in general requires

various soft and abstract skills where the main techniques for elicitation are interviews, questionnaires, task analysis, domain analysis, introspection, repertory grids, card sorting, laddering, group work, brainstorming, joint application development, requirements workshop, ethnography, observation, protocol analysis, apprenticing, goal-based approaches, scenarios, and viewpoints (Zowghi and Coulin, 2005). Schön (1983, 1987) highlights key problems teaching an abstract subject of this nature:

- It is learnable but not didactically or discursively teachable: it can be learned only in and through practical operations.
- It is a holistic skill and parts cannot be learned in isolation but by experiencing it in action.
- It depends upon the ability to recognise desirable and undesirable qualities of the discovered world. However, this recognition is not something that can be described to learners, instead it must be learned by doing.
- It is a creative process in which a designer comes to see and do things in new ways. Therefore, no prior description of it can take the place of learning by doing.

In relation to software engineering and systems design, the main traditional teaching approaches are lectures, role-play, paper-based case studies and live-through case studies. It is important to understand the advantages and disadvantages of these approaches with regards to requirements collection and elicitation (Brooks, 1987). Studies that have investigated traditional teaching methods have been synthesized in Table 1 adapted from Bonwell (1996), Cashin (1985), Wehrli and Nyquist (2003), Davis (2001), the ADPRIMA Instructional Methods information website (2023), Biggs (1999), Shaw and Dermoudy (2005), Polack-Wahl (1999), Andrianoff and Levine (2002), Cope and Horan (1996), Bernstein and Klappholz (2001), Barrett (1997), Simsarian (2003) and Connolly, Stansfield, McLellan, Ramsay and Sutherland (2004). Table 1 shows a compilation of the attributes of traditional teaching approaches rated on a low, medium, high scale. For a full discussion of traditional techniques see Hainey (2010).

**Table 1: Attributes of traditional teaching approaches**

Attributes	Lectures	Role-play	Paper-based case studies	Live-through case studies
<b>Instructor has control of the learning, has to be an effective speaker and communicate the intrinsic interest of a subject through enthusiasm.</b>	HIGH	MEDIUM	LOW	LOW
<b>Participants are actively involved in the exercise provide experiential learning and deeper understanding to have a considerable lasting effect.</b>	LOW	HIGH	MEDIUM	LOW
<b>Can be time consuming and therefore not always practical.</b>	MEDIUM	MEDIUM	MEDIUM	HIGH
<b>Scales with larger numbers of students and/or quantity of material</b>	HIGH	LOW	HIGH	HIGH
<b>Can provide a model of how professionals address problems and questions.</b>	HIGH	MEDIUM	LOW	HIGH
<b>Can provide elements of variety, reality and specificity.</b>	MEDIUM	HIGH	MEDIUM	MEDIUM
<b>Likelihood of transfer to the real world is improved.</b>	MEDIUM	HIGH	MEDIUM	MEDIUM
<b>Appeals to learners who learn by listening.</b>	HIGH	HIGH	LOW	LOW
<b>Appeals to learners who learn by reading.</b>	HIGH	LOW	HIGH	HIGH
<b>Appeals to learners who learn from demonstration.</b>	LOW	HIGH	LOW	LOW
<b>Appeals to learners who learn from discussion.</b>	MEDIUM	HIGH	LOW	LOW
<b>Appeals to learners who learn from practice by doing</b>	LOW	HIGH	HIGH	MEDIUM
<b>Presents a risk for learners for example, tension or embarrassment.</b>	LOW	HIGH	LOW	LOW
<b>Allows the learners to identify the mistakes of previous developers, thus reducing the possibility of making the same mistake.</b>	MEDIUM	LOW	LOW	HIGH

