TEPIAN Vol. 4 No. 1 (March 2023) p-ISSN 2721-5350 e-ISSN 2721-5369
doi.org/10.51967/tepian.v4i1.2232

Application of Collision Detection in Arjuna's Adventure Game

Pajar Pahruddin
STMIK Widya Cipta Dharma,
Samarinda, 75123, Indonesia
pajar@wicida.ac.id

Siti Lailiyah
STMIK Widya Cipta Dharma,
Samarinda, 75123, Indonesia
lail.59a@gmail.com

Dicky Ary Setiawan *
STMIK Widya Cipta Dharma,
Samarinda, 75123, Indonesia
dickyari96@gmail.com
**Corresponding author*

Submitted: 2023-02-23; Accepted: 2023-02-28; Published: 2023-03-01

***Abstract*—**This research makes how to build a 2 dimensional game with the theme of a young man named Arjun who is on the battlefield and is attacked by "Buto Ijo's troops and must go through attacks from Buto Ijo's troops to protect the "Banakeling" kingdom. The background in taking the theme Arjuna game with side scrolling game to make this game easy to play. In game research that was built using the stages of multimedia development and also using the collision detection method. The Arjuna's Adventure Game is built based on Android. The design system in this study uses a flowchart. Testing is done with the White Box and Black Box.

***Keywords*—**Collision Detection, Games, Arjuna

I. INTRODUCTION

Games are currently experiencing very rapid development, including in Indonesia. Today's games have been played by many people from young to old. Game types are increasingly varied in line with developments in hardware and software technology. The development of games is currently growing rapidly, currently they are not only played on computers, which can only be played with friends who are known, but now there are many games being played in cyberspace which are often referred to as online games which are played by many people in one game even though they are far apart and don't know each other.

Today's children prefer games so that traditional things such as folklore have begun to be abandoned or lost in children's lives. Therefore, a game with the theme of folklore will be made which is presented with the concept of a side scrolling game so that it can be played and understood by children. One method that is often used in game development is the collision detection method. Collision detection is a collision detection technique to find out what objects touch in a certain coordinate field.

In other words, all forms of activity that require thought, intellectual agility and achievement of certain targets can be said to be games. Side scrolling game is one of the game genres with a camera viewpoint as if it is next to the character, and follows the movement of the player who generally moves from the left side to the right side of the layer to reach the specified target or location.

Based on this background, the theme of this thesis is taken with the title Application of the Collision Detection Algorithm in the Arjuna's Adventure Game. And it is hoped that these games or games can entertain the players or make them a means of entertainment and can also be used as material for further research for students and can be further developed for local game developers.

II. LITERATURE REVIEW

A. *Games*

According to Putra (2017), the game (game) in general is a recreational activity with the aim of having fun, filling spare time, or light exercise. The game is usually done alone or together. Meanwhile, computer games are video games that are played on personal computers, and not on game consoles or ding-dong machines. Computer games have evolved from simple graphical systems to become complex and sophisticated.

B. *Game History*

According to Wibowo (2017) The beginnings of video games were at the beginning of the cathode ray tube based missile defense system in the late 1940s. These existing programs were then adapted into another simple game in the 1950s. In the late 1950's and through the 1960's. More computer games were developed (mostly on mainframe computers).
Gradually the level of sophistication and complexity also increases. (Rosa AS, M. Salahuddin. 2015)After this period, video games have diverged into multiple platforms such as arcades, mainframes, consoles, personal computers and later handheld games.

