Available online at www.worldscientificnews.com

World Scientific News

An International Scientific Journal

WSN 96 (2018) 149-178 EISSN 2392-2192

Private Eye: A Gameplay for Problem Solving Skills

Tomas U. Ganiron Jr*, Johara G. Alfad, Blaise A. Florendo and **Reylan J. Amoncio**

College of Computer Studies, AMA Computer College, Makati City, Philippines

*E-mail address: tuganironjr@amaes.edu.ph

ABSTRACT

The use of game-based learning relates to the use of games to enhance the problem-solving skills. This paper will present the findings of observational research of Private Eye as a gameplay using rudimentary programming concepts. The Private Game Play encompasses executables, source code, architecture descriptions, and so on. As a result, the notion of user extends to operators as well as to programmers, which are users of components such as software libraries. The standard provides a framework for organizations to define a quality model for a software product. In doing so, however, it leaves up to each organization the task of specifying precisely its own model. This may be done, for example, by specifying target values for quality metrics which evaluates the degree of presence of quality attributes. The study will assess the software quality of the Private Eye Gameplay based on ISO/IEC, 9126-1 It will also contribute to the empirical evidence in games-based construction by providing the results of observational research across different levels of its software quality and will provide pedagogical guidelines for assessing programming ability using a games-based construction approach.

***Keywords*:** Animation, functionality, gameplay, private eye, software engineering, usability, video game

1. INTRODUCTION

A video game is essentially the same form of entertainment but refers not only to games played on a personal computer but also to games run by a console or arcade machine. It also

(Received 14 March 2018; Accepted 02 April 2018; Date of Publication 03 April 2018)

include games which display only text or which use other methods, such as sound or vibration, as their primary feedback device, or a controller (console games), or a combination of any of the above.

Everyone knows the best gameplay are those that are challenging. Defeating that "impossible" boss or getting that "Ultra-rare" achievement feels like real accomplishments (Lin et al., 2006). That rush of dopamine really does teach users perseverance and personal accountability. The users teach to focus on their own actions and help to control the situations around them. Lessons like this are easily applied in the real world.

In 2012, a research team in New Zealand used a game called SPARX to help treat depressed teenagers. The game, to be fair, was specifically designed to provide therapy for kids that were both fun and therapeutic. The study included 168 teenagers with an average age of 15 who were struggling with depression. Half of them were used as a control group and given regular one-on-one counseling whilst the other played SPARX. The SPARX group of users showed encouraging results. 44% of the players completely recovered from depression with only 26% from the conventional treatment group (Ke, 2008).

Games that have a large following or social aspect can really help with making friends. Some games even require users to go outside, take Pokemon-Go, for example. The real benefit from games isn't playing them per se, but the community they build around the shared experience. Gaming conventions, online multiplayer sessions, and competitions help players meet new people and make friends with a common interest. Some happily married couples can even claim to have met because of gaming (Paulk, 1993).

In February of 2013, Italian researchers submitted evidence that playing computer games can help everyday skills like reading. They took two groups of children aged 7 to 13 and had one team play "Rayman Raving Rabbids" whilst the other played a slower-paced game (O'Connor, 2012). When they tested the reading skills of the two groups, they found that the ones who played the action game could actually read faster and more accurately.

The intensive situation awareness and management needed when playing computer games like Call of Duty seem to improve gamers' ability to deal with visual and auditory distractions simultaneously. This might seem obvious but it really does help with real life productivity. Developing multi-tasking is a great everyday skill to master. It helps productivity in general. This will spill over to the real-world applications, at home, and in the workplace (Wagner, 2013).

In 2013, German researchers studied how computer games like Super Mario 64 affect the brains of players (Lorant-Royer et al., 2010). They asked 23 adults with the median age of 25 to play "Super Mario 64" for 30 minutes a day for two months. They also had a control group who played no games at all. After scanning the brains of the players and control group using an MRI machine, they found the gamers had a remarkable increase in gray matter in the right hippocampus, right prefrontal cortex, and cerebellum (Lorant-Royer et al., 2010). These are the areas responsible for memory, strategy, fine motor skills and spatial navigation.

A 2009 study by the University of Rochester found that shooting "bad guys" can actually improve your vision. Players who enjoyed wasting pixelated enemies in games like Unreal Tournament 2004 or Call of Duty had a boost in their contrast sensitivity function. This is the ability to discern subtle changes in brightness of the image (Olsina et al., 2006). The researchers also believe that video games could be used in the future to help as an aid to correct bad eyesight.

It is not yet clear whether computer games improve general intelligence but they certainly seem to help teach us new skills and reinforce them. This doesn't necessarily mean users are getting smarter though. There are some games that are designed for problem-solving, critical thinking and reading comprehension built in (Aoyama, 2004). Of course, there is also a large swathe of educational apps and games for children of all ages too.

Whether or not users are a hardcore gamer, perhaps treating life like a computer game could improve their general quality of life. It could, for example, improve their fitness or help them fulfill their goals. If they were to think of themselves as a character starting at levels 1, users might be able to identify areas in need of improvement. Work in those abilities and keep track of their progress.

Video game traditionally referred to a raster display device, but as of the 2000s, it implies any type of display device that produces two or three-dimensional images. Some theorists categorize video game as an art form, but this designation is controversial. As video games are increasingly the subject of scientific studies, game genres themselves are becoming a subject of study (Aloupis et al., 2015).

However, logic games require the player to solve logic puzzles or navigate complex locations such as mazes (Ryan et al., 2006). They are well suited to casual play, and tile-matching puzzle games are among the most popular casual games. This genre frequently crosses over with adventure and educational games.

The researchers created a Private Eye Gameplay. This is a unique mystery game with two levels and challenging puzzles. Compelling stories are aided in the game with a dialogue of the characters with voices and adjusting the sounds of the game. The gamers can only play this game on a computer. The operating system of the game should be at least Windows 7 (64bit) in order to play this game.

To validate the quality of Private Eye, it should pass the ISO 9126 Software. ISO 9126 is an international standard for the evaluation of software (Castellar et al., 2014). The standard is divided into four parts which address, respectively, the following subjects: quality model; external metrics; internal metrics; and quality in use metrics. The ISO 9126-1 software quality model identifies 6 main quality characteristics, namely: functionality, reliability, usability, efficiency, maintainability, and portability (Dickey, 2005).

Functionality is the essential purpose of any product or service. For certain items, this is relatively easy to define, for example, a ship's anchor has the function of holding a ship at a given location. The more functions a product has, e.g. an ATM machine, then the more complicated it becomes to define it is functionality. For software a list of functions can be specified, i.e. a sales order processing systems should be able to record customer information so that it can be used to reference a sales order (Carnagey et al., 2007).

Reliability once a software system is functioning, as specified, and delivered the reliability characteristic defines the capability of the system to maintain its service provision under defined conditions for defined periods of time. One aspect of this characteristic is fault tolerance that is the ability of a system to withstand component failure. For example, if the network goes down for 20 seconds then comes back the system should be able to recover and continue functioning (Shaker et al., 2011).

Usability only exists with regard to functionality and refers to the ease of use for a given function (Chisholm et al., 2010). For example, a function of an ATM machine is to dispense cash as requested. Placing common amounts on the screen for selection, i.e. $20.00, $40.00, $100.00 etc., does not impact the function of the ATM but addresses the Usability of

the function (Bang et al., 2006). The ability to learn how to use a system (learnability) is also a major sub characteristic of usability (Boot et al., 2008).

Efficiency is concerned with the system resources used when providing the required functionality (Behm-Morawiz et al., 2006). The amount of disk space, memory, network etc. provides a good indication of this characteristic. As with a number of these characteristics, there are overlaps. For example, the usability of a system is influenced by the system's Performance, in that if a system takes 3 hours to respond the system would not be easy to use although the essential issue is a performance or efficiency characteristic (Castel et al., 2005).

Maintainability is the ability to identify and fix a fault within a software component is what the maintainability characteristic addresses (Cain et al., 2012). In other software quality model, this characteristic is referenced as supportability. Maintainability is impacted by code readability or complexity as well as modularization. Anything that helps with identifying the cause of a fault and then fixing the fault is the concern of maintainability. Also, the ability to verify (or test) a system, i.e. testability, is one of the sub-characteristics of maintainability environment or with its requirements (Dromey, 1995). The sub-characteristics of this characteristic include adaptability. Object-oriented design and implementation practices can contribute to the extent to which this characteristic is present in a given system.

In this research, the story of the Private Eye Gameplay is all about a detective named Mike Doyle who is working at X Eye Company. Mike is a hard-working detective who has a jealous boss. That extreme jealousy can do everything. His boss will try to destroy Mike. This game will show that jealous will destroy human being and to fight in a proper way. Gameplay also teaches to be a grown-up that man should act.

The objective of the study is to help the players to improve their critical thinking and logical thinking because they engage themselves in the process of clearly moving from one related thought to another and solving complex problems. This game is an epoch-making discovery in providing knowledge entertainment to the players and satisfaction to the gamers who completed the given puzzles.

This proposed game will test problem-solving skills including logic, pattern recognition and solving sequences. The limitation of this game reduces a complex system to their core aspects and are merely a program representation of the real world, yet the world in which we live is more complex and cannot be boiled down to the simple logic of the game. The game played in computers only.

2. PROJECT DEVELOPMENT

In Figure 1, the researchers develop Private Eye Gameplay through evaluating the system in the standard of ISO 9126 in terms of functionality, reliability, usability, maintainability, efficiency, and portability. Analyze the standard of ISO 9126 and research about previous mysterious puzzle game to be able to get an idea. Evaluate the game if it passed the standard of ISO 9126 or the system failed. If the system passed the game, it should be published but if it failed, the system must retake the first process.

