AppRhythm Generation : A Game using Binary Space Partitioning Algorithm

A Thesis

Presented to

The Faculty of the College of Computer Studies

Lyceum of the Philippines University - Batangas

In Partial Fulfillment

Of the Requirements for the Degree

Bachelor of Science in Computer Science

Specialized in Game Development

by:

Kevin D. Patio

Katherine Shenna D. Patio

March 2017

APPROVAL SHEET

In partial fulfillment of the requirements for the degree Bachelor of Science in Computer Science (Specialized in Game Development), this thesis entitled "**AppRhythm Generation: A Game using Binary Space Partitioning Algorithm**" has been prepared and submitted by **KEVIN D. PATIO** and **KATHERINE SHENNA D. PATIO** and is hereby recommended for oral examination.

Mrs. Roselie B. Alday, MCS, PhD. Candidate
Adviser

Defended in an oral examination before a duty constituted panel with a grade of _______

Mrs. Maria Cristina M. Ramos, MSCS
Chairman

Ms. Maria Julieta A. Saldua, MSIT
Member

Mrs. Janice E. Velasquez, MSCS
Member

Ms. Jessica H. Cabaces, MAEd
Member

Accepted in partial fulfillment of the requirements for the degree Bachelor of Science in Computer Science (Specialized in Game Development).

Mrs. Roselie B. Alday, MCS, PhD. Candidate
Dean, College of Computer Studies

Acknowledgment

We would like to thank our dean Mrs. Roselie B. Alday for helping us and guiding us through the development of our special project thesis, to Mrs. Janice E. Velasquez for teaching us about game development, along with being our class adviser during our summer classes on game development, and to the LPU-B CCS Faculty for being friendly and helpful in educating us and taking us this far into our course.

Dedication

We dedicate this thesis to our parents for helping us get to school every school day and funding our studies. We thank them for all they did for us, we could not get this far without them. We hope to repay them by graduating our course, and to become game developers. Also, we dedicate this game to our friends, whose company means so much to us. Thank you to all.

Table of Contents

List of Figures

Figures **Page**

AppRhythm Generation: A Game using Binary Space Partitioning Algorithm

Kevin D. Patio
Lyceum of the Philippines
University - Batangas
Capitol Site, Batangas City
kevinpatio.ccs@gmail.com
ORCID No.: 0000-0002-3869-2343

Katherine Shenna D. Patio
Lyceum of the Philippines
University - Batangas
Captiol Site, Batangas City
katherinepatio.ccs@gmail.com
ORCID No.: 0000-0001-9185-2126

Abstract

AppRhythm Generation: A Game using Binary Space Partitioning Algorithm, is a game about three girls that go through the world and the internet to find and battle internet hackers, net-pirates, and their minions. AppRhythm Generation: A Game using Binary Space Partitioning Algorithm was developed using Unity, which utilizes the BSP tree for 3D collisions and interactions in a 3D space, and also uses the C# language for the coding of AI, player interactions, and other gameplay aspects. The game is a mix of action-adventure and rhythm, and can be played by using either a keyboard and mouse, or a gamepad. The game is at its base proportions, where it’s early base models are complete and it’s first 3 stages are playable. As for functional purposes, the menus, moving, attacking, and simple artificial intelligence are also finished. With the required, minimum aspects completed, the player will be able to interact and experience with their surroundings in the game’s 3-Dimensional environment.

Keywords:

Binary Space Partitioning, BSP tree, C#, Game Development, Unity, Unity Engine

1.0 INTRODUCTION

AppRhythm Generation, using Binary Space Partitioning tree to detect 3D collisions from one object to another, is a 3D computer game where you move, dodge, and attack enemies and try to reach the end of each level while following a musical beat pattern to predict the enemies’ movements.

The Binary Space Partitioning tree is what Unity uses for its 3D collisions. The algorithm may be different if another game engine would be used. The purpose of the tree is to create collisions which allow player interaction in a 3D space. It will allow the player character to interact with terrain, objects and other entities within the game for a more reliable and challenging gaming experience.

Unity was chosen for it is one of the easiest 3D game engines to use and is a more suitable choice for the game’s mechanics, among all the other possible free game engine choices. Unity is also the

3D game engine that the researchers learned how to use during their studies, which is another reason why it was selected.