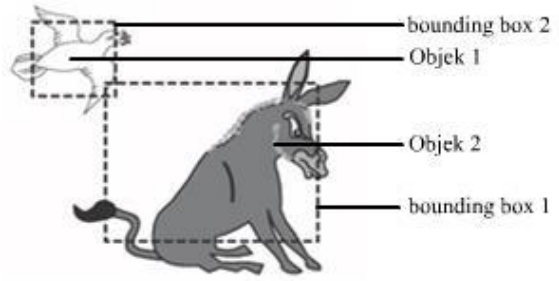
The first commercial video game console company was Computer Space which existed in 1971. Which was the foundation for the new entertainment industry in the late 1970s in the United States, Japan, and also Europe. However, this company did not last long. This was largely due to the flood of video game competitors coming to the market which resulted in a total collapse of the console game industry worldwide, and finally shifted market dominance from North America to Japan.(Istiyanto et al. 2013). But this only affects the console game market, the computer game market is largely unaffected. Although several attempts were made by North American and European companies, with the

Pahrudin, P., Lailiyah, S., & Setiawan , D. A. (2023). Application of Collision Detection in Arjuna's Adventure Game. TEPIAN, 4(1). 45-51.
https://doi.org/10.51967/tepian.v4i1.2232

fourth generation of video game consoles their efforts ultimately failed. (Ziveria et al. 2014),It wasn't until the sixth generation of video game consoles that a non-Japanese company released a commercially successful console system.(Subagio, Ariyadi 2014).

C. *Collision Detection*

According to Schwarzl, (2012), Collision detection is the process of detecting collisions between two objects. In fact, in the game itself collisions do not only occur between two objects, but can also occur between one object and many objects or many objects with many objects.(Mustaqbal et al. 2015). In games, accurate collision detection is needed because after determining whether a collision occurred or not, we must determine what happened to the object that was hit or hit. (Nafianto et al. 2017)Collision detection is also useful for determining the position of one object to another so that no objects penetrate each other. So that the game to be made has similarities with the existing reality can be seen in Picture 1.



Picture 1. Example of Reduced Size Bounding Box Collision Detection

The application of the reduced size bounding box Collision Detection algorithm in Picture 2 shows that if object 1 which has been inserted by bounding box 1 collides with object 2 which has been inserted by bounding box 2 then the Collision is true otherwise the collision is false. Below is an example script using the reduced size bounding box collision detection algorithm.

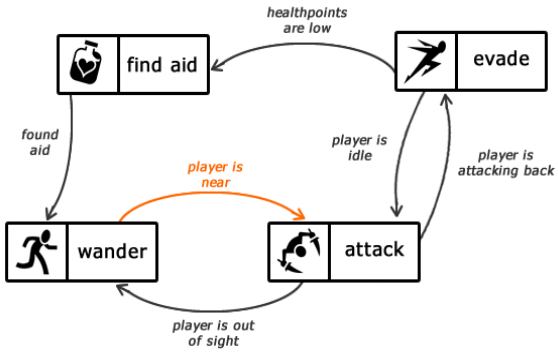
```
If(object1.boundingbox1.hitTes
t(object2.boundingbox2){collis
 ion = true;} else { collision
          = false; }
```

Picture 2. Example of a Collision Detection Script

D. *Finite State Machine*

The definition of a Finite State Machine (FSM) or often referred to as a Finite State Automata (FSA). According to Fernando Bevilacqua (2013), a finite state machine, or abbreviated FSM, is a computational model based on a hypothetical machine made of one or more states.

Only one state can be active at a time, so the machine must move from state to state in order to perform different actions. (Roedavan 2017)FSMs are typically used to organize and represent execution flows, which is useful for implementing AI in games. An enemy's "brain", for example, can be implemented using FSM: each state represents an action, such as attack or evasion**.**The FSM diagram representing the enemy's brain can be seen in Picture 3.

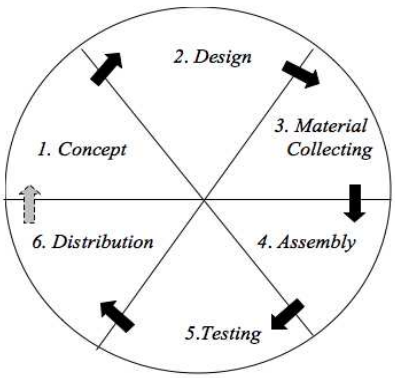


Picture 3. FSM Diagram Representing Enemy Brain

III. RESEARCH METHODS

A. *Multimedia Development Stages*

According to Iwan Binanto, (2010), the multimedia methodology consists of six stages, namely concept, design, material collecting, assembly, testing, and distribution. (Indrajani, 2011) These six stages cannot be exchanged. Even so, the concept stage should be the first thing to do. (Kevin 2018). So that multimedia development can be included in learning, it must go through well-designed and coherent stages so that the resulting multimedia products are of good quality and are appropriate for use in learning. The stages of system development can be seen in Picture 4.