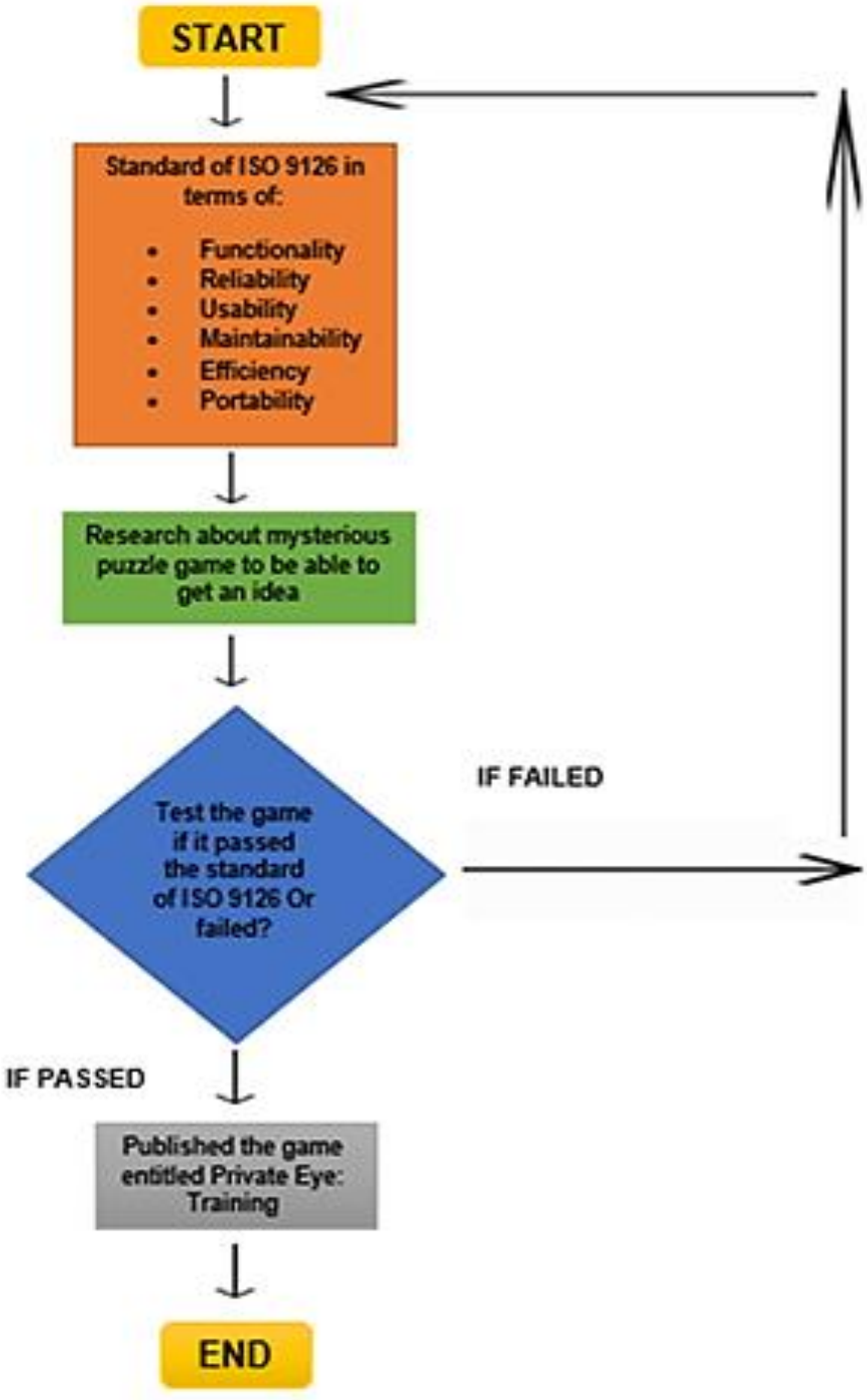

Figure 1. Project Development

In Figure 2, the image shows the house where the player roamed and solves the puzzle while in Figure 3, the image shows that the game can interact with the things like doors, cabinets and etc. Figure 4 shows the code for the movements of the character. The Language is C# and the proponent used Unity in creating the game.

Figure 2. The Player is roaming and solving the puzzle

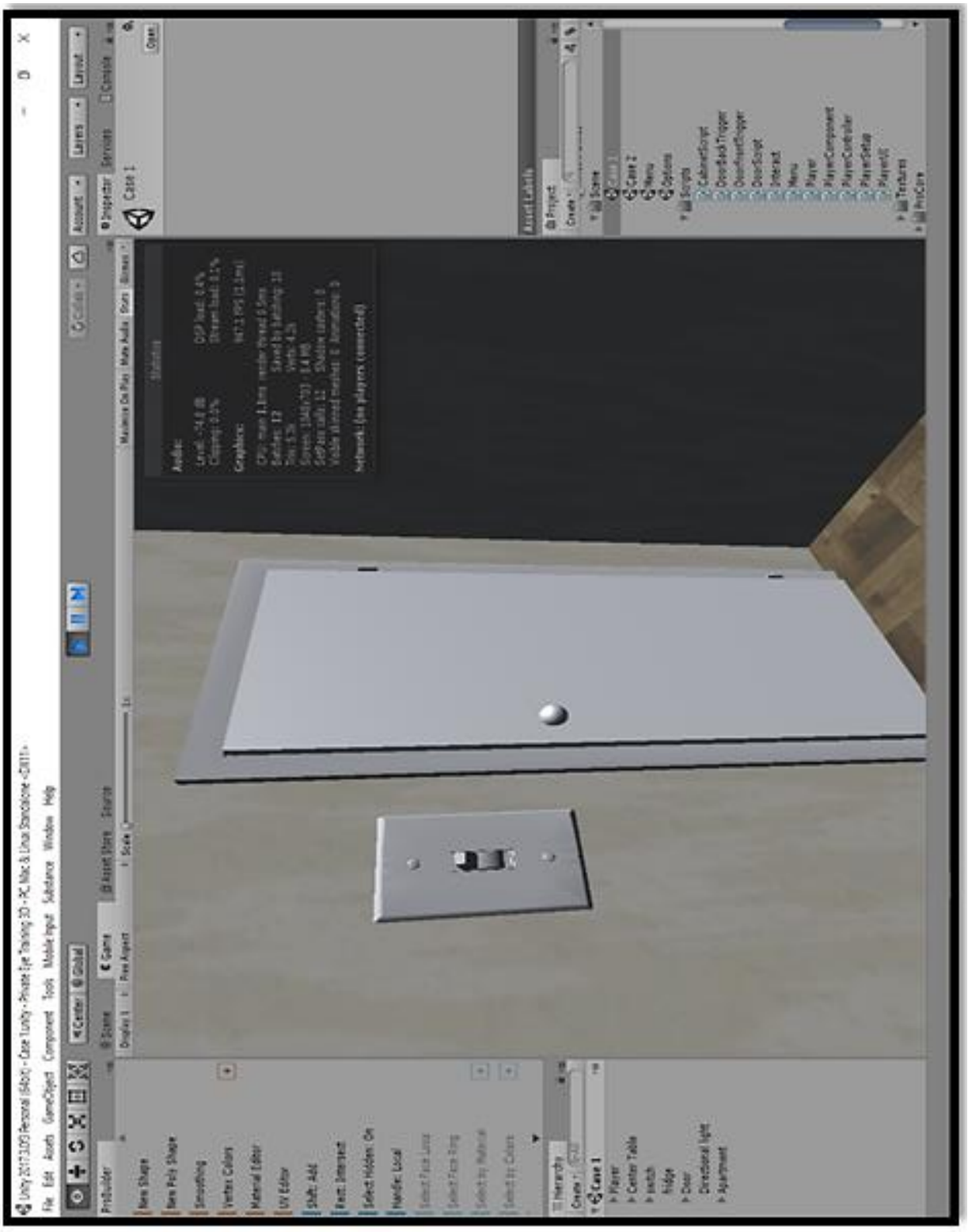

Figure 3. Game interaction

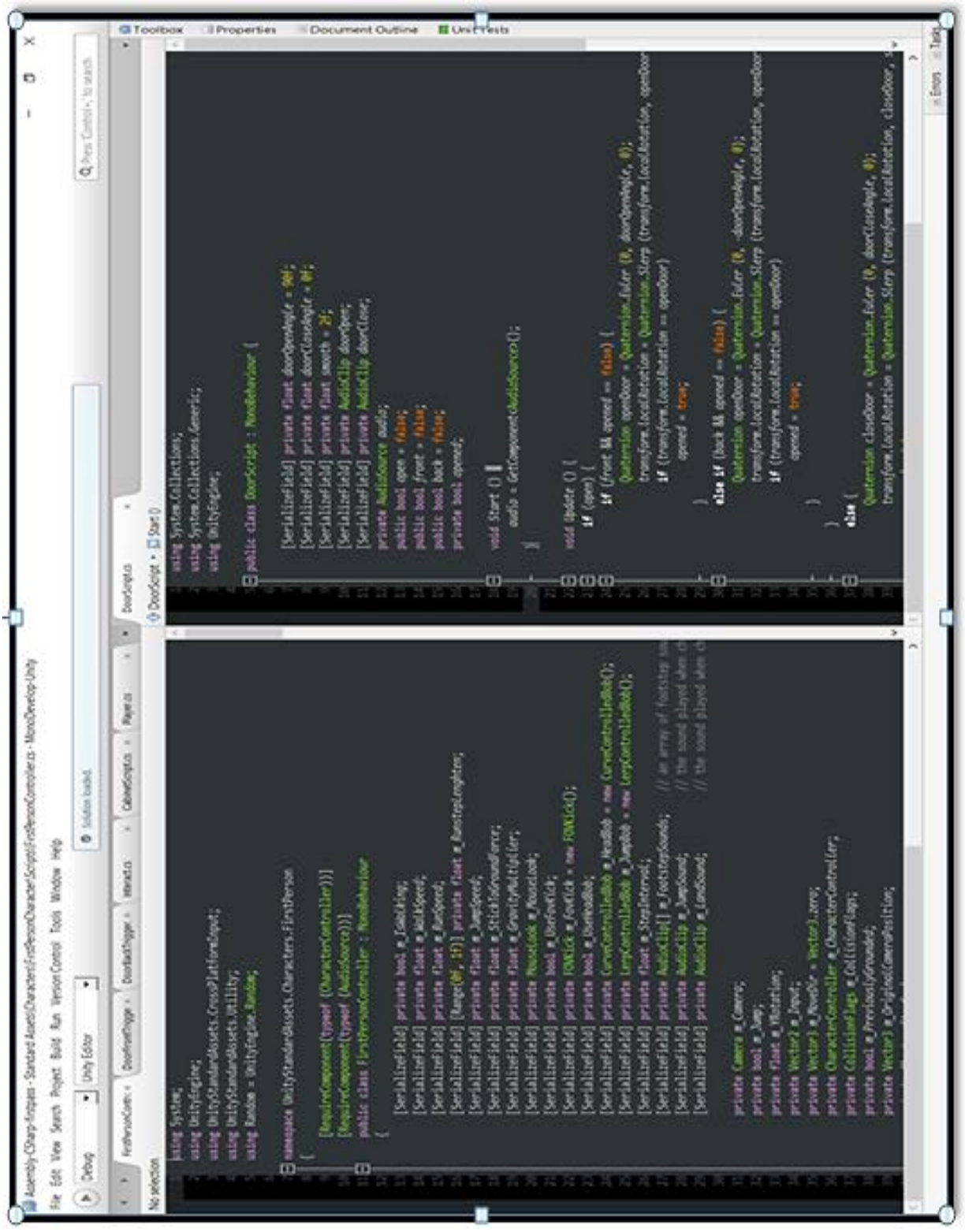

Figure 4. Code and movements of the character

3. TECHNICAL BACKGROUND

The proponents used Unity2017.1.1f1 and the language used is C#. The Private Eye Gameplay is a mysterious puzzle game that can boggle the user's mind and has a unique puzzle. The proponents used ISO 9126 to evaluate the game in terms of functionality, reliability, usability, efficiency, maintainability, and portability.