2.0 THE RESEARCH

2.1 OBJECTIVES

1. To develop a game that will give a 3-Dimensional experience to PC gamers.
2. To use Binary Space Partitioning (BSP) tree algorithm to develop the game.
3. Use C# programming language in the game's development, as it is the language supported by Unity's code engine and that the codes can be easily modified for development purposes.

2.2 LITERATURE REVIEW

2.2.1 About Gaming

Gaming is the activity of playing computer games, though this can be done on more than just computer games, which could be on console, arcade, cellphone, or traditional games, Gaming can be done alone in single-player games, or with other players in multi-player games.In playing these games, it may also become a kind of sport, where players can join in and team up and/or compete against one another. And like sports, gaming is also a method of communication, where a player could interact with other players, from locally to all over the world. And as such, gaming is another way to learn and experience, as games could provide content and other unique activities that could otherwise never be done in real life.

2.2.2 Game Development

Game development is the process of creating or developing a game. This process is done by the game developers, who provide the codes, sounds, models, and other sources required to make a game. In order for the game developers to do so, they use game development programs called game engines, like Eclipse or Unity, and the developer could also add extra assets into the game from the use of open-source sites.

Game development also includes deciding what system the game will be for, and what game genre the game should lean more towards.

There are many who are known for game development and their contribution to the gaming industry. These people create and develop games for people to play and enjoy. Game companies are composed of a large group of people who work on games and are well known for the popular games they make.   Their method of game development may vary, as they may either use a currently existing game engine, or they may create a game engine of their own for more freedom. Indie developers, which is composed of an individual or a small group of individuals, also develop games, albeit with lesser people compared to a game company, and may run their project with the help of funding or donations, and on some occasions, for free. Their methods of game development is similar to game companies, in that they can use a currently existing game engine or create their own. Their games are classified as "indie games".

2.2.3 Types or Kinds of Games

There are many kinds of games. Games could be played on a console, computer, a mobile device, or traditional. The console, computer and mobile could have games that come in 2D or 3D, while Traditional games composed of board games, puzzle games, and the like. However, these concepts can also be implemented on the former three game types as well.

Console games and computer games usually come in 3D. The systems of a console or PC is usually strong enough to support 3D settings and their respective graphic layouts. Their game's maps are usually vast and overall settings are highly graphical. Physics is also a notable factor in 3D games. Depending on the 3D game, the designs could either be simple or complex. Simple designs and effects in a 3D game makes it easier to run, while complex ones may require higher specifications.

Mobile games usually come in 2D as their system is only capable of handling low graphics to increase the devices' performance. Sprites are used most of the time. Most of these games are played with touchscreen cellphones.

This is not always the case for these systems, as it is not uncommon for a computer to have a 2D game, or on occasion, a mobile device with a 3D game. The systems used certain specs, and these specs will determine which game can be played and which ones cannot. Consoles are slightly different, as they can only run games made for their respective console, but some game titles have multi-platform releases, meaning a game could be played on different consoles (using the consoles' respective game disc/cartridge).

2.2.4 Game Genres

A game is categorized by genres. Genres determine the game's play-style and what kind of game it will be. There are many genres, the most notable ones are the following:

- Adventure games require puzzle solving and non-confrontational character interaction to progress through the game's story. Visual novels fit this type.

- Action games require eye-to-hand coordination. Fighting and platform games are an example.

- Action-Adventure games are a mix of action and adventure, and sometimes introduce other elements into the game that makes it unique from the two. The Legend of Zelda series has plenty of this.

- Role-playing games follow a character or a set of characters through a story or setting, and could also have decisions to choose from that could change the story as the player advances. The Final Fantasy series is a good example.

- Sports games are games derived from sports. Car Racing and Basketball games are sports games.

- Simulator games allow players to interact as/with vehicles, objects, professions, etc., like how one would in real life. To put, certain racing games are also simulators.

- Strategy games require strategy, like how to properly command an army of soldiers, to creating a base at a strategic location (this can also be against other players in real time). StarCraft can be classified as this.

There are many other genres aside

from the ones listed before. Some genres are minor, like rhythm games or casual games. Others are a combination. And because of how games are made nowadays, most if not all game genres also have sub genres under them to further describe and categorize a game.

2.2.5 Use of Algorithm in Gaming

Algorithms are used in the script part of the game's development which determines whatever happens in the game. They are the rules and calculations of the game. For this game, the algorithm used is for moving, attacking, etc. which deals with physics.