Attributes	Lectures	Role-play	Paper-based case studies	Live-through case studies
Provides a safe environment to increase practice experience when real life experiences are unavailable, for example highlighting client frustration.	MEDIUM	HIGH	MEDIUM	LOW
Immediate feedback is provided.	LOW	HIGH	LOW	LOW
Can give new perspectives on situations.	MEDIUM	HIGH	MEDIUM	LOW
Learners can apply newly developed analytical problem solving skills in their own time to find a large number of solutions for complex issues.	LOW	MEDIUM	HIGH	HIGH
Maintains the attention of the learner.	LOW	HIGH	MEDIUM	MEDIUM
Suited for teaching abstract, complex subjects, or higher order thinking skills such as values, motor skills, analysis and application etc.	LOW	MEDIUM	MEDIUM	MEDIUM
Takes into account that learners are at different levels of understanding and learn at different paces.	LOW	MEDIUM	LOW	LOW
Puts pressure on the learner to perform.	LOW	HIGH	LOW	LOW
May require a debriefing session.	MEDIUM	HIGH	MEDIUM	MEDIUM
Has to be well planned, monitored and orchestrated or it may lack focus.	MEDIUM	HIGH	MEDIUM	MEDIUM

### 3. Previous Games on Requirements Collection and Elicitation

Several serious games and games-based learning applications were identified in the areas of computer science, software engineering and information systems by Connolly, Stansfield and Hainey (2007) including: KMQuest, The Incredible Manager, Open Software Solutions and SimSE. A further study performed by Hainey (2009) identified other games including: SIM VBSE RPG-SE and Anti-Phishing Phil which were relatively mature games at that time. The research performed was looking for any relatively mature games used to teach requirements collection and analysis or more generally software project management and lead to the development and evaluation of a specially constructed requirements collection and analysis game (Hainey, 2010). Hainey, Connolly, Stansfield and Boyle (2011) performed an evaluation that game for requirements elicitation finding that games-based learning can be a suitable approach to teaching requirements collection specifically in relation to:

- Identifying user requirements
- Identifying system requirements
- Resolving conflict and seeking clarification.

Given the increased interest in serious games both from a developmental and research perspective, there has been relatively limited further advancement in relation to developing serious games or games-based learning applications specifically for requirements capture/elicitation and requirements collection and analysis when comparing it to a mainstream subject such as Maths in the past 13 years. This is possibly because of the widespread development of serious games in relation to more easily applicable topics where this may be exacerbated by the fact that abstract topics are difficult to teach in general but have to be particularly well planned in a serious game. Ibrahim et al., (2019) identified 10 serious games for requirements elicitation where only one of them was from 2019 and the rest were pre-2010, 2010 and 2011. Soo and Aris (2018) performed an overview of games-based learning in requirements engineering and discovered eight games where some are duplicates with the same sort of time frame ranging from 2003 to 2011 with very little reported between 2011 and the present. This indicates that very few games have been developed on this topic over the past decade. A highly interesting point to reflect on is what the activities associated with requirements elicitation entail in relation to learning outcomes. After performing an extensive literature reviewing and analysing the games discovered, Ibrahim et al., (2019) linked learning objectives to learning activity and the measurement to ascertain if learning had occurred. Table 2 presents these learning outcomes linked to measurements and game activity.

Table 2: Learning outcomes linked to game activity and measurements (Ibrahim et al., 2019)

Learning outcomes	Activity	Measurement
Discovering requirements	Identifying NPCs, documents, objects, and scenes in the game to collect the requirements	NPCs, documentation, scenes, objects, and observations correctly identified, and requirements collected.
Analyzing requirement completeness	Identification of any additional requirements not identified in the game	Percentage/number of correctly identified requirements. Number of additional requirements identified.
Analyzing requirement relevance	Determining requirements are correct	Number of correct requirements included and excluded from the final list
Analyzing requirement consistency	Determining if requirements are inconsistent	Number of correctly identified inconsistent requirements
Analyzing requirement practicality	Determining if requirements are impractical	Number of correctly identified impractical requirements
Negotiating final system requirements	Collaborative discussion with the learners about the requirements	Number of negotiated requirements that are acceptable