By observing, gathered evidence of actual behaviors in an everyday living of some players rather than reported behaviors. This allows the researchers to test assumptions about games and another genre of the game.

The proponents primarily gather all the materials and information needed for developing the game Private Eye Gameplay. After the game has been developed, the proponents evaluated the game entitled Private Eye based on the standards of ISO 9126.The standards are allocated to the game for assessment. The evaluation is categorized into six characteristics (factors) which are subdivided into sub-characteristics (criteria). The characteristics are manifested externally when the software is used as a consequence of internal software attributes. With the use of these criteria, the proponents will be able to identify if the game application meets the desired standard (Bartholow et al., 2008).

In Figure 5, the characteristics of reliability are maturity, fault-tolerance, recoverability, and compliance. Maturity is a sub-characteristic allows drawing conclusions about how mature software is. Fault-tolerance is sub-characteristic allows drawing conclusions about how fault-tolerant software is. Recoverability is sub-characteristic allows drawing conclusions about how well software recovers from software faults or infringement of its specified interface. Compliance is sub-characteristic allows drawing conclusions about how well software adheres to application-related standards, conventions, and regulations in laws and similar prescriptions (Brockmyer et al., 2009).

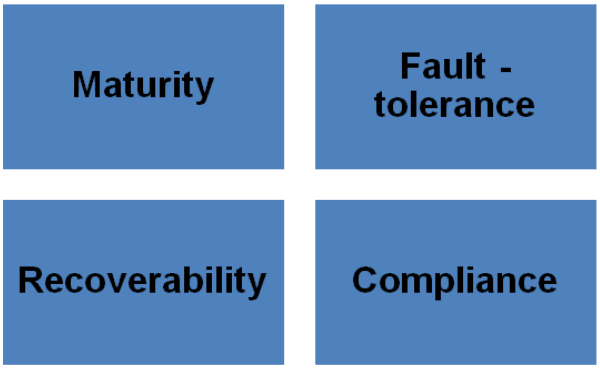

Figure 5. Characteristics of Reliability

In Figure 6, the characteristics of usability are understandability, learnability, operability, attractiveness, and compliance. Understandability is sub-characteristic allows drawing conclusions about how well users can recognize the logical concepts and applicability of software. Learnability is sub-characteristic allows drawing conclusions about how well users can learn the application of software (Hefner et al., 2007). Operability is sub-characteristic allows drawing conclusions about how well users can operate the software.

Attractiveness is sub-characteristic allows drawing conclusions about how attractive software is to the user. Compliance is sub-characteristic allows drawing conclusions about how well software adheres to application-related standards, conventions, and regulations in laws and similar prescriptions (Ferguson, 2007).

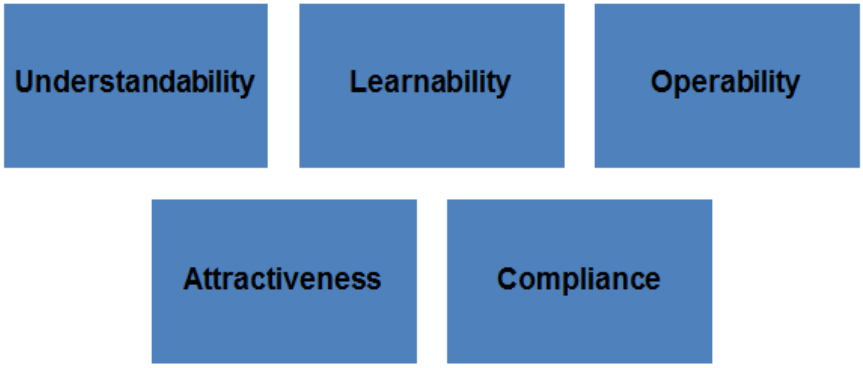

Figure 6. Characteristics of Usability

In Figure 7, the characteristics of efficiency are time behavior, resource utilization, and compliance. Time Behavior is sub-characteristic allows drawing conclusions about how well the time behavior of software is for a particular purpose. Resource utilization is sub-characteristic allows drawing conclusions about the number of resources utilized by the software. Resource utilization is sub-characteristic allows drawing conclusions about the number of resources utilized by the software (Durkin et al., 2002).

Figure 7. Characteristics of Efficiency

In Figure 8, the characteristics of maintainability are analyzability, changeability, stability, testability, and compliance. Analyzability is sub-characteristic allows drawing conclusions about how well software can be analyzed (Egli et al., 1984). Changeability is sub-characteristic allows drawing conclusions about how well software can be changed. Stability is sub-characteristic allows drawing conclusions about how stable software is. Testability is sub-characteristic allows drawing conclusions about how well software can be tested and is tested. Compliance is sub-characteristic allows drawing conclusions about how

well software adheres to application-related standards, conventions, and regulations in laws and similar prescriptions (Anderson, 2004).

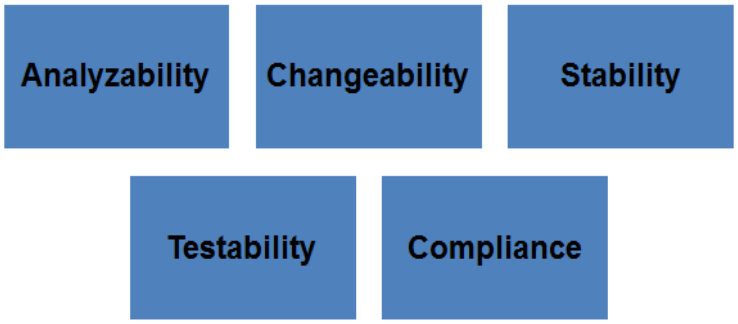

Figure 8. Characteristics of Maintainability

In Figure 9, the characteristics of portability are adaptability, installability, co-existence, replaceability, and compliance. Adaptability is sub-characteristic allows drawing conclusions about how well software can be adapted to environmental change (Nogueira et al., 2013). Installability is sub-characteristic allows drawing conclusions about how well software can be installed in a designated environment. Replaceability is a sub-characteristic allows drawing conclusions about how well software can replace other software or parts of it (Drachen et al., 2013). Co-existence sub-characteristic allows drawing conclusions about how well software can co-exist with other software products in the same operational environment. Compliance is sub-characteristic allows drawing conclusions about how well software adheres to application-related standards, conventions, and regulations in laws and similar prescriptions (Dye et al., 2009).

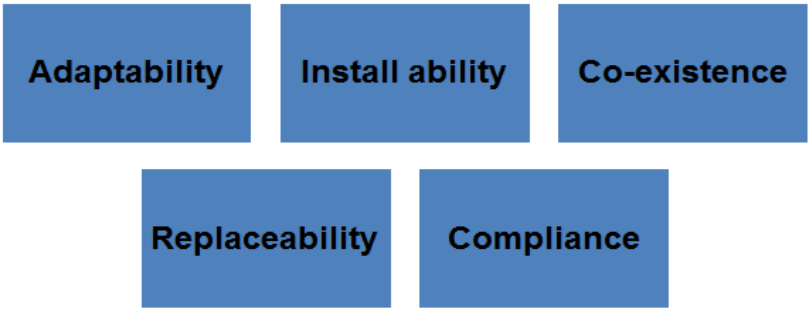

Figure 9. Characteristics of Portability

Figure 10 shows the game controller composed of the transform, character controller, first person controller (script), the rigid body and lastly the audio source

Figure 11 shows the C# language that was used in Unity. This code is for the interaction of the character.