2.2.6 Binary Space Partitioning Tree (BSP tree)

A BSP tree is a hierarchical subdivision of n dimensional space into convex subspaces. Each node has a front and back leaf. Starting off with the root node, all subsequent insertions are partitioned by the hyperplane of the current node. In 2 dimensional space, a hyperplane is a line. In 3 dimensional space, a hyperplane is a plane. The end goal of a BSP tree if for the hyperplanes of the leaf nodes to be trivially "behind" or "in front" of the parent hyperplane.

BSP trees are very useful for real time interaction with displays of static images. Before the static scene is rendered, the BSP tree is calculated. BSP trees can be traversed very quickly (linear time) for hidden surface removal and shadow casting. With some work, BSP trees can be modified to handle dynamic events in a scene. (Carl Shimmer, 1997)

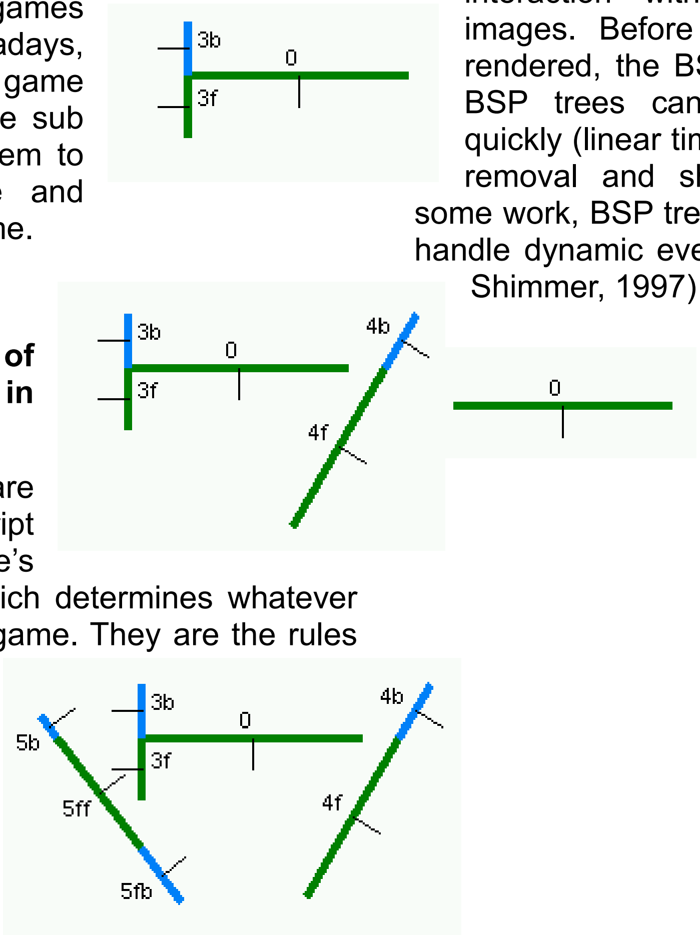


Figure 1.1: BSP Tree (planes)

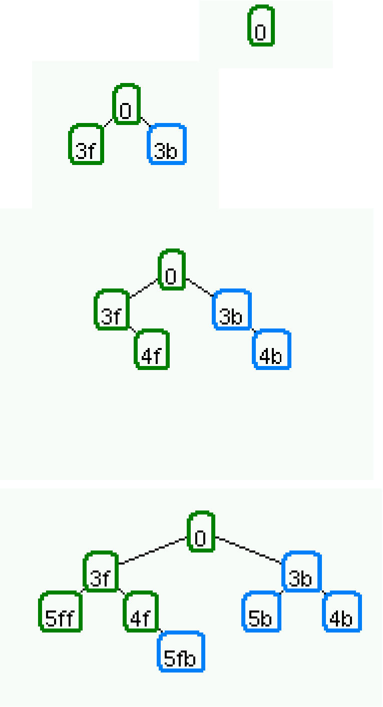


Figure 1.2: BSP Tree (nodes)

2.2.7 Implementation of Algorithm

The algorithm is implemented in the collision, physics and hit-boxes of the game, which is everything on the screen having a 3D form and can be interacted with. When an object touches another object, there will be a returned value which will give results like damage calculations, position axis, etc. making an equivalent visual effect on the gameplay.

