GAM6001-18 Major Project

Individual Report

Assessment 02

McDonald, Bradley

1608793

Contents

[Abstract (Do Last) 2](#_Toc5263322)

[1.Background 2](#_Toc5263323)

[1.1 What was your team aiming to do? 2](#_Toc5263324)

[1.2 What was the composition of the team? 2](#_Toc5263325)

[1.3 What role did you play in the team? 2](#_Toc5263326)

[1.4 What state was your prototype in at the point when this module started? 2](#_Toc5263327)

[2.Introduction 3](#_Toc5263328)

[3.Technical Art 3](#_Toc5263329)

[3.1 A Designer Creating Technical Art 3](#_Toc5263330)

[3.1.1 Brief discussion of relevant literature 3](#_Toc5263331)

[3.1.2 How this work was approached (design etc) 4](#_Toc5263332)

[3.1.3 Problems, solutions and evaluation of this work in comparison to research undertaken 4](#_Toc5263333)

[3.2 The second task undertaken(change) 4](#_Toc5263334)

[3.2.1 Brief discussion of relevant literature 4](#_Toc5263335)

[3.2.2 How this work was approached (design etc) 4](#_Toc5263336)

[3.2.3 Problems, solutions and evaluation of this work in comparison to research undertaken 4](#_Toc5263337)

[4. Reflection 4](#_Toc5263338)

[5. Conclusion 5](#_Toc5263339)

[6. Bibliography 6](#_Toc5263340)

Figure Table

[Figure 3- Moving Platform Tool 6](file:///E:\Uni%20Work\3rd_Year\Dissertation\MCDONALDBradley_Draft_Dissertation004.docx#_Toc6577879)

[Figure 4- Fence Tool 6](#_Toc6577880)

# Abstract (Do Last)

This report covers the intricate details of the development of the project Koala Rama from one of the designers. The topics covered are to be backed up with relevant literature and

# 1.Background

The Team, for this module, created a 3D platforming game. The game was based on a bi-pedal Koala, who has to rescue his friends throughout a variety of worlds by utilising a variety of mechanics. The title of this game was Koala Rama.

## 1.1 Aims of the project.

The team wanted to create an aspiring 3D platformer with added unique gameplay mechanics whilst also including generic 3D platformer mechanics (such as Walking, Running, Jumping and double Jumping) as to still seem familiar to people who have played 3D platformers before. The team decided to add the unique elements as to set it aside from other previously released games in the same genre. Games of the same genre consist of Mario and Spyro, though these are developed by different companies they fall within the same genre, a 3D platformer. A 3D platformer as described by… (Minkkinen, 2016) “’Platformers’ or ‘platform games’ are games that mainly revolve around a character controlled by the player, which runs and jumps to avoid obstacles and/or to defeat enemies”. This was the main gameplay element of the project, Koala Rama to explore multiple world whilst jumping to multiple platforms floating or not, avoiding hazards along the way in the form of obstacles or enemies.

## 1.2 The Team Composition

The team was made up of five third year undergraduate games students from various disciplines, an artist, a programmer and three designers. Each person applied skills that adhered to their role. Whilst branching into artist or programmer depending on workload. The artist created 3D models, unwrap, textured and imported into engine for the designers to place within the game. The Programmer focused primarily on Enemy AI and fixed bugs when attempting to build the game. The Designers would create tools and polish to the game in the form of particle effects or sound whilst creating and designing multiple levels for the player to navigate in Unreal Engine 4 (UE4), (Epic Games,1998).

## 1.3 The authors role in the team

The Author carried out his role in the team as a designer in a small independent game development (indie) team would. As quoted by Mike Bithell: 'The highs and lows of creative freedom are very addictive'; Having developed Thomas Was Alone and Volume, the indie games designer is both a one-man studio and a devoted collaborator. But what does his work actually involve? 2015, , Guardian Newspapers’. Which implies to use and learn a wide range of skills within the project, this is different than being a designer on a large AAA team, where the role would have been more specific. The role of being a generic designer meant going into different aspects of game development, such as; Gameplay Programming, Animating, UI Design, Technical Art and VFX. This allows for constant learning and developing new skills as the project progressed.

## 1.4 The state of production from first milestone

The prototype was at as stage where the core mechanics were working, such as; walking, jumping, rolling and throwing the boomerang. With these mechanics there were a small number of visual glitches to do with animations, of which needed polishing. The game itself was playable with one main level. However, with the overall design of the level done and with the better tools developed for level design, it became easier and more apparent to make more levels. The team then began planning on creating more levels and refining the ones already made.



## 2.Introduction

As one of the Designers of the team, many roles had to be undertaken to provide a quality product. The document will go into detail of these roles, what they were, how they were done and what effect these implementations had on development. These effects can range from implementing feedback from testing results, to setbacks in development which led to rework of the original design. This document will also go into detail of relevant literature and research of which informed the approach in this project, and if these findings actually were reflected in the project or not. The document will go into detail of the work the author has done on the project, the testing of the project and the potential political implications of the project. This will be followed by a reflection and conclusion of the authors work through the project, Koala Rama.