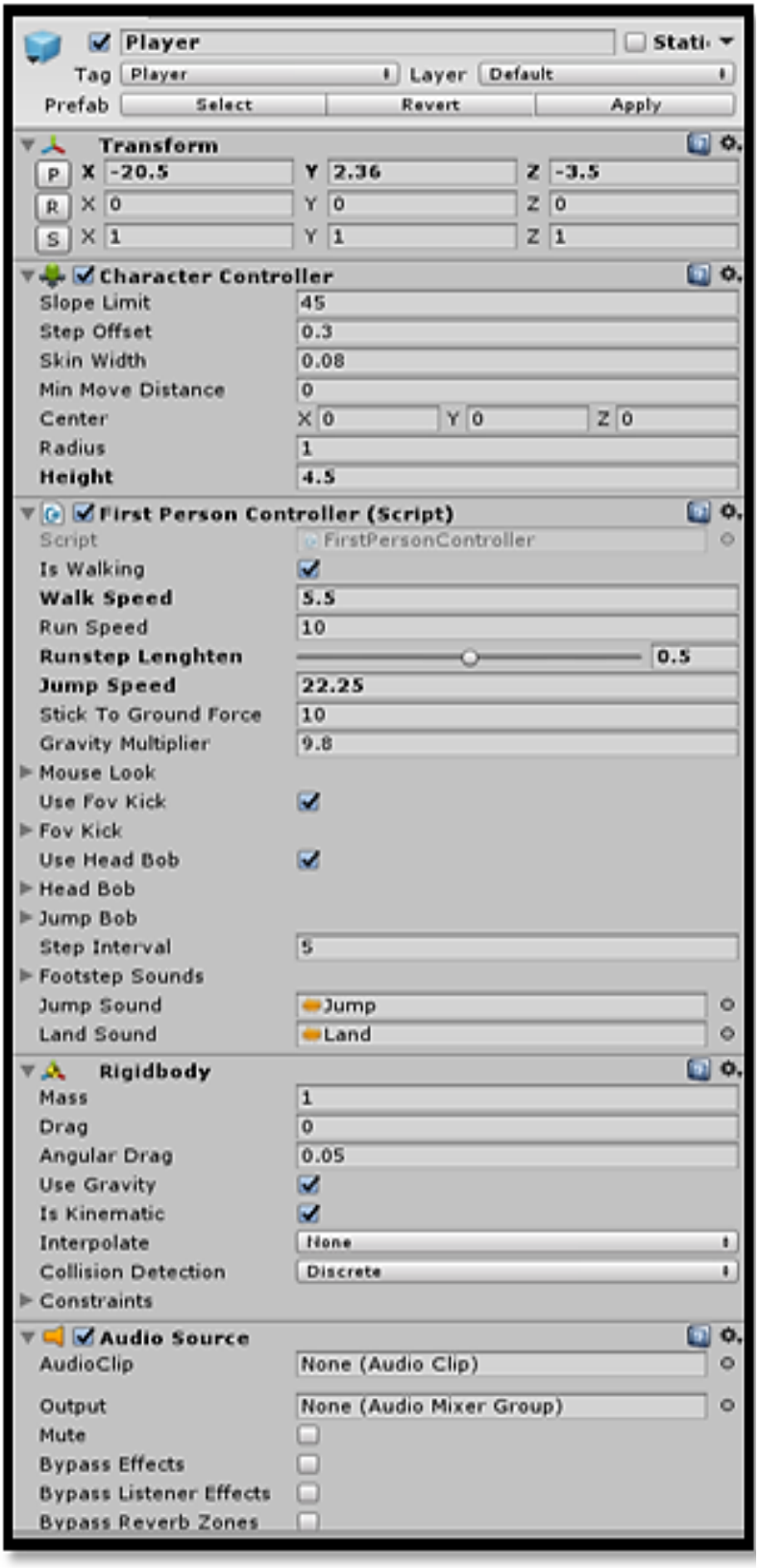

Figure 10. Game Controller

Figure 11. C#Code

Figure 12 shows the statistics that the proponents used in achieving a good quality sound of the game. Audio has four sub-characteristics. These are the level, clipping, DSP load and stream load (Geurtsen et al., 2015). Graphics has nine sub-characteristics. These are the CPU, batches, tri, screen, set pass calls, visible skinned meshes, saved by batching, verts, shadow cast, and animations (El Nasr, 2016). The "Network: (no players connected)" meaning that there are no players connects to the server.

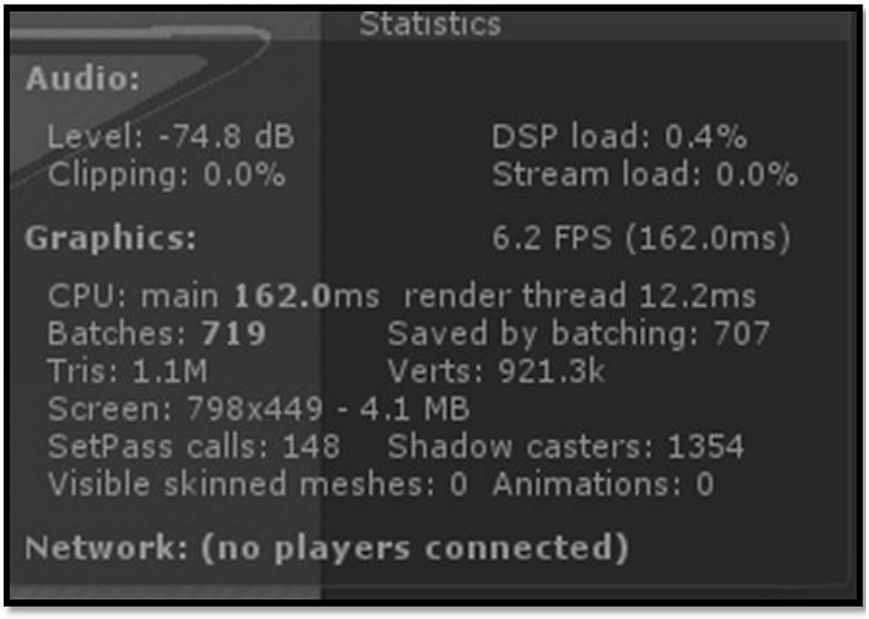

Figure 12. Proponents of the Game

3. 1. The Scope of the Project

The story of the Private Eye Gameplay is about Mike Doyle, a detective who is dedicated to his work. Roy Ronald, the Chief of Mike is envied him for being a good detective and with that evilness, the game will start. If the player starts the game dialogue, the conversation of Detective Mike and Chief Roy Ronald will appear and after the conversation, the case will start. The controllers of the game are mouse to see around and spacebar to jump, to walk use Letter W, A, S, D and with sprint use shift while holding the letters (W, A, S, D) and lastly to interact objects press Letter E. The game is played only in Laptop and Computer Windows 7 (64-bit) and up. The video card must have approximate memory 2MB and the processor must have 1-3 Hertz. Ram must have 4GB and above

4. METHODOLOGY

The specifications of this study are operational feasibility, technical feasibility, schedule feasibility, economic feasibility, requirements modeling.

Figure 13 shows puzzle game defects and it is placed in the head of the fish. The categories are people, design, and environment. These are placed in the backbone of the fish.

The people causes are lack of understanding, lack of patience and level of standard. The design cause is not recognizable and vague is measured in terms of sounds. The environment causes are the plodding simulation, dull storyline and complexity of the game. All the causes are put in an appropriate category.

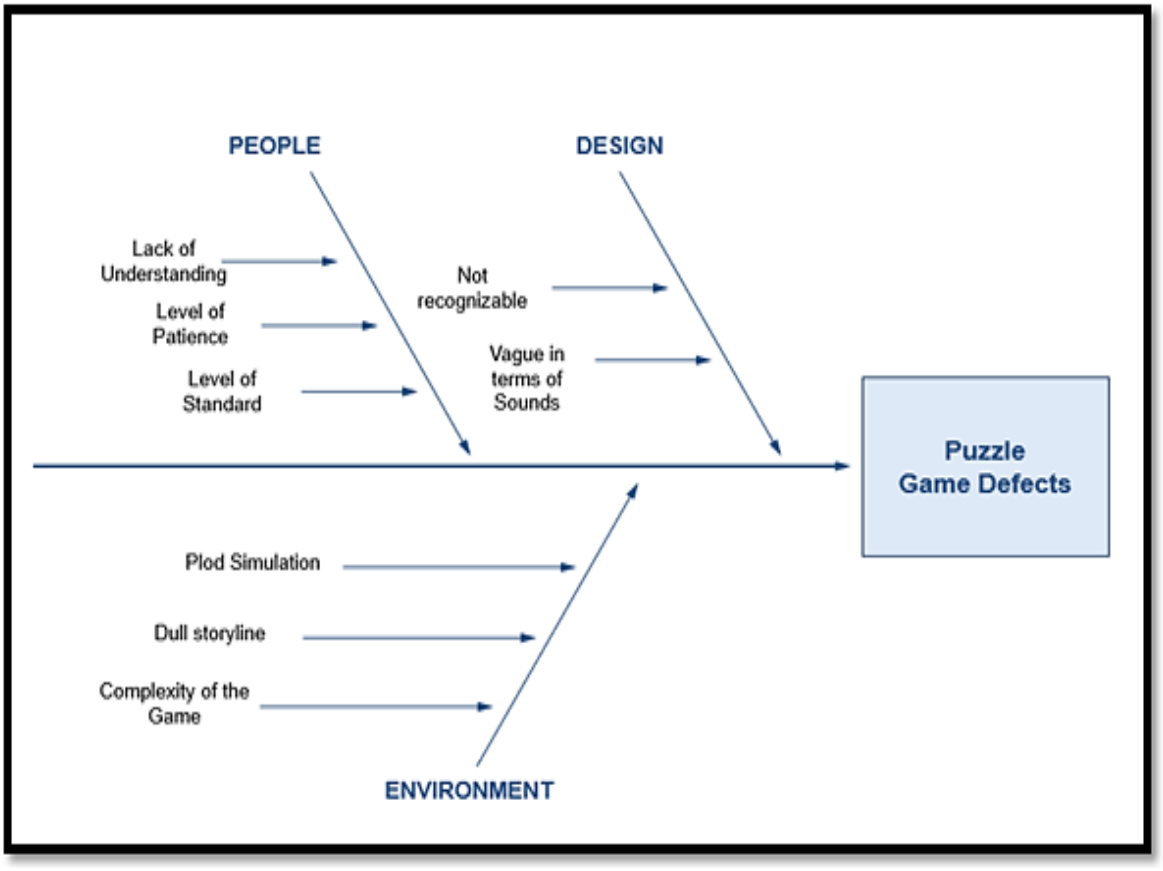

Figure 13. Causes of Puzzle Game Defects

Functional decomposition corresponds to the various functional relationships as how the original complex business function was developed. It mainly focuses on how the overall functionality is developed and its interaction between various components (Sasao et al., 1993).

Large or complex functionalities are more easily understood when broken down into pieces using functional decomposition. Functional Decomposition Diagram of the game is shown in Figure 14.

If the user starts to play the game entitled Private Eye, the story will appear and the puzzle will start into two cases. The first one is murder and the second is kidnapped.