2.2.8 The use of C# in the Game

C# is used because it is a language supported by Unity, alongside JavaScript and Boo. C# is widely used in 3D computer games and is slightly faster than other languages. C# is used for the games codes because it would be easier for the researchers to develop the game, not just because of its pros but because they are more used to this language compared to the other languages used for the engine. And because it is easier, and that it is also used by other computer game developers in which they can get a hand from, they decided that C# is the recommended language for the game's development.

It is used in the coding of the algorithm which affects how the physics will go, how fast objects will fly, how objects interact with each other, how the character is controlled, etc.

2.3 METHODS

2.3.1 Game type to be used

The game the researchers planned to make is most likely for the computer, as it is 3D, with the genre mix of Action-adventure and rhythm. The game play is that of Action-adventure were you can roam around the levels and attack enemies, while the structure of the game is of Rhythm were you gain points by doing something at the right time by following a beat.

2.3.2 Sound/Music

Music is an important part of the game for it determines the flow of whatever may happen in the game. Creation of all things in the audio is done using instruments and proper programs for the expected quality. In case of lack of instruments or a proper program, a MIDI program is used to create the music.

2.3.3 Visual Design

Everything is mostly created in 3D using a 3D modeling program like Autodesk Maya, and rigged with the help of mixamo.com, making this one of the hardest parts in the game's development.

Character Designs are based on companies and other computer related objects, with the main characters having designs based on Cellphone companies. The appearances of the main characters are; Appi is based of Apple, Sami is based of Samsung, and Sonia is based of Sony. Other characters that appear are; a character for Java, Apple's IOS, Android and others.

Level and enemy designs are given a futuristic-technology feel.

2.3.4 Game Play

In 3D person, the player controls the characters to fight piracy and illegal doings in the industry by using your rough or slick styles. You play as three characters with their own different capabilities and attitudes.

The whole game runs on a rhythm system which does not affect how you win or lose, but rather it affects the score for each level. Enemies and certain terrain move based on the music of the game. Rhythm points are gained and lost through what you do at specific tunes of the music, like defeating an enemy or dodging their attacks. You move freely as the character and most enemies and certain bosses move based on a pattern of the music being played.

The characters can equip applications. These serve as a weapon which can be used to attack, however they all have a unique purpose; an application could shoot straight lasers, another application could shoot homing rockets, and other apps make you melee and use swords or your fist. You can only use one app at a time, but you can switch between them. Equipped apps are placed in your weapon slots, while the rest of the apps are stored in your inventory. Apps can be bought at the start of the stage, or they can be gained throughout the story. Your stats are affected by your Software and Hardware which can be upgraded by going through the levels. Enemies can be defeated by attacking them using strategy in your actions, like hitting them on a certain spot.

You DO NOT need to defeat any of the enemies, with the only exception being the stage/level's boss, to finish a level or gain rhythm points. You can skip non-boss enemies to get to the finish and dodge or destroy their attacks to gain rhythm points. If you want to fight or dodge your way to the finish, it is up to you.

2.3.5 Design Document

AppRhythm Generation is a game about fighting piracy and figuring out the rights and wrongs about it. You play as three girls that go through the world and internet to find the hackers and pirates in their city. They must fight hacked utilities, robots, viruses, security firewalls, and whatever else the enemies use to stop your advance to their hideouts.

2.3.6 Development Model

The model used is a model called "Game Development Process Model" which has three parts. They are:

- Pre-production phase - All the concepts of the game are made in this phase.
- Production phase - This phase is where the game itself is being created.
- Post-production phase - This is where the game is tested, debugged and improved based on the players' experience.

The purpose of this model was to help the researchers in developing the game and to track its development phases. It helped them determine whether or not the game can be successfully created or if there will still be problems encountered during the development process.

2.3.7 The Mood Board

The researchers used a mood board to help determine what the game will be like. They derived ideas from other games, such as HyperDimension: Neptunia U and NecroDancer, while referencing to styles and the appearances of modern technology to implement the overall theme and design the game.

Target Audience

The game generally targets teenagers (13-19), male and female, that have experience with other 3D action-adventure games but may also apply to the younger and older audiences that have the same criteria.

Art Style

The game applies the Japanese anime Styles to its 3D models making them look more cartoon than real.