Very few full games for requirements elicitation, requirements capture, or requirements collection and analysis have been developed since 2011, however there was the development of a requirements application process model (Durugbro and Kalverkamp, 2012). Gul et al., (2021) performed a systematic literature review to identify gamification elements used in requirements elicitation and discovered the following techniques other than Problem-Based Learning:

- Trophies,
- Levelling Up,
- Medals and Scoreboards,
- Virtual Goods and Levels,
- User Profile and Storytelling,
- Activity Feed and Story,
- Voting/Voice,
- Feedback and Missions,
- Levels and Time Limit,
- Blocked Content,
- Teammates,
- Achievement and Gifting,
- Quest and Guilds or Teams,
- Voting Mechanism and Status,
- Penalty and Count Down,
- Count Down and Performance Graph,
- Novelty,
- Progression.

Garcia et al., (2018) performed a study at the University of Mexico utilising a game called Biyubi (the word meaning to explore, analyse and discover). The study focused on interviews and utilised 94 participants who

were randomly divided into an experimental and control group. The experimental group utilised the game while the control group utilised a traditional teaching approach. The game was more advanced than previous games developed in 2016 and was 3D and immersive with three simple learning outcomes. The results indicated that the motivation, satisfaction, and the learning experience of the students increased.

#### **4. Game Design**

The development of a serious game in Unreal Engine 5 is set to be an office environment for the purposes of situated/contextualised learning by providing authentic activity and assessment (Brown, Collins and Duguid, 1989). The game allows character customisation and players can choose between two different characters, male or female, and interact with approximately 10 NPCs, including a receptionist, manager, HR person, and others in various roles. The game features a questing/task system for players to complete tasks and a dialogue system for communication between the player and NPCs.

The 10 scenarios are designed to meet the needs of the project focused on the requirements elicitation process and using a serious game to achieve learning outcomes. The project aims to teach learners about the process of gathering software requirements, understanding different software development lifecycles, practicing interview and response skills, and adapting to changing requirements in real-world client situations. Each scenario is crafted to address specific aspects of these learning objectives. The scenarios are as follows:

**Scenario 1: Office Orientation** - Players become familiar with the office environment, colleagues, and company culture, which can impact the requirements elicitation process. Upon starting at the new office, the player receives a task to familiarise themselves with the workspace and meet their colleagues. The player must interact with different NPCs, such as the receptionist, HR person, and manager, to learn about their roles and gather information about the company culture and office policies.

**Scenario 2: Health and Safety Awareness** - Players identify potential hazards, indirectly learning about the importance of considering non-functional requirements such as safety. The player is assigned a task to identify potential health and safety hazards in the office. They must navigate the office space and locate hazards such as blocked fire exits, exposed electrical wires, or cluttered walkways. Once all hazards have been identified, the player must report their findings to the HR person (NPC) and discuss ways to mitigate the risks.

**Scenario 3: Collaborative Project Planning** - Players collaborate with team members, learning about their roles and responsibilities, and develop a project plan, which is a critical aspect of the software development lifecycle. The player is assigned to a new project team and must collaborate with other team members (NPCs) to create a project plan. The player needs to engage in dialogue with each team member to understand their roles, responsibilities, and potential dependencies. Based on the discussions, the player must then help the team develop a project timeline and allocate resources effectively.

**Scenario 4: Gathering Requirements** - Players practice the core skill of requirements elicitation by conducting interviews and collecting information from NPCs. The player receives a task to interview a client (NPC) who wants to develop a custom software solution. The player must use their dialogue options to ask open-ended questions and gather the client's needs, expectations, and constraints. They must then document the collected information and present it to their manager for review.

**Scenario 5: Prioritising Requirements** - Players learn to prioritise requirements based on importance, complexity, and dependencies. The player must participate in a group brainstorming session with other team members (NPCs) to identify potential features and requirements for a new database management system. The player must use the dialogue system to contribute ideas, discuss pros and cons, and reach a consensus with the team. The outcome of the session should be a list of prioritised features and requirements.