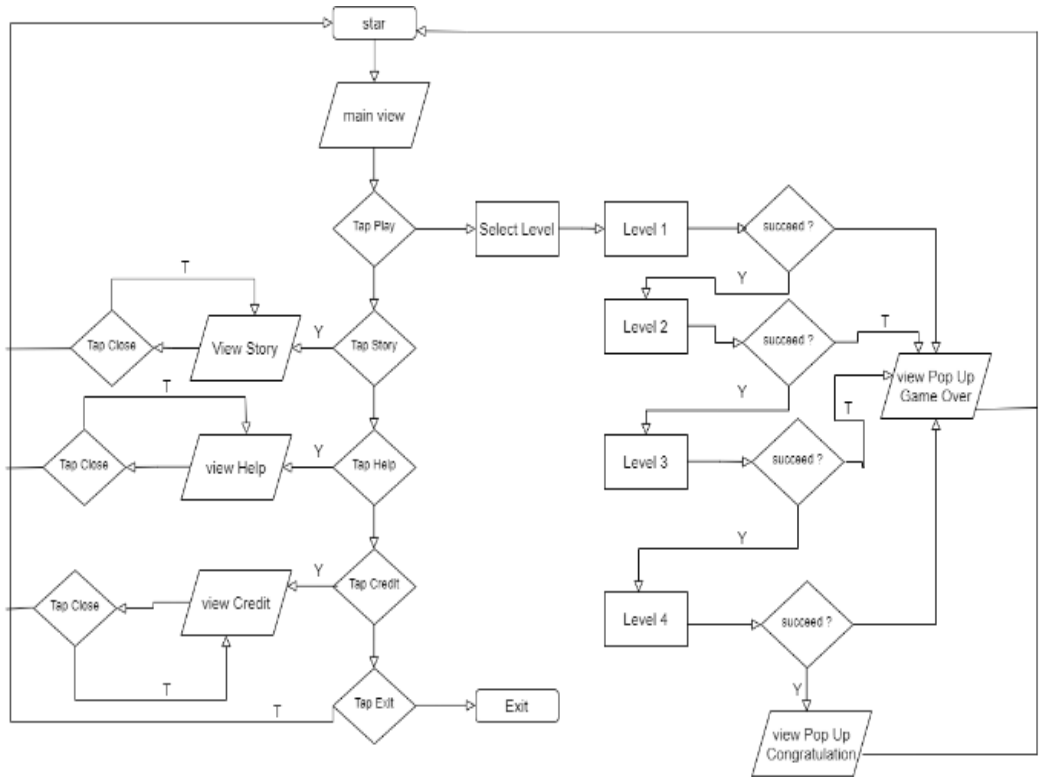
Picture 4. System Development Stages

B. *Game Flowchart of Arjuna Adventure*

In the initial appearance of the game "Arjuna's Adventure" displays the main menu in the game which consists of Play, Story, Help, About, Exit Game. If you select the Play menu, it will display Select Level, then the mission that must be completed at level 1. If level 1 successfully completes the mission it will proceed to the next mission but if the mission at level 1 is not fulfilled or the game is over then it will return to the start menu. (Asbudi 2015) Then there is a Story menu which contains

Pahrudin, P., Lailiyah, S., & Setiawan , D. A. (2023). Application of Collision Detection in Arjuna's Adventure Game. TEPIAN, 4(1). 45-51.
https://doi.org/10.51967/tepian.v4i1.2232