2.3.8 Gameplay Overview

Starting the Game

The game can be booted by either: 1.) double clicking the icon on the desktop, 2.) right click its desktop icon then click "find the file location" and double click its executable, or 3.) if on Steam; go to libraries then navigate to the game title then click play.

Figure 2.1: Icon

Figure 2.2.: Icon on the desktop

Once the game starts, it will open up the splash screen,

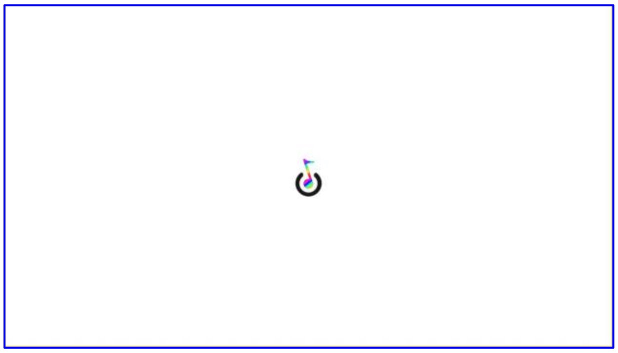


Figure 3: Splash Screen

followed by the Main Menu where you are given options to select a New game, Load a game, and Options menu.

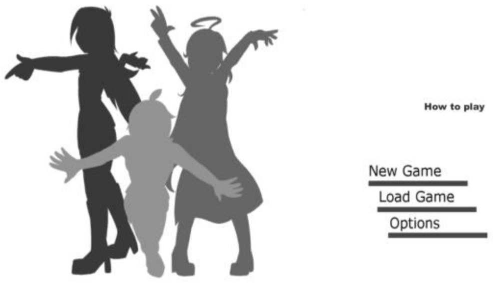
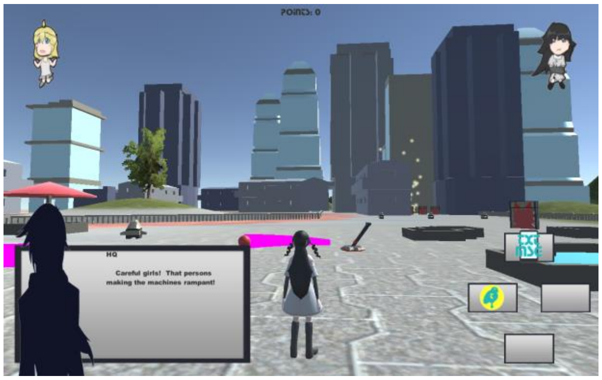


Figure 4: Main Menu



HQ

Careful girls! That persons making the machines rampant!

Figure 5: In-game UI

<u>*Controls*</u>

In-game commands are controlled by the player through either the keyboard and mouse, or a video game controller. They will allow the player to interact with their surroundings and with their in-game menu.

2.4 RESULTS

For the results, the game is, so far, developed to its base proportions. The early base models and the first 3 stages at least, are complete. As for functional purposes, the menus, moving, attacking, and simple artificial intelligence are also complete.

Because of the lack of manpower, it is estimated that 2-4 years may be required to fully complete the game. As a result, the game's current state is what was currently accomplished in less than a year.