**Scenario 6: Updating Requirements** - Players experience the dynamic nature of requirements as they adapt to changing client needs. The player must manage a situation where a client (NPC) changes their requirements midway through the project. The player must first understand the new requirements by engaging in a dialogue with the client. They must then assess the impact of these changes on the project timeline and budget and negotiate with the client to reach a mutually acceptable solution.

**Scenario 7: Requirement Validation** - Player's practice validating and refining requirements to ensure they are accurate and complete. The player receives a task to identify and interview key stakeholders (NPCs) within the organisation to gather their perspectives on the development of a serious game. Through dialogue, the player must determine each stakeholder's needs and concerns and consolidate this information to create a comprehensive set of requirements for the game.

**Scenario 8: Documenting Requirements** - Players learn to create a well-structured requirements specification document for formative feedback and assessment. After gathering and documenting requirements for a software project, the player must present their findings to the project team (NPCs) for validation. The player must use the dialogue system to address questions and concerns raised by the team members and revise the requirements document as needed to ensure that it accurately reflects the stakeholders' needs and expectations. The final requirements specification should then be submitted for official formative feedback and assessment.

**Scenario 9: Conflict Resolution** - Players learn to navigate conflicts and find solutions, an essential skill when dealing with diverse stakeholder requirements and expectations. The player witnesses a conflict between two colleagues (NPCs) regarding a project decision. The player must intervene and use their dialogue options to understand the root cause of the disagreement, facilitate a constructive conversation, and help the colleagues reach a mutually acceptable resolution.

**Scenario 10: Reflective Learning** - Players reflect on their learning experience, considering feedback from their colleagues and assessing their progress in mastering requirements elicitation skills. The player's manager (NPC) schedules a performance review meeting. The player must gather evidence of their achievements and contributions to the company, such as completed projects or positive feedback from clients or colleagues. The player should interact with various NPCs to gather this information and compile it into a summary to present during the performance review.

By using a serious game, players can engage with the learning material in an immersive, interactive, and enjoyable manner. The game provides a realistic simulation of the requirements elicitation process, allowing learners to practice and apply their skills in a safe, low-stakes environment. This approach promotes experiential learning (Kolb 1984), enabling players to build their knowledge and understanding of the requirements elicitation process through hands-on experience, ultimately leading to better retention and application of the skills in real-world situations. The initial idea was to create a more immersive contemporary version of the requirements collection and analysis game by Hainey, Connolly, Stansfield, and Boyle (2011) and attempt to bring it into modern times. The idea was also to change the case studies to allow collection of the requirements for normal functional software, a serious game and a game for entertainment. A screen shot of the requirements collection and analysis game is provided in Figure 1 for comparison. Figure 2(a) and Figure 2(b) show a screen shot of the new game developed to modern standards i.e., an 3D immersive game.



Figure 1: The requirements collection and analysis game (Hailey, 2010)

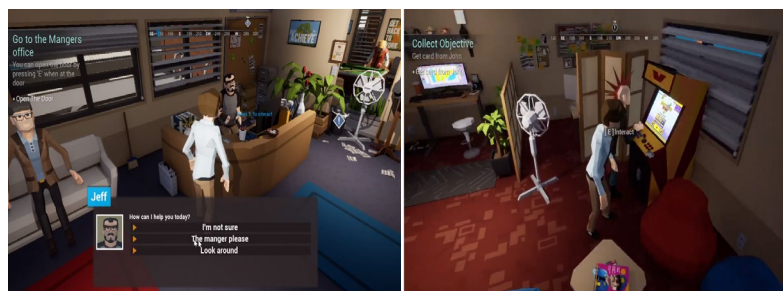


Figure 2(a) and Figure 2(b): Screen shots of Cubicle Conundrums: Quest for Clarity

#### **4.1 Games Design Document**

The Games Design Document for the developed game is as follows:

**GAME TITLE:** Cubicle Conundrums: Quest for Clarity

**GAME OVERVIEW:** Cubicle Conundrums is an immersive serious game that simulates the software requirements elicitation process in an office environment. The game aims to teach learners about various aspects of gathering software requirements, navigating different software development lifecycles, and practising interview and response skills. The target audience for this game includes students, software developers, project managers, and other professionals interested in improving their requirements elicitation skills. The game will be developed for PC platforms.