The author will relate back to previous submitted work at relevant parts to show how the project developed based upon certain aspects of the submitted work and project. This will primarily be focused upon when discussing testing and implementation of design into the developed project, including the potential political aspects of the project.

# 3.Technical Art

This section will cover the role of a technical artist and how the author took up this role through development to provide a greater service to the team and project.

The job of a technical artist as defined by (Full Sail University, 2018)

“Something of a hybrid between an artist and a programmer is the video game technical artist. This person works under the direction of the art director and technical art director, and is responsible for the systems and tools associated with creating and porting art assets”.

Though that is a definition of a technical artist, it is not the clearest. As an article by (Sokanu, 2019) states that

“The role of technical artist is a relatively new one, but it is becoming increasingly important as consoles and PC hardware become more complicated. A technical artist works closely with the lead artist and the creative director, as well as the lead programmers. Their responsibilities include setting up and maintaining the workflow of art production, deciding which art packages and tools a studio should use, investigating new techniques, and then going ahead and implementing them”.

Both quotes interlink with saying that the technical artist works alongside programmers and artists however verge when discussing tools, (Full Sail University, 2018) makes it seem that the technical artist simply decides on the tools that are being implemented whereas (Sokanu, 2019) goes into that the technical artist actually creates the tools needed for production. A technical artist at Ubisoft Bucharest (Ubisoft, 2016) goes onto say that “In the beginning we’re focused on creating the pipeline, workflow and the tools the artists can work with”. This keeps with how the author approached the role, creating various tools for the designers and artists whilst maintaining a stable version of the game through optimisation.

To take up the role of a technical artist, as based upon by the above, is to utilise both art and programming to create a multitude of tools so that Designers can speedily design, create and test level layouts and adjust them easily based on feedback. Be that feedback internal or external. That was one aspect of the role. As the author’s role of being a designer, with experience in both art and programming. The job was possible. With existing knowledge of game design, allowed the author to tailor the tools so that a fellow designer would be able to fully utilise the tool with little or no tutorial. As they knew themselves of what a tool needed and how it needed to function in the editor. As designers have an advantage of knowledge in other disciplines, as described by (Schell, 2008) in the Chapter ‘What Skills Does a Game Designer Need?’ she foes onto describe “ In Short, all of them. Almost anything that can be good at can become a useful skill for a game designer”. She then goes to list a number of skills from Anthropology to Visual arts. As the team was mostly designers and were all disciplined in these wide variety of skills, communication was simple through development of tools and of the project overall.

Another task that comes with the role of a technical artist it to optimise the game, by viewing the in-engine profiler, to see if any imported assets, created by the artist, effect the game. Either by performance (Frames per second) or if they create visual glitches (missing polys or wrong facing normals). This can lead to unanticipated stops in production of which then need to be solved before any new alteration or iterations are added to the game.

Figure 2- Unreal Engine Profiler- ‘Koala Rama After Optimisation’. Final Build

Figure 1- Unreal Engine Profiler- ‘Koala Rama Before Optimisation’. Early Build

The green bar seen in figure 1 above, shows the memory usage for the Third Person character. This visualisation aids in debugging which assets need correcting. In this instance it was the Third Person Character Blueprint that was taking the majority of the memory. So, to optimise the game the author researched a variety of methods to optimise the game. Originally in the design of the game the player was to gradually gain their abilities such as double jump and roll, with this in mind the author researched ‘Composition over inheritance’ a resource is (Johansson, 2015) whom details to ‘define objects of what they can do rather than what they are’. So that later I could define the playable character as an object that could ‘fight’ or ‘roll’ rather than constantly checking if they could due do those things by what it is. This method was ideal however later in production the team decided it was better to ensure the player has all the abilities from the start of the game. This decision was made due to internal testing and also development time constraints, the team could not get the ‘unlock ability’ system working whilst also implementing other assets into the game. As the system was never in place, the level designer never designed a level based upon gradual unlock and made them more open as they were used to having everything unlocked.

## 3.1 A Designer Creating Technical Art tools

The tools that were created were to primary aid designers in prototyping levels and to quickly alter levels for further development. The tool as shown in figure 3 is the moving platform tool. The designer can place a platform of any mesh they want and designate the location they want it to go, the designer can also decide how long they want the platform to get there over a designated amount of seconds, they can also chose if they want the platform to rotate and in any direction they want, the platform will always go back to its original location and rotation. This made implementing gloating platforms and jumping puzzles easy as the designer can simply place them in and with the seconds over time feature can easily match up two or more platforms in a row so that they do not become out of synch.