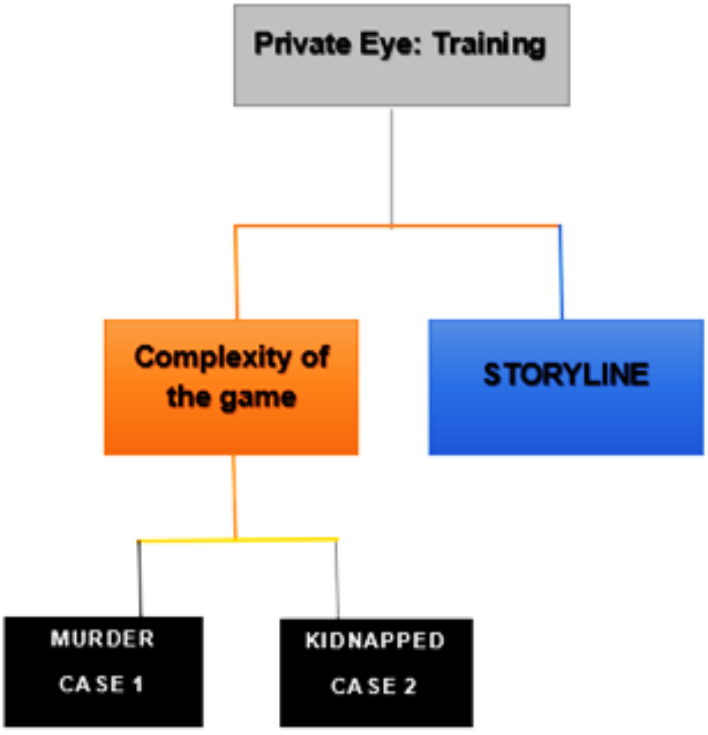

Figure 14. Functional Decomposition Diagram

4. 1. Technical Feasibility

Figure 15 shows the unity requirements. These are OS, CPU, and GPU. OS needs to be Windows 7 SP1 +,8,10,64-bit versions only and MAC OS X 10.9+ Server versions of Windows & OS are not tested. CPU needs to be SSE2 instruction set support. GPU must have graphics card with DX10 (shader model 4.0) capabilities. The rest mostly depends on the complexity of projects.

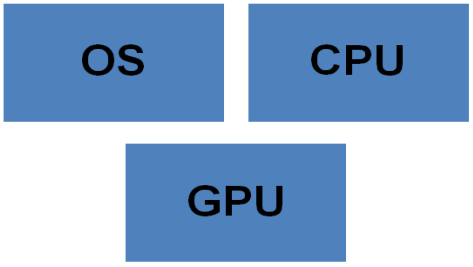

Figure 15. Unity System Requirements

In Figure 16, the operating system of the proponents is windows 10 Home Single Language 64-bit. The processor is Intel® Core™ i3-6006U CPU @ 2.00GHz (4 CPUs), 2.0GHz. The memory of the proponents used 4096MB RAM. DirectX Version is DirectX 12.

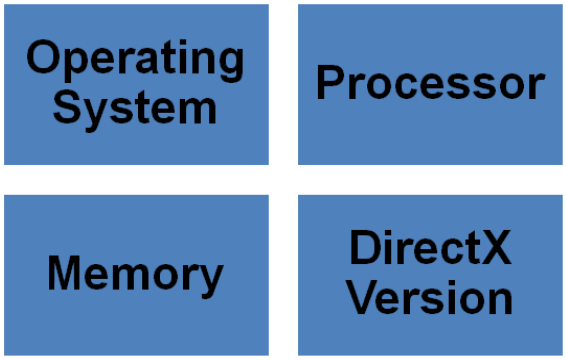

Figure 16. System Information

Importance of computers and laptops in modern times in the modern times, laptops and computers has become the most used items in the human world. Be it home, office, college, shop or any other commercial unit, these devices are in frequent use for multiple purposes. These devices play vital role in people everyday life and are of great help. They store huge amount of data, are easy to use and handle, and perform many functions that have made the life of students and professionals easy. However, it is not feasible for everyone to buy a new computer or laptop.

The proponents used the Scrum methodology in developing the game Private Eye Gameplay. Scrum can be implemented at the beginning or in the middle of the project. Agile has interrelated practices and rules that optimize the development environment, reduces organizational overhead and closely synchronized market requirements. Scrum causes the best software to be constructed given the available resources and acceptable time. Useful game functionality is the instant update of every transaction to the players as requirements, architecture, and design using the common and used technologies.

Scrum is an agile project management methodology or framework used primarily for software development projects with the goal of delivering new software capability every 2-4 weeks (Buttussi et al., 2007). It is one of the approaches that influenced the Agile Manifesto, which articulates a set of values and principles to guide decisions on how to develop higher-quality software faster (Hoover et al., 2015)

In Figure 17, scrum framework divides the product development into iterations known as "Sprints" which are time-boxed to a fixed length of 1 – 4 weeks (Duchting et al.,2007). Iteration should attempt to build a potentially shippable (properly tested) product increment. The time duration of the sprint and duration it lasts is decided by the team based on their requirements and capabilities. The sprint duration once finalized should not be modified. Product increment is at the end of every sprint the development team delivers a potentially shippable product that is tested and works well. Since sprints are short on durations, only important features would be developed first. This also gives customers a product that has the basic features that they can test and provide feedback on.

Product Backlog is the set of all requirements broken down into small work items and prioritized into a list is called product backlog (Canossa, 2013). The product backlog emerges over a period of time. Refinement of the product backlog is done during the Product Backlog refinement meeting. Product Backlog is the requirements for the project and these requirements may be refined during each Sprint. Sprint Backlog contains all the known User

Stories (or requirements) in the current Sprint (Ogan et al., 2010). The requirement with top priority listed first. The team pulls the stories into the Sprint and work collectively. Sprint Backlog is basically a list of all the requirements that need to be completed during the Sprint ordered by priority (Torok et al., 2015). User Stories is detailed requirements in agile software development are captured in the form of User Stories, from the point of view of the user rather than the organization or project. It captures who, what and why of the requirement from the user perspective (Ryan, 2009). User stories are short and concise statements. They are recorded on sticky notes, index cards etc. so that they can be stuck on walls or tables to be rearranged or used during the discussion.

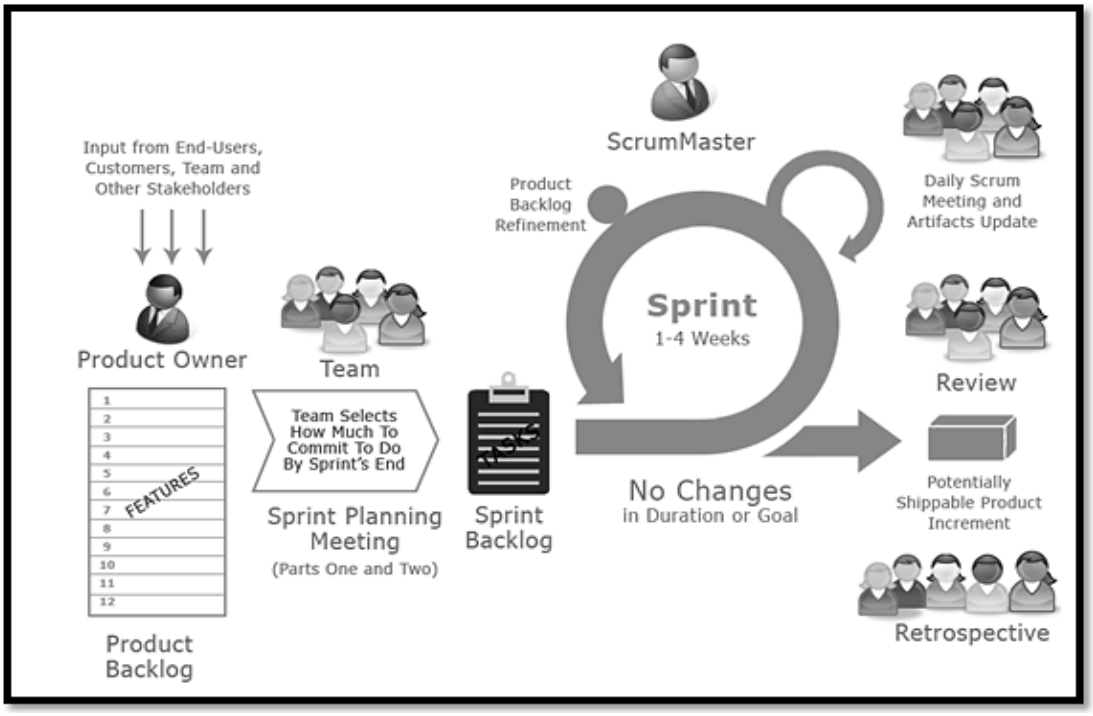

Figure 17. Scrum Methodology

Definition of done is a checklist of all exit criteria that must be completed by the team to call it done (Alimomeni et al., 2013). Definition of done exists at user story level, Sprint level, and release level. Time boxing is a concept of fixed time duration in which the team is expected to complete the committed features of work. Every ceremony in Scrum is time-boxed as per the recommendations are given in the Scrum guide. In Scrum methodology of agile software development, teams hold a daily planning meeting called the "Daily Scrum Meeting" or "Scrum Meeting" or "Stand-up meeting" (Schwaber, 1997).

This research aims to evaluate the game in terms of functionality, reliability, usability, maintainability, efficiency, and portability on the basis of empirical data. To this end, the proponents conducted questionnaire to evaluate their own game entitled Private Eye Gameplay. Agreeableness's is typically measured using self-report questionnaires, where a person is asked to review an adjective or statement and then report the degree to which it describes their personality on a Likert scale (Bevan, 1995).

The use of such questions to assess agreeableness allows a researcher to provide answers which provide a greater insight into their personality than the observation of his or her behavior might. However, questionnaires are also required researchers to be honest when providing answers. The questionnaires are allocated to the proponents for assessment. With the use of Likert Scale in Table 1, the researchers will be able to identify if the game application meets the desired function.

Table 1. Likert Scale

Scale	Verbal Interpretation
4.21-5.00	Strongly Agree
3.41-4.20	Agree
2.61-3.40	Moderately Agree
1.81-2.60	Disagree
1.00-1.80	Strongly Disagree

4. 2. Schedule Feasibility

A project will fail if it takes too long to be completed before it is useful. Typically, this means estimating how long the system will take to develop, and if it can be completed in a given time period using some methods like payback period. Schedule feasibility is a measure of how reasonable the project timetable is.