**GAME WORLD AND SETTING:** The game is set in a contemporary office, featuring an open layout and various rooms representing typical office spaces. The game world is designed with low-poly assets and simple shaders, providing a stylised and visually appealing environment. The atmosphere is light-hearted, with humorous elements and familiar office stereotypes. The office includes areas such as a reception, meeting room, HR office, manager's office, dining area, kitchen, server room, cubicles, media room, storage rooms, and bathrooms.

#### **CHARACTERS**

**Playable characters:** The game offers two playable character options (1 male, 1 female) with distinct appearances and clothing options. The characters have similar abilities and interactions in the game world.

**Non-playable characters (NPCs):** The game features around 10 NPCs, including a receptionist, manager, HR person, and other colleagues in various roles. Each NPC has a unique appearance, role, and interactions with the player.

#### **GAMEPLAY MECHANICS**

**Controls:** Players can control their character using a keyboard and mouse or a game controller, following a typical third-person control scheme found in 3D games.

**Interactions:** Players can interact with NPCs and objects in the game, such as opening doors, collecting items, and interacting with screens. Players initiate dialogue with NPCs by pressing the 'E' key when close to them.

**Quest/task system:** The game features a quest/task system that assigns objectives for players to complete as part of the gameplay experience. The system is integrated into a user interface (UI) HUD display, with options for tracking tasks and a compass system to guide players.

**Progression and rewards:** The game progresses as players complete tasks, unlock new scenarios, and gain experience in the requirements elicitation process.

#### **USER INTERFACE (UI) AND USER EXPERIENCE (UX)**

**Heads-up display (HUD):** The game features a HUD displaying relevant information such as active tasks, compass direction, and interaction prompts.

**Menus and screens:** The game offers menus for character selection, settings, and other in-game options.

**Control mapping:** Controls are mapped to allow for seamless gameplay on keyboard/mouse or game controller setups.

**Visual and audio feedback:** The game provides visual and audio cues to guide the player and signal progress or successful interactions.

#### **DIALOGUE SYSTEM**

**System overview:** The game utilises a Blueprint/UMG-based dialogue and scripting system that enables developers to create dialogue content using data-driven design and a range of inline text formatting tools and configurable interface options. The system supports branching and non-branching dialogue styles.

**Integration:** The dialogue system is integrated into the game world using actors, actor components, and UMG widgets.

## **QUESTS AND SCENARIOS**

The game features 10 scenarios, each focusing on different aspects of the requirements elicitation process, such as gathering requirements, prioritising, updating, validating, documenting, and resolving conflicts. Each scenario has a set of tasks and learning objectives that the player must complete to progress through the game. The game employs a save/load system to store progress for each quest, ensuring players can pick up where they left off in their learning journey.