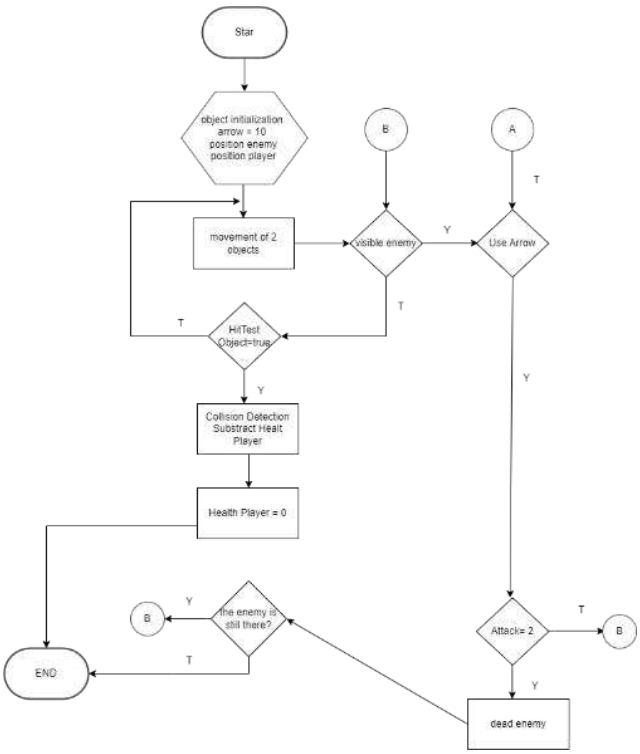
the story in this game. (Kuniadi et al. 2017) The Help Menu contains information on which buttons are used on the Smartphone Screen layer to play the game "Arjuna's Adventure". The Credit menu contains the profile of the creator of the game "Arjuna's Adventure". On the exit menu, the game will stop and exit the game can be seen in Picture 5.



Picture 5. Arjuna's Adventure Game Flowchart

C. Collision Detection Player and Enemy

The game play of Arjuna's Adventure requires a collision process between an object with other objects, Collision detection is implemented in the game Arjuna's Adventure so that it doesn't penetrate if one object collides with another object so that the collision process will be visible real, Collision detection begins by detecting the position of the player and the position of the "Buto Ijo" troop as an enemy. According to Erwin, & Purba, FR (2013). After the player objects move and collide with each other, the logic equation will detect whether the two objects they collide with each other or not. If a collision occurs, then collision detection will occur and continue programming flow to the next step subtract health.(Munir 2017) Likewise if players collide with other "Buto Ijo" troop objects, the collision detection method will be run again.(Hotijah, Siti 2018). In the flow shown in the image below there are several object initializations that will run, then there is the process of object movement and several decisions are seen to provide branch lines in each condition, starting from the movement of the player when he sees the "Buto Ijo" troops so they can attack arrows from a distance. (Haryono, 2015) If the player collides with the "Buto Ijo" troop, there is a reduction in the player's health. The Collision detection method for collisions between 2 objects can be seen in Picture 6.

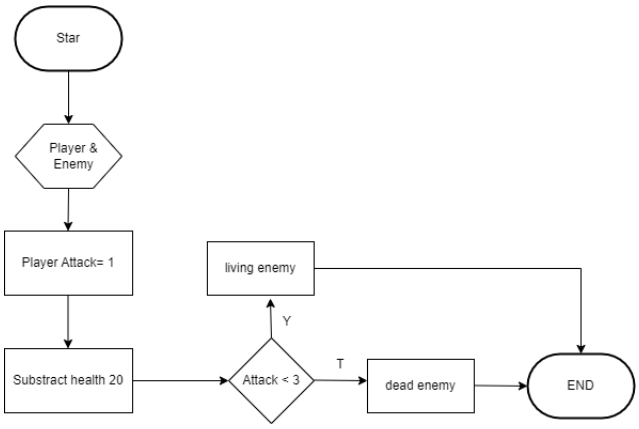


Picture 6. Collision Detection Player and Enemy

Pahrudin, P., Lailiyah, S., & Setiawan , D. A. (2023). Application of Collision Detection in Arjuna's Adventure Game. TEPIAN, 4(1). 45-51.
https://doi.org/10.51967/tepian.v4i1.2232

D. Flow chart Attack Enemy

The game play of Arjuna's Adventure requires a collision process between one object and another object, such as when the player encounters an enemy and an attack occurs on the player. Starting from the movement of the player. When he sees an enemy, the player shoots arrows.(Bevilacqua, Fernando, 2013), If the attack is less than 3. Then the enemy is still alive, if the attack is equal to 3 then the enemy is dead. That's how the collision of 2 objects can be seen, as seen in Picture 7.