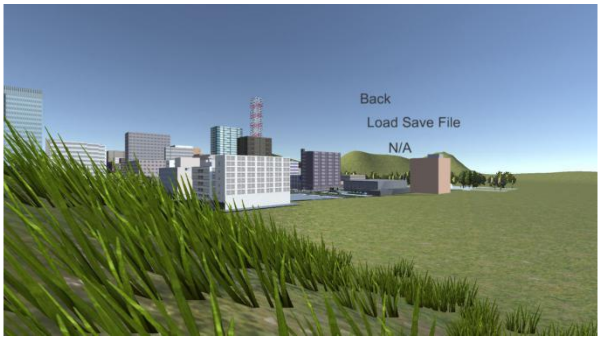


Figure 7: Game Settings - Load Game Placeholder

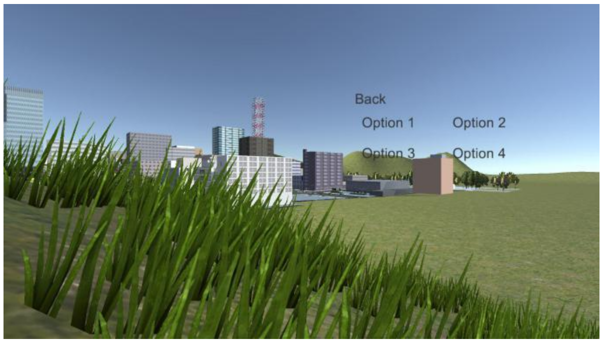


Figure 8: Game Settings - Options Placeholder

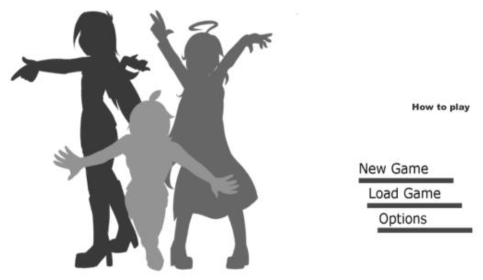


Figure 9: Updated Main Menu

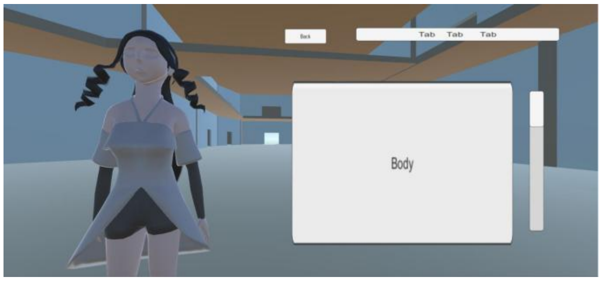
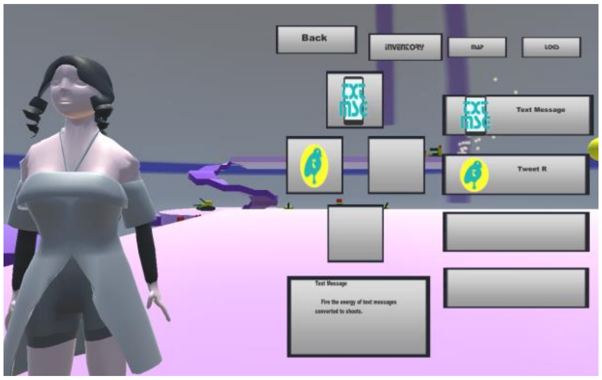


Figure 10: Character Menu Placeholder



Fire the energy of text messages converted to shoots.

Figure 11: Updated Character Menu

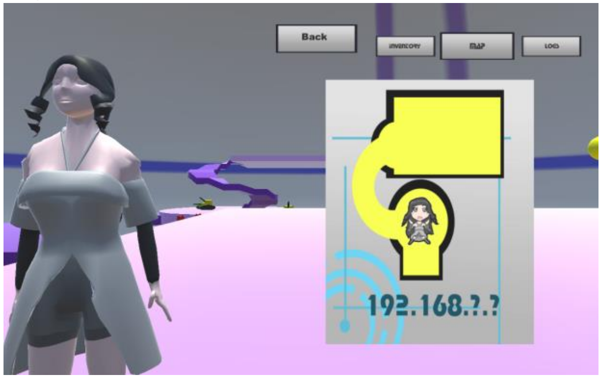


Figure 12: Updated Character Menu - Map

The Main Menu is functional with all its settings being accessible. “New Game” starts the game, and “Exit” ends the application. However, the “Load Game” setting (shown in Figure 7) is not fully functional, with only the “Back” button available to use to return to the Main Menu, and currently serves as a placeholder. The “Options” setting (shown in Figure 8) also currently serves as a placeholder, as it does not have any functional purpose besides the “Back” button. The same applies to the updated Main Menu.

In-game, the player is capable of moving and attacking, and can interact with enemies. However, the player can only interact with placeholder enemies by running into them and attacking only, and the game's main objectives are not yet implemented. The player can also access an in-game menu (shown in Figure 11 & 12) that allows them to manage their equipment, view the map, and view their quest log while playing.

The game's current state is incomplete, with most of its features not yet implemented or are bugged. And due to its current functionality, it is more of a prototype than a game in alpha phase.

2.5 Conclusions and Recommendations

2.5.1 CONCLUSIONS

The main emphasis of the game was to create an action, rhythm-like game, which teaches the lessons of digital pirating.