4. 3. Requirement Modelling

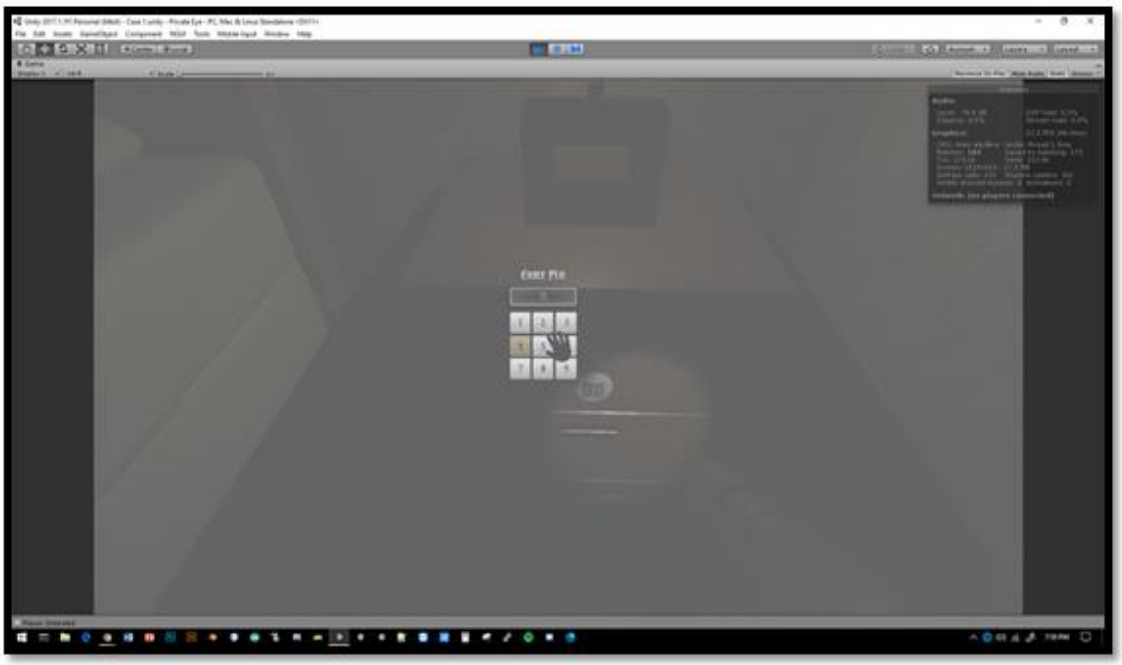

Figure 18. The Input of the Game

In Figure 18, the player must find the clues in order to know the pin, if the pin is right the player must proceed to the next level.

In Figure 19, the image shows that after typing the right pin while the player will wait to the next level.

Figure 19. The Output of the Game

4. 4. Requirement Modelling

In Table 2, the hazard is the software that the proponents used. The risk is the problem that the player encountered while control measure is the solution to the risk. The emergency action is to install the corresponding requirements that are needed.

Table 2. Risk Assessment and Analysis

Hazard	Risk	Control measures	Emergency action
Unity	Corrupted file	Check if the requirements of Unity is pass to the computer system of the proponents	Install the requirements of unity and the computer system if the requirements are compatible

4. 5. Output and User Interface Design

Figure 20 shows the image that the player is finding the clue to type the right pin. If the players find the clue he/she must find the lock and the player should enter the pin and if the pin is right, the player must proceed to the next level shown in Figure 21.

Figure 20. Output Interface Design

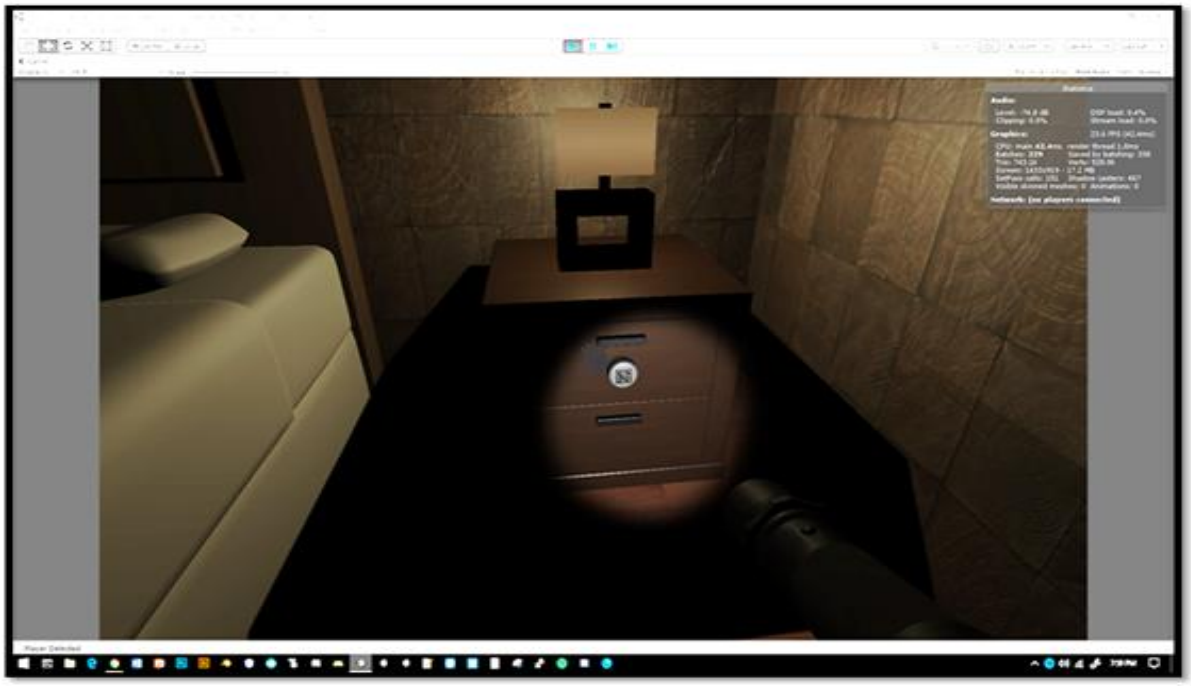

Figure 21. User Interface Design

In Figure 22, the player plays the Private Eye and the story begins. The puzzle appears. The puzzle is divided into two Murder and Kidnapped

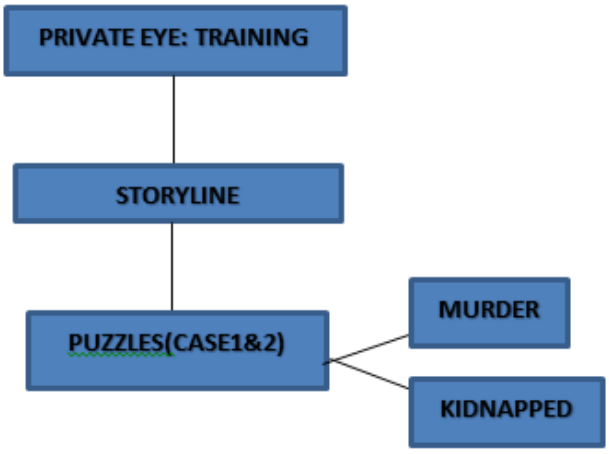

Figure 22. Network Model

4. 6. Software Specification

In Figure 23, the processor is Intel ® Core ™ i3-6006u CPU @ 2.00Ghz and the system type is 64 – bit Operating System

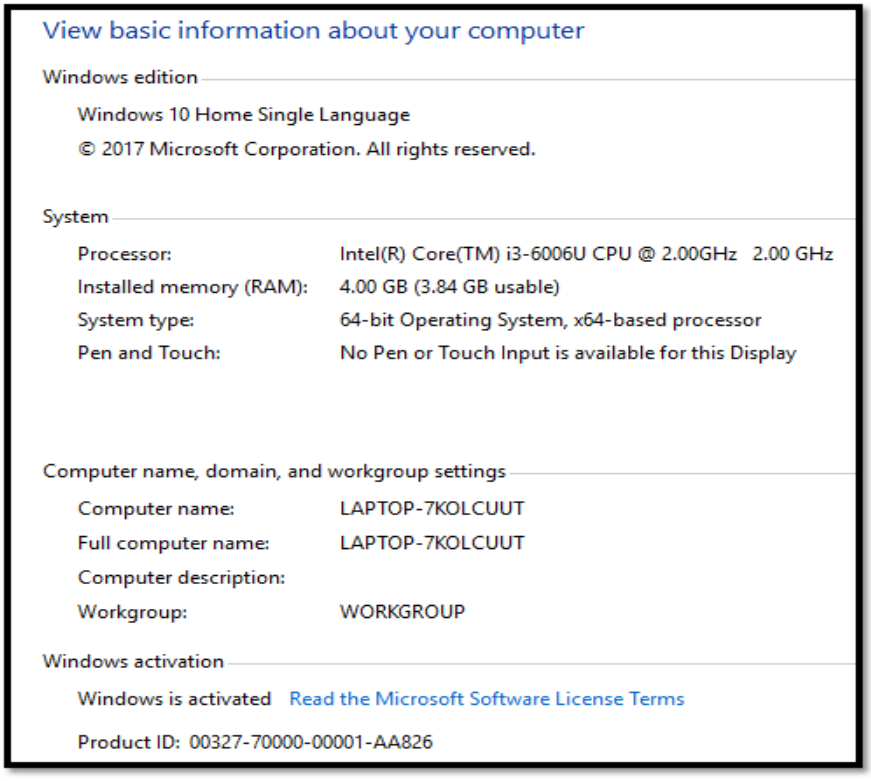

Figure 23. Basic Information of Computer

In Figure 24, the user used the specific requirements that require in unity in order to run the software without error.

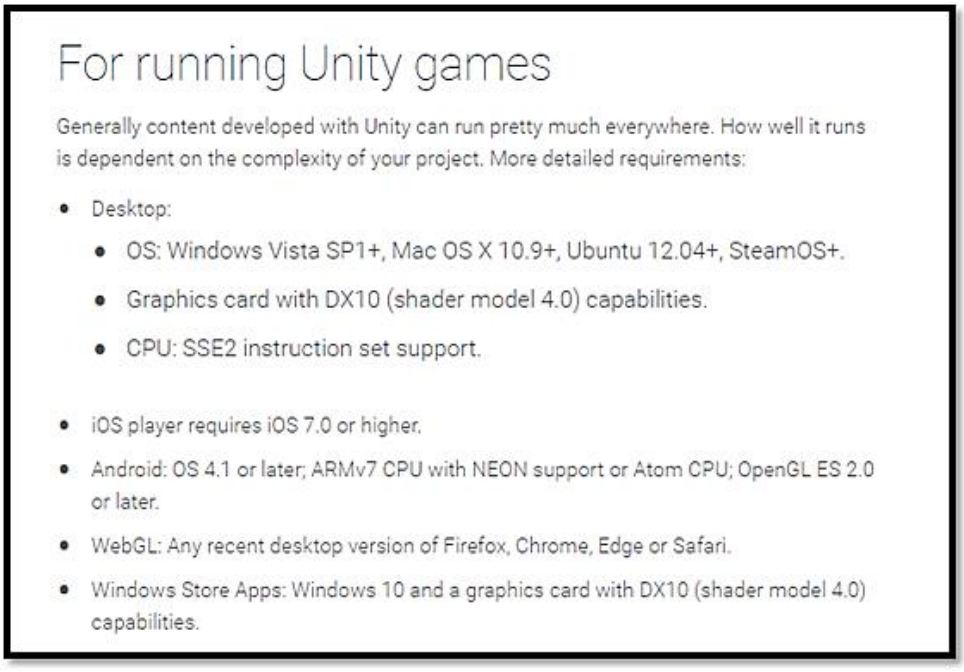

Figure 24. Running Unity Games

In Figure 25, the image shows that the player is solving the case 1 puzzle.