Picture 7. Flowchart Attack Enemy

III. RESULTS AND DISCUSSION

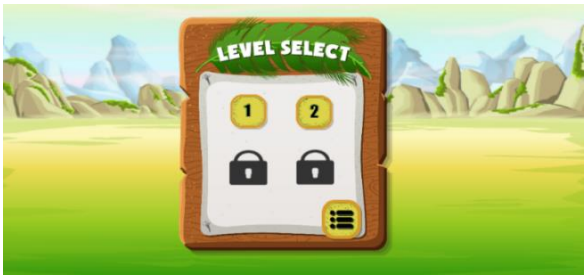
A. Home Menu Display

The menu is the main display of the "Arjuna's Adventure" game when new players enter the application. There is a button menu display from this game. Each button on the main menu has its own function. Respectively. When the player clicks/taps the Play button it will display the Level Select menu, and in the desired level/stage when the player clicks/taps the Story button it will display a menu to explain the story of this game. The help button will display the help menu, if the player clicks the exit button then the player will exit the game, if the player clicks the i button (info) then a credits menu will be displayed as shown in Picture 8.

Picture 8. Main Menu Display

B. The Select Levels display

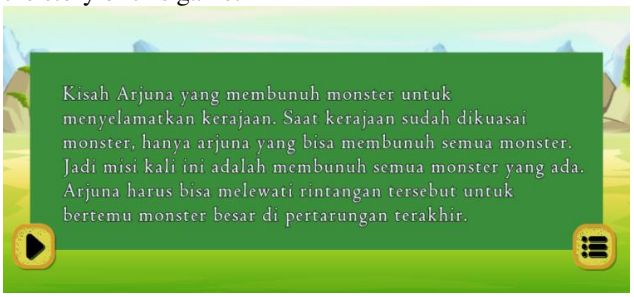
In Picture 9, the level select menu contains level buttons that players can choose. In this menu there are five level selection buttons which when clicked will enter the level page. Inside there is also a menu button to return to the main menu.



Picture 9. Select Levels

C. Story view

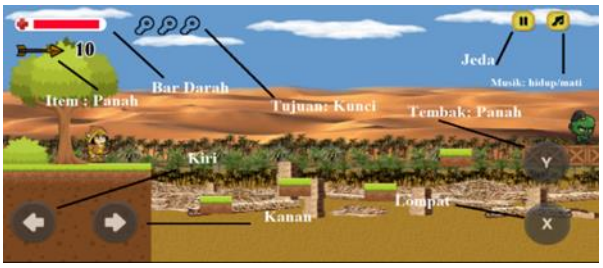
In Picture 10. The Story menu is a menu to explain the story of this game.



Picture 10. Story Display

D. Help display

In Picture 10, the Help menu contains several explanations about the parts contained in the game, in which there is a close button if you want to return to the main menu.



Picture 10. Display Help

E. Info/Credit View

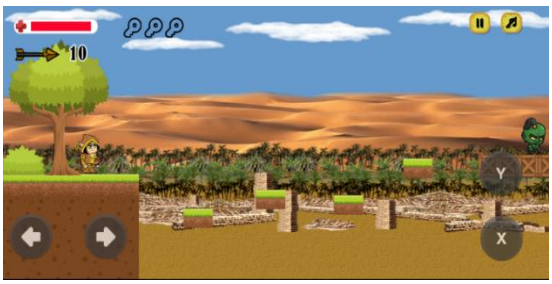
In Picture 11 the Info menu displays only the credit menu and displays the close button to return to the main menu.