The use of 3-Dimensions for the game instead of 2D made the experience more satisfying.

For the Binary Space Partitioning Tree algorithm, it did its magic to make the game functional as a game with 3D space and all the necessary gameplay aspects needed for the game.

The C# language used made the development better for the researchers, for that it was a language compatible with the Unity Engine, and that it was a computer language that they knew well.

2.5.2 RECOMMENDATIONS

The following recommendations are given for the improvement of the game.

1. As time was of the essence in making a game, things were rushed to finish it. So, more time may be needed to fully polish the UI, game mechanics and AI logic, as well as all the art assets needed so that the game would feel and look better.
2. As the story and lore may be concrete and fully written, if they were to fully implement it within the game it would make the experience fulfilling and complete.
3. To make the game accessible to the lower-end PCs, it would be great to widen the scope of audiences and players, so aiming for low graphical settings option as well as providing an option for high graphical settings would be great for the game.
4. Being able to work on other platforms like Mac and Linux, it would also be a great advantage to widen the audience a little further.
5. Lastly, adding more things to do in the game other than the main story alone would be make it more entertaining and provide a decent challenge, like adding new features, modes, perhaps easter eggs, or additional side-quests.

3.0 REFERENCES

1. *Baker, Martin. (2015).*
   3D Theory – Binary Space Partitioning (BSP) tree.
   Retrieved from
   http://www.euclideanspace.com/threed/solidmodel/spatialdecomposition/bsp/index.htm

2. *Baker, Martin. (2015).*
   3D Theory – Collision Detection.
   Retrieved from
   http://www.euclideanspace.com/threed/animation/collisiondetect/

3. *Definition of "gaming"*
   Retrieved from
   http://www.merriam-webster.com

4. *How does 3D collision / Object Detection work?*
   Question answered in
   http://stackoverflow.com

5. *List of video game genres*
   Retrieved from
   https://en.wikipedia.org

6. *Shimer, Charl. (1997).*
   Binary Space Partition Trees.
   Retrieved from
   http://web.cs.wpi.edu/~matt/courses/cs563/talks/bsp/bsp.html

7. *Sung, Kelvin. Shirley, Peter. (1992).*
   Ray tracing with the BSP tree.
   Reference from
   http://dl.acm.org/citation.cfm?id=130791

8. *Unity Tutorials - C# vs JS syntax*
   Retrieved from
   https://unity3d.com/learn/tutorials/modules/beginner/scripting/c-sharp-vs-javascript-syntax

9. *Unity Tutorials – Physics*
   Retrieved from
   https://unity3d.com/learn/tutorials/topics/physics

CURRICULUM VITAE

PERSONAL INFORMATION

Name:	Katherine Shenna D. Patio
Age:	21
Gender:	Female
Date of Birth:	April 4, 1995
Address:	Sitio Balisong, Conde Labac, Batangas City
Contact Number:	09052645595
Email Address:	katherinepatio.ccs@gmail.com
Nationality:	Filipino-American

FAMILY BACKGROUND

Father's Name:	Roberto H. Patio	**Contact Number:**	N/A
Occupation:	Medical Technician		
Mother's Name:	Leonora D. Patio	**Contact Number:**	09167110366
Occupation:	Housewife		

EDUCATIONAL BACKGROUND

Primary:	Boulevard Heights Elementary (5th Grade - FL)
	Carmel School of Batangas (6th Grade - Philippines)
Secondary:	Carmel School of Batangas
Tertiary:	Lyceum of the Philippines University – Batangas

CURRICULUM VITAE

PERSONAL INFORMATION

Name:	Kevin D. Patio
Age:	19
Gender:	Male
Date of Birth:	March 22, 1997
Address:	Sitio Balisong, Conde Labac, Batangas City
Contact Number:	09152407967
Email Address:	kevinpatio.ccs@gmail.com
Nationality:	Filipino-American

FAMILY BACKGROUND

Father's Name:	Roberto H. Patio	**Contact Number:**	N/A
Occupation:	Medical Technician		
Mother's Name:	Leonora D. Patio	**Contact Number:**	09167110366
Occupation:	Housewife		

EDUCATIONAL BACKGROUND

Primary:	Carmel School of Batangas
Secondary:	Carmel School of Batangas
Tertiary:	Lyceum of the Philippines University – Batangas