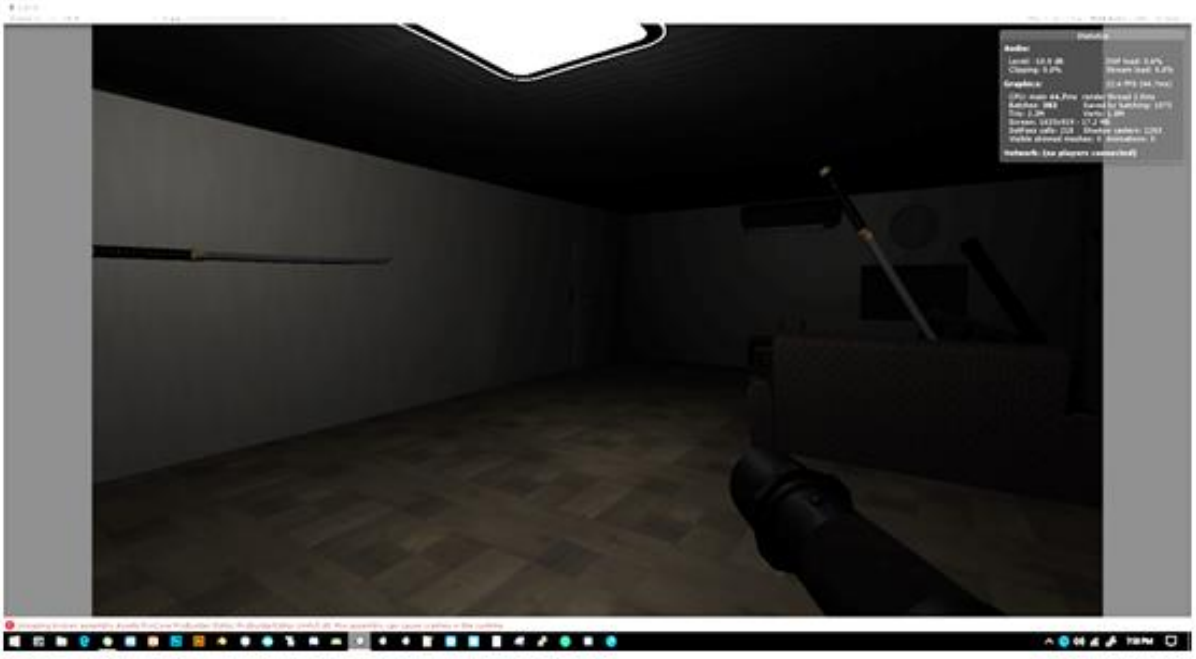

Figure 25. Front-end

In Figure 26, the language that the proponent used was C#. This code is the interaction between the things.

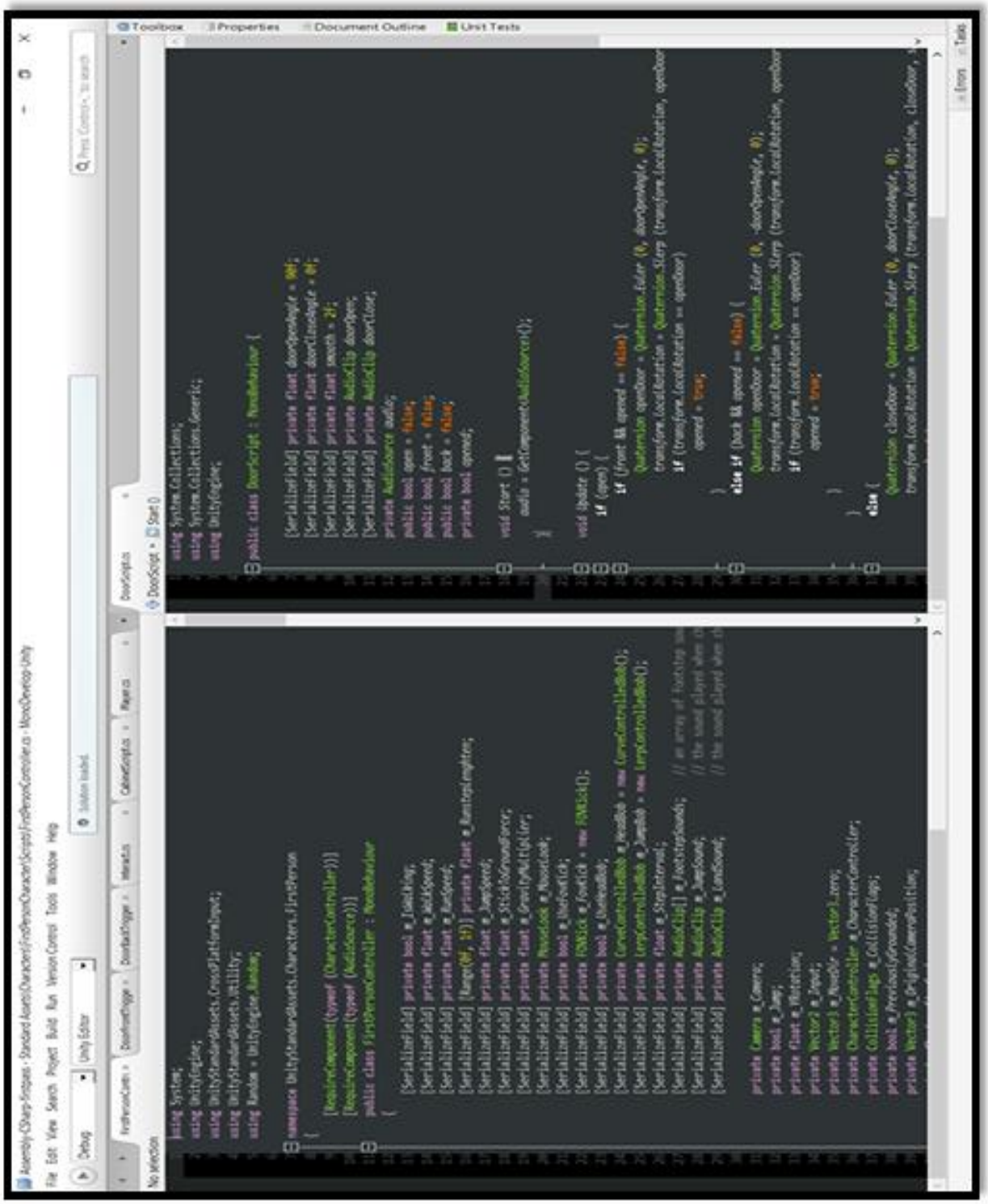

Figure 26. Back-end

5. RESULTS AND DISCUSSIONS

The researchers evaluated the game based on the Software Quality ISO Standard and with the sub-characteristics. The evaluation is divided into 5 categories: Strongly Agree (SA), Agree (A), Moderately Agree (MA), Disagree (D), and Strongly Disagree (SD).

5. 1. Mean Responses of Researchers on Level of Agreeableness by Functionality

Table 3 shows that the respondents' perception on the level of agreeableness in terms of functionality has a mean of 4.73 verbally interpreted as "Strongly Agree". This reveals that the application provides ease in accessing data as assessed by the proponents. It implies that the system is easily accessible.

Table 3. Level of Agreeableness by Functionality

Questions	Scale	Verbal Interpretation
1. Can software perform the tasks required?	5.00	SA
2. Can the system interact with another system?	4.21	SA
3. How accurate in terms of sounds?	5.00	SA
Mean	4.73	SA

5. 2. Mean Responses of Researchers on Level of Agreeableness by Efficiency

Table 4 shows that the respondents' perception on the level of agreeableness in terms of efficiency has a mean of 4.76 verbally interpreted as "Strongly Agree". This implies that the Private Eye Gameplay was easy to use in terms of efficiency. Generally, the user could easily follow the commands or instructions provided by the application.

Table 4. Level of Agreeableness by Efficiency

Questions	Scale	Verbal Interpretation
1. Is the game working properly?	5.00	SA
2. Is the storyline efficient?	4.30	SA
3. Is the sound of the game systematic?	5.00	SA
Mean	4.76	SA

5.3. Mean Responses of Researchers on Level of Agreeableness by Maintainability

Table 5 shows that the respondents' perception on the level of agreeableness in terms of maintainability has a mean of 4.73 verbally interpreted as "Strongly Agree". The data shows that the game is measured the ease and speed where the game is restored to operational status after a failure occurs.

Table 5. Level of Agreeableness by Maintainability

Questions	Scale	Verbal Interpretation
1. Can the software be easily modified?	5.00	SA
2. Can the software be tested easily?	5.00	SA
3. Can faults be easily diagnosed?	4.20	SA
Mean	4.73	SA

5. 4. Mean Responses of Researchers on Level of Agreeableness by Usability

Table 6 shows that the respondents' perception on the level of agreeableness in terms of usability has a mean of 5.00 verbally interpreted as "Strongly Agree". This means that the application performs its functions precisely and continuously works without errors.

Table 6. Level of Agreeableness by Usability

Questions	Scale	Verbal Interpretation
1. Can the user learn to use the system easily?	5.00	SA
2. Does the interface look good?	5.00	SA
3. Can the user use the system without much effort?	5.00	SA
Mean	5.00	SA

5. 5. Mean Responses of Researchers on Level of Agreeableness by Portability

Table 7 shows that the respondents' perception on the level of agreeableness in terms of portability has a mean of 4.33 verbally interpreted as "Strongly Agree". This means that the software complies with portable standards.