Picture 11. Info/Credit Display

Pahrudin, P., Lailiyah, S., & Setiawan , D. A. (2023). Application of Collision Detection in Arjuna's Adventure Game. TEPIAN, 4(1). 45-51.
https://doi.org/10.51967/tepian.v4i1.2232

F. Game View

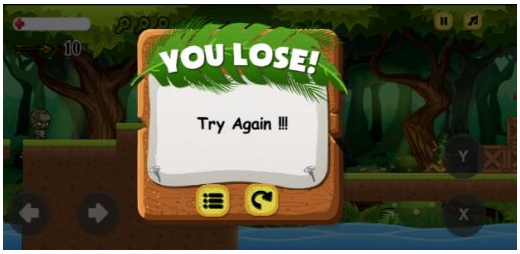
In Picture 12, there is a brief description of the game. And if the player selects the start game button, it will enter the level select menu then the player selects the desired level by clicking on the level then it will enter the selected level and start the game.



Picture 12. Game Display

G. Lose View

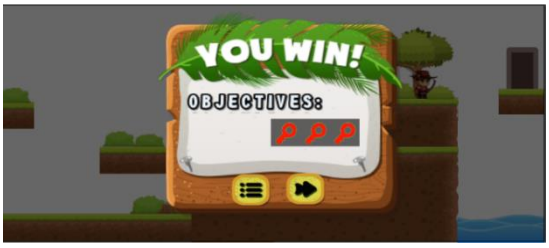
In Picture 13 Display when a player fails to pass an obstacle or dies it will display a popup you lose.



Picture 13. Display Lose

H. Win View

In Picture 14 Display when the player successfully passes the obstacle and manages to collect all the objectives, the next level will open and when passed it will display a popup you win.



Picture 14. Display Win

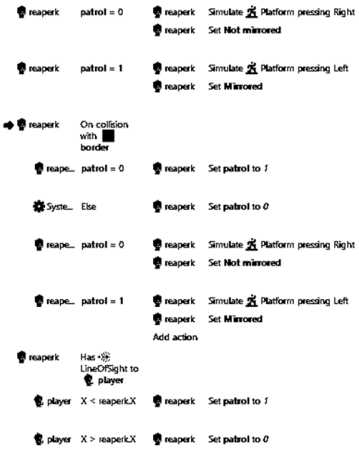
I. Testing

In the testing stage of the Game "Arjuna's Adventure" the author uses two methods, namely:

1. White Box Testing

White Box testing will test several program modules in the game Arjuna's Adventure, testing includes types of testing, types of programs, expected results and test results. The modules tested are as follows can be seen in table 1 White Box Testing

Table 1. WhiteBox Testing

Test Items	Program Code	Results
Collision Detection Algorithm (on Player during collision on Monster)		When a collision occurs it will reduce the health of the player itself namely by adding the subtract from heart parameter to the player.
Line Of Sight algorithm on the enemies before the last enemy		The enemy will patrol when no player is in range or reach, and when the player is right in range from the enemy, the enemy will follow the player.

Pahrudin, P., Lailiyah, S., & Setiawan , D. A. (2023). Application of Collision Detection in Arjuna's Adventure Game. TEPIAN, 4(1). 45-51.
https://doi.org/10.51967/tepian.v4i1.2232

2. Black Box

Testing black box done to build games Arjuna's adventure is to find out the functions on the device whether the software meets the requirements made for the program to build this side scrolling Adventure of Arjuna, focusing on the functional specifications of the software. This test tests the start game button, help button, info button, music button, home button, menu button, restart button, pause button, resume button, next button, level buttons, exit button, and back button which can be seen in table 2 Black Box Test Results.