## **5. Conclusions and Future Directions**

Requirements elicitation, engineering and requirements collection and analysis is a highly abstract and difficult subject with many approaches including, identification of stakeholders, interviews, questionnaires, task analysis, domain analysis, introspection, repertory grids to name but a few. Serious games have now developed to the stage where they can be developed more easily utilising game engines, however, there has been very few games developed for this particularly important area in the past 13 years besides one highly advanced game yielding positive results to teach interviewing and stakeholder identification (Garcia et al., 2018). This paper has provided an overview of the development and design of a highly immersive serious game for requirements collection and analysis following on from research performed in 2010. The game developed in this project was demonstrated at a Games Development showcasing event with industry representatives/subject matter experts at the University of the West of Scotland in May 2023. The intricate level design of the game and highly detailed environment received exceptionally high praise from games industry representatives in relation to design. The game is now in the final stages of development where particular case studies associated with finding the requirements of a game for entertainment and the requirements of a serious game are being integrated into it in relation to endogenous, intrinsic content integration. The game will then be evaluated in a Serious Games course as well as a Computer Games Development Programme to determine if a serious game can be an effective tool for learning in this context. To some extent this has already been proven, however increased focus on whether the immersive experience is more effective and whether highly detailed case studies associated with games for entertainment, functional software and serious games can be effective in isolation. Performing comparisons between the case studies to ascertain if a serious game for requirements collection can be more effective in various scenarios is a potential future research direction given that modern serious games can be created with sufficient complexity in terms of pedagogical content, level design, graphics, audio and immersion.

## **References**

- ADPRIMA Instructional methods information website (2023). Retrieved 21 May, 2023, from <http://www.adprima.com/teachmeth.htm>
- Andrianoff, S. and Levine, D. (2002). Role playing in an object-oriented world. Proceedings of the 33rd SIGCSE Technical Symposium on Computer Science Education, Kentucky, USA. ACM Press, 121–125.
- Barrett, M.L. (1997). Simulating requirement gathering. In *Proceedings of the 28<sup>th</sup> SIGSCE Technical Symposium on Computer Science Education*, San Jose, California, USA, 310–314.
- Bernstein, L. and Klappholz, D. (2001). Getting Software Engineering into Our Guts. *Cross Talk, The Journal of Defense Software Engineering*, July 2001, 25-26.
- Biggs, J. (1999). What the Student Does: teaching for enhanced learning, *Higher Education Research & Development*, 18:1, 57-75, DOI: 10.1080/0729436990180105.
- Bonwell, C.C. (1996). Enhancing the lecture: Revitalizing a traditional format. In *Using active learning in college classes: A range of options for faculty*, New Directions for Teaching and Learning No. 67 (Eds: T.E. Sutherland and C.C. Bonwell), 31 - 44. Place: Wiley Periodicals Inc.
- Brooks, F. (1987). No silver bullet: Essence and accidents of software engineering. *IEEE Computer*, 20(4) 10-19. 160.
- Brown, J. S., Collins, A., & Duguid, P. (1989). Situated Cognition and the Culture of Learning. *Educational Researcher*, 18(1), 32–42. <https://doi.org/10.3102/0013189X018001032>.
- Cashin, W.E. (1985). Improving lectures Idea Paper No. 14. Manhattan: Kansas State University, Center for Faculty Evaluation and Development.
- Connolly, T. M. and Begg, C. (2014). *Database Systems: A Practical Approach to Design, Implementation, and Management*, Global Edition, PEARSON.
- Connolly, T.M., Stansfield, M.H. and Hainey, T. (2007). An application of games-based learning within software engineering. *British Journal of Educational Technology*. 38(3), 416–428.
- Connolly, T.M., Stansfield, M.H, McLellan, E., Ramsay J. and Sutherland J. (2004). Applying Computer Games Concepts to Teaching Database Analysis and Design. In *Proceedings of the International Conference on Computer Games, AI, Design and Education*, Reading, UK, November 2004
- Cope, C. and Horan, P. (1996). The Role Played Case: An Experiential Approach to Teach.
- Davis, H. A. (2001). The quality and impact of relationships between elementary school children and teachers. *Journal of Contemporary Educational Psychology*, 26, 431–453.