Table 7. Level of Agreeableness by Portability

Questions	Scale	Verbal Interpretation
1. Can the software be moved to other environments?	5.00	SA
2. Can the software be installed easily?	4.00	A
3. Does the software comply with portability standards?	4.00	A
Mean	4.33	SA

5. 6. Mean Responses of Researchers on Level of Agreeableness by Reliability

Table 8 shows that the respondents' perception on the level of agreeableness in terms of reliability has a mean of 4.76 verbally interpreted as "Strongly Agree". This implies that the proponents can easily generate the function of the application. Generally, the application met the desired functions under stated conditions for a specified period.

Table 8. Level of Agreeableness by Reliability

Questions	Scale	Verbal Interpretation
1. Is the game free from falsified information?	4.30	SA
2. Is the game free from any errors?	5.00	SA
3. Is the game has a good sound?	5.00	SA
Mean	4.76	SA

6. CONCLUSIONS

The results of the evaluation showed that the Private Eye Gameplay was easy to use and run using the available equipment in CISCO. Further, the students claimed that the game contributed to increased learning and motivation. The users also claimed that they would more likely to attend to more lectures in animations if Private Eye Gameplay was used. More work needs to be done to evaluate how regular usage of Private Eye Gameplay will affect the students' motivation and how much they learn. The main benefit of using this gameplay in animation course is to provide a fun way for students and improve their critical thinking and logical thinking.

Game usability and mobility problems are quite easy to identify since the procedure is very similar to the evaluation of utility software. Playability problems related to gameplay are more difficult to identify, but gameplay heuristics helped evaluators to focus on important aspects of the gameplay.

References

[1] Alimomeni, M., Safavi-Naini, R., & Sharifian, S. (2013, November). A true random generator using human gameplay. In *International Conference on Decision and Game Theory for Security* (pp. 10-28). Springer, Cham.

[2] Aloupis, G., Demaine, E. D., Guo, A., & Viglietta, G. (2015). Classic Nintendo games are (computationally) hard. *Theoretical Computer Science*, *586*, 135-160.

[3] Anderson, C. A. (2004). An update on the effects of playing violent video games. *Journal of Adolescence*, *27*(1), 113-122.

[4] Aoyama, Y., & Izushi, H. (2003). Hardware gimmick or cultural innovation? Technological, cultural, and social foundations of the Japanese video game industry. *Research Policy*, *32*(3), 423-444.

[5] Bang, M., Torstensson, C., & Katzeff, C. (2006, May). The powerhhouse: A persuasive computer game designed to raise awareness of domestic energy consumption. In *International conference on Persuasive Technology* (pp. 123-132). Springer, Berlin, Heidelberg.

[6] Bartholow, B. D., Bushman, B. J., & Sestir, M. A. (2006). Chronic violent video game exposure and desensitization to violence: Behavioral and event-related brain potential data. *Journal of Experimental Social Psychology*, *42*(4), 532-539.

[7] Behm-Morawitz, E., & Mastro, D. (2009). The effects of the sexualization of female video game characters on gender stereotyping and female self-concept. *Sex roles*, *61*(11-12), 808-823.

[8] Bevan, N. (1995). Measuring usability as quality of use. *Software Quality Journal*, *4*(2), 115-130.

[9] Brockmyer, J. H., Fox, C. M., Curtiss, K. A., McBroom, E., Burkhart, K. M., & Pidruzny, J. N. (2009). The development of the Game Engagement Questionnaire: A measure of engagement in video game-playing. *Journal of Experimental Social Psychology*, *45*(4), 624-634.

[10] Buttussi, F., Chittaro, L., Ranon, R., & Verona, A. (2007, June). Adaptation of graphics and gameplay in fitness games by exploiting motion and physiological sensors. In *International Symposium on Smart Graphics* (pp. 85-96). Springer, Berlin, Heidelberg.

[11] Cain, M. S., Landau, A. N., & Shimamura, A. P. (2012). Action video game experience reduces the cost of switching tasks. *Attention, Perception, & Psychophysics*, *74*(4), 641-647.

[12] Canossa, A. (2013). Meaning in gameplay: filtering variables, defining metrics, extracting features and creating models for gameplay analysis. In *Game Analytics* (pp. 255-283). Springer, London.

[13] Carnagey, N. L., Anderson, C. A., & Bushman, B. J. (2007). The effect of video game violence on physiological desensitization to real-life violence. *Journal of Experimental Social Psychology*, *43*(3), 489-496.

[14] Castel, A. D., Pratt, J., & Drummond, E. (2005). The effects of action video game experience on the time course of inhibition of return and the efficiency of visual search. *Acta Psychologica*, *119*(2), 217-230.

[15] Castellar, E. N., Van Looy, J., Szmalec, A., & De Marez, L. (2014). Improving arithmetic skills through gameplay: Assessment of the effectiveness of an educational game in terms of cognitive and affective learning outcomes. *Information Sciences*, *264*, 19-31.

[16] Chisholm, J. D., Hickey, C., Theeuwes, J., & Kingstone, A. (2010). Reduced attentional capture in action video game players. *Attention, Perception, & Psychophysics*, *72*(3), 667-671.

[17] Dickey, M. D. (2005). Engaging by design: How engagement strategies in popular computer and video games can inform instructional design. *Educational Technology Research and Development*, *53*(2), 67-83.

[18] Drachen, A., Canossa, A., & Sørensen, J. R. M. (2013). Gameplay metrics in game user research: Examples from the trenches. In *Game Analytics* (pp. 285-319). Springer, London

[19] Dromey, R. Geoff. "A model for software product quality." *IEEE Transactions on Software Engineering* 21, no. 2 (1995): 146-162.

[20] Düchting, M., Zimmermann, D., & Nebe, K. (2007, July). Incorporating user-centered requirement engineering into agile software development. In *International Conference on Human-Computer Interaction* (pp. 58-67). Springer, Berlin, Heidelberg.

[21] Durkin, K., & Barber, B. (2002). Not so doomed: Computer game play and positive adolescent development. *Journal of Applied Developmental Psychology*, *23*(4), 373-392.

[22] Dye, M. W., Green, C. S., & Bavelier, D. (2009). The development of attention skills in action video game players. *Neuropsychologia*, *47*(8-9), 1780-1789.

[23] Egli, E. A., & Meyers, L. S. (1984). The role of video game playing in adolescent life: Is there reason to be concerned?. *Bulletin of the psychonomic society*, *22*(4), 309-312.

[24] El-Nasr, M. S., Drachen, A., & Canossa, A. (2016). *Game analytics*. Springer London Limited.

[25] Ferguson, C. J. (2007). Evidence for publication bias in video game violence effects literature: A meta-analytic review. *Aggression and Violent behavior*, *12*(4), 470-482.

[26] Geurtsen, A., Lamers, M. H., & Schaaf, M. J. (2015, September). Interactive digital gameplay can lower stress hormone levels in home alone dogs—A case for animal welfare informatics. In *International Conference on Entertainment Computing* (pp. 238-251). Springer, Cham.

[27] Hefner, D., Klimmt, C., & Vorderer, P. (2007). Identification with the player character as determinant of video game enjoyment. In *Entertainment Computing–ICEC 2007* (pp. 39-48). Springer, Berlin, Heidelberg.

[28] Hoover, A. K., Cachia, W., Liapis, A., & Yannakakis, G. N. (2015, April). Audioinspace: Exploring the creative fusion of generative audio, visuals and gameplay. In *International Conference on Evolutionary and Biologically Inspired Music and Art* (pp. 101-112). Springer, Cham.

[29] Ke, F. (2008). A case study of computer gaming for math: Engaged learning from gameplay?. *Computers & Education*, *51*(4), 1609-1620.

[30] Lin, J. J., Mamykina, L., Lindtner, S., Delajoux, G., & Strub, H. B. (2006, September). Fish'n'Steps: Encouraging physical activity with an interactive computer game.

In *International conference on ubiquitous computing* (pp. 261-278). Springer, Berlin, Heidelberg.

[31] Lorant-Royer, S., Munch, C., Mesclé, H., & Lieury, A. (2010). Kawashima vs "Super Mario"! Should a game be serious in order to stimulate cognitive aptitudes?. *Revue Européenne de Psychologie Appliquée/European Review of Applied Psychology*, *60*(4), 221-232.

[32] Nogueira, P. A., Rodrigues, R., & Oliveira, E. (2013, September). Real-time psychophysiological emotional state estimation in digital gameplay scenarios. In *International Conference on Engineering Applications of Neural Networks*(pp. 243-252). Springer, Berlin, Heidelberg.

[33] O'Connor, R. V. (2012, May). Evaluating management sentiment towards ISO/IEC 29110 in very small software development companies. In *International Conference on Software Process Improvement and Capability Determination*(pp. 277-281). Springer, Berlin, Heidelberg.

[34] Ogan, A., Aleven, V., Kim, J., & Jones, C. (2010, June). Intercultural negotiation with virtual humans: The effect of social goals on gameplay and learning. In *International Conference on Intelligent Tutoring Systems* (pp. 174-183). Springer, Berlin, Heidelberg.

[35] Olsina, L., Covella, G., & Rossi, G. (2006). Web quality. In *Web Engineering* (pp. 109-142). Springer, Berlin, Heidelberg.

[36] Paulk, M. C. (1993). Comparing ISO 9001 and the capability maturity model for software. *Software Quality Journal*, *2*(4), 245-256.

[37] Ryan, R. M., Rigby, C. S., & Przybylski, A. (2006). The motivational pull of video games: A self-determination theory approach. *Motivation and emotion*, *30*(4), 344-360.

[38] Sasao, Tsutomu. "FPGA design by generalized functional decomposition." In *Logic synthesis and optimization*, pp. 233-258. Springer, Boston, MA, 1993

[39] Schwaber, K. (1997). Scrum development process. In *Business object design and implementation* (pp. 117-134). Springer, London.

[40] Shaker, N., Asteriadis, S., Yannakakis, G. N., & Karpouzis, K. (2011). A game-based corpus for analysing the interplay between game context and player experience. In *Affective Computing and Intelligent Interaction* (pp. 547-556). Springer, Berlin, Heidelberg.

[41] Torok, L., Pelegrino, M., Trevisan, D. G., Clua, E., & Montenegro, A. (2015, September). A mobile game controller adapted to the gameplay and user's behavior using machine learning. In *International Conference on Entertainment Computing* (pp. 3-16). Springer, Cham.

[42] Wagner, S. (2013). *Software product quality control* (pp. I-XII). Berlin: Springer.