Table 2 Black Box

#	Tested Items	Results To Be Achieved	Number of Tests	Test result
1	Play Button	The Start Game button functions successfully	3x	Succeed
2	Story button	The Story button function worked	3x	Succeed
3	Help button	The Help button function worked	3x	Succeed
4	Info button	The Info button function was successful	3x	Succeed
5	Exit button	The Exit button function was successful	3x	Succeed
6	Music button	The Music button functions successfully	3x	Succeed
7	Paused button	The Paused button function was successful	3x	Succeed
8	Resumes button	The Resume button function was successful	3x	Succeed
9	Menu Button	The Menu button function was successful	3x	Succeed
10	Next Level button	The Next Level button functions successfully	3x	Succeed
11	Back button	Back button function was successful	3x	Succeed
12	Right Directional Key	The Right Point key functions successfully	3x	Succeed
13	Left Direction Button	The Left Point key functions successfully	3x	Succeed
14	X (Jump) Button	The X (Jump) button functions successfully	3x	Succeed
15	Y Button (Range Attack)	The Remote Attack Button function was successful	3x	Succeed

V. CONCLUSION

The game Arjuna's Adventure was built based on Android so that it is easier for players to play it anywhere. The Collision Detection function in the game Arjuna's Adventure is to detect objects that collide with each other. The Black-box test shows that each button in the game works well and the White-box test shows a method written in the event sheet line.

REFERENCES

Asbudi. 2015. Building a Side Scrolling Game "Grim Reaper Flying In The Darkness". Thesis. STMIK Widya Cipta Dharma.

Bevilacqua, Fernando, 2013. Finite-State Machines Theory and Implementation.

Binanto, Ivan. 2010, Basic Digital Multimedia Theory and Development. Yogyakarta: Andi Offset Publisher.

Erwin, & Purba, FR (2013). "TRUE DESTINY" RPG Game USING THE RPG MAKER VX APPLICATION. Journal of Engineering and Computer Science, Vol. 02 No. 1.

Haryono, SR (2015). Design of a Surabaya Culture Themed Side Scrolling Action Role-Playing Game Using Android-Based Unity.

Hotija, Siti. 2018. Building a Smart Edugame "Flora The Explorer" with Finite State Machine Modeling. Thesis. STMIK Widya Cipta Dharma.

Indrajani. 2011. Database Design in All in 1, PT. Elex Computindo Media,

Istiyanto, Jazi Eko. 2013. Smartphone Programming Using SDK

Android and Android Hacking. Yogyakarta: Science Graha.

kevin. 2018. Implementation of Line of Sight and Image Recognition Algorithms of the Level in Android-Based Game Platforms. Palembang : STMIK Palcomtech

Kuniadi, B., Puspitaningrum, D., & Coastera, FF (2017). Dan Design

Making a Text Message Steganography Application on Digital Audio With

Munir, 2017. Multimedia Concepts and Applications in Education.

Mustaqbal, MS, Firdaus, RF & Rahmadi, H., 2015. Application Testing Using Black Box Testing Boundary Value Analysis. Scientific Journal of Applied Information Technology, Volume I.

Nafianto, Rizky. Mohammed. 2017. Implementation of the Finite State Machine Algorithm as a Determinant of Behavior Change in Non-Player Characters in the Kiki Adventure Game. Thesis. STMIK Widya Cipta Dharma.

Roedavan, 2017. Construct 2 Game Engine Tutorial. Bandung: Bandung Informatics Publisher

Rosa AS and M. Salahuddin. 2015. Structured and Object Oriented Software Engineering. Bandung : INFORMATICS.

Schwarzl, T, 2012. 2D Game Collision Detection, USA : Createspace Independent Publishing Platform.

Subagyo, Ariyadi. 2014. Learning Construct 2 (Design and create your own engaging, extensible, and addictive game using Construct 2). Birmingham. Packt Publishing Ltd

Wibowo, A. (2017). There are 4 Types and 11 Game Genres, Which is Your Favorite? Retrieved, fromsz.

Wijaya, E. (2013). Analysis of the Use of the Breadth First Search Algorithm in the Concept of Artificial Intelligence. TIME Journal, Vol. II No.

Wijaya, E. (2013). Analysis of the Use of the Breadth First Search Algorithm in the Concept of Artificial Intelligence. TIME Journal, Vol. II No.

Ziveria, Mira., Sefina, Ridha. & Rusli, Muhammad., (2014), Design Training

Graphics Using Adobe Photoshop Tools For Photo Manipulation For YPU's Information Technology Team. Journal of Abdimas, Volume 1 no 1 February 2020