- Durugbo, C and Kalverkamp, M. (2012). Uncovering requirements using serious games, *18th International ICE Conference on Engineering, Technology and Innovation*, Munich, Germany, 2012, pp. 1-10, doi: 10.1109/ICE.2012.6297674.
- García, I.A., Pacheco, C.L., León, A., & Calvo-Manzano, J.A. (2018). Experiences of using a game for improving learning in software requirements elicitation. *Computer Applications in Engineering Education*, 27, 249 - 265.
- Gul, S., Dar, H. S., Malik, A. R., Zulqarnain, M., & Imtiaz, S. (2021). Gamification and gaming elements for software requirements elicitation: A systematic literature review. *International Journal on Electrical Engineering and Informatics*, 13(4), 931-950. doi:<https://doi.org/10.15676/ijeei.2021.13.4.11>
- Haine, T. (2009). Games-based Learning in Computer Science, Software Engineering and Information Systems. School of Computing. Computing and Information Systems. 13(3).
- Haine, T. (2010). Using Games-based Learning to Teach Requirements Collection and Analysis at Tertiary Education Level: Unpublished Doctoral Dissertation (Available: May 2023) from: [https://www.researchgate.net/publication/230649488\\_Using\\_Games-Based\\_Learning\\_to\\_Teach\\_Requirements\\_Collection\\_and\\_Analysis\\_at\\_Tertiary\\_Education\\_Level](https://www.researchgate.net/publication/230649488_Using_Games-Based_Learning_to_Teach_Requirements_Collection_and_Analysis_at_Tertiary_Education_Level)
- Haine, T., Connolly, T. M., Stansfield, M., & Boyle, E. A. (2011). Evaluation of a game to teach requirements collection and analysis in software engineering at tertiary education level. *Computers & Education*, 56(1), 21–35, doi: <https://doi.org/10.1016/j.compedu.2010.09.008>
- Ibrahim, Z. Soo, M. C. Soo, M. T. and Aris, H. (2019). Design and Development of a Serious Game for the Teaching of Requirements Elicitation and Analysis," *2019 IEEE International Conference on Engineering, Technology and Education (TALE)*, Yogyakarta, Indonesia, 2019, pp. 1-8, doi: 10.1109/TALE48000.2019.9225987.
- Kolb, D. A. (1984). *Experiential learning: Experience as the source of learning and development* (Vol. 1). Englewood Cliffs, NJ: Prentice-Hall.
- Knauss, E., Schneider, K. and Stapel, K. (2008). A Game for Taking Requirements Engineering More Seriously, 2008 Third International Workshop on Multimedia and Enjoyable Requirements Engineering - Beyond Mere Descriptions and with More Fun and Games, Barcelona, Spain, 2008, pp. 22-26, doi: 10.1109/MERE.2008.1.Polack-Wahl, J.A., (1999). Incorporating the client's role in a software engineering course. *SIGCSE Bulletin*, March, pp. 73 - 77.
- Shaw, K. and Dermoudy, J. (2005). Engendering an empathy for software engineering. In *Proceedings of the 7th Australasian Computing Education Conference (ACE2005)*, Newcastle, Australia, 42, 135–144.
- Simsarian, K. T (2003). Take it to the next stage: the roles of role playing in the design process. In *CHI '03 Extended Abstracts on Human Factors in Computing Systems (CHI EA '03)*. Association for Computing Machinery, New York, NY, USA, 1012–1013. <https://doi.org/10.1145/765891.766123>
- Schön, D.A. (1983). *The Reflective Practitioner: How Professionals Think in Action*. Basic Books, New York.
- Schön, D.A. (1987). *Educating the Reflective Practitioner: Towards a New Design for Teaching in the Professions*. Jossey-Bass Inc., San Francisco.
- Soo MT, Aris H (2018) Game-based learning in requirements engineering: an overview. In: *Proceedings of 2018 IEEE conference on e-Learning, e-Management and e-Services (IC3e)*, pp 46–51
- Wehrli, G. and Nyquist, J. G. (2003). Creating an Educational Curriculum for Learners at Any Level. Paper presented at the AABB Conference.
- Zowghi, D., & Coulin, C. (2005). Requirements Elicitation: A Survey of Techniques, Approaches, and Tools.