Figure 3- Moving Platform Tool

## 

The Fence tool as shown below in figure 4 allows the designer to quickly place fences around a given area whilst maintaining the same direction as the previously placed fence as to give the fence aligning some continuity. This feature can be overridden easily by manually adjusting the fence to create more sharp shapes such as for creating corner or to square of a section. Using this method also decreased draw calls within the level which means the level loads faster, as the single spline mesh is loaded rather than individual meshes. As shown by “””””””””, by utilising spline meshes and reducing draw calls we were able to place more assets into the level there by making them more decorated and pleasing for the player to look at and attempt to make the feel like part of the work as demonstrated by “””””””.

Figure 4- Fence Tool

## 3.1.1 Brief discussion of relevant literature

* James Miller Uni lecture
* Real-time cinematography for games-by Hawkins, Brian

## 3.1.2 How this work was approached (design etc)

The work of technical art was not originally planned but rather was needed by the team as the project progressed. The game needed to become more stable as the game become larger, with more dense levels and more levels altogether. Optimisation techniques were researched and implemented as discussed. The tools idea came out with designers complaining of how it was taking several hours to do a single job which when using the tool could be done in minutes, tools like the path generator and the fence generator allowed the designer to quickly implement their ideas. This made the designers more adamant to level design, which were tested, with the results better levels for the players.

## 3.1.3 Problems, solutions and evaluation of this work in comparison to research undertaken

The problem with technical art is that there a numerous amount of aspects to research before anything can be implemented. This can range from optimisation by using the in engine profiler as described in Figure 1 and Figure 2 to developing tools for the designers and researching how to create the tools within the engine whilst also updating and maintaining the tools to keep the demand of the fellow team members of whom were using them for primarily level design unless the tool was designed for another use. The tools themselves needed further testing before being fully integrated, and if the tool was updated it would then need further testing as discussed by “””””””” .

## 3.2 Testing method

Testing types- internal and external

When designing the game, the team were inclined to create a game with many mechanics all of which needed to be tested as they were implemented into the game. As to make sure that the various mechanics do not interfere with one another and cause bugs, of which of course they did. Attempting to do multiple actions at once, for instance, trying to aim the boomerang whilst rolling would result in the player not being about to throw the boomerang at all. Meaning that at some points where the player needs to hit a target the game was incompletable. That bug was discovered by various external testers, as developers of the game, the testing is internal. As the team developed the game the team played it properly, so this bug never occurred until external testing was done.

## 3.2.1 Brief discussion of relevant literature

## 3.2.2 How this work was approached (design etc)

## 3.2.3 Problems, solutions and evaluation of this work in comparison to research undertaken

## Reflection

## Conclusion

Audio Notes

Ashraf Ismail (Game Director) of Assassins creed- <http://interactive.libsyn.com/ashraf-ismail-of-assassins-creed>

-Talks about respecting culture and history (32:25

-Prototyping mechanics (44:10-44:40)

Tim Shafer -<http://interactive.libsyn.com/tim-schafer-of-double-fine-productions>

-Brainstorming

-Humour in games

-Crowdfunding

Jenova Chen (thatgamecompany) <http://interactive.libsyn.com/thatgamecompanys-jenova-chen>

-interpreting test data like a doctor (36:50-

(37:30-42:50)

42:50-) “think like an experience doctor, evaluate the data from experience rather than from knee jerk reaction”

Cory Barlog (Creative director) <http://interactive.libsyn.com/cory-barlog>

-Implementing features (47:25-48:15

Testing (46:30-46:50) Playtesting every two weeks

-level design (50:00- 51:30

Ian Dallas -<http://interactive.libsyn.com/ian-dallas>

Mechanics -13:50

# Bibliography

Hawkins, BM 2005, Real-Time Cinematography for Games, Charles River Media, Hingham. Available from: ProQuest Ebook Central. [1 April 2019].

HacknPlan

Unique New Academic Degree in Visual & Game Programming for Southern California; Curriculum to Train 'Technical Artists' for Animation and Video Game Industries 2006, , PR Newswire Association LLC, New York

# Bibliography

Full Sail University. (2018). *Technical artist*. Retrieved from GetInMedia.com: http://getinmedia.com/careers/technical-artist

Johansson, M. P. (2015, 9 20). *Composition Over Inheritence*. Retrieved from Medium.com: https://medium.com/humans-create-software/composition-over-inheritance-cb6f88070205

Minkkinen, T. (2016). *Basics of Platform Games.* kajaanin: kajaani university of applied sciences.

Schell, J. (2008). *The Art of Game Design: A book of lenses.* Boston: CRC Press.

Sokanu. (2019, 04 15). *What does a technical artist do?* Retrieved from CareerExplorer: https://www.careerexplorer.com/careers/technical-artist/

Ubisoft. (2016, 04 14). *Tehnical Art At Ubisoft.* Retrieved from Youtube: https://www.youtube.com/watch?v=mJCkPqpn\_Zk

Freeman, W. (2015, Oct 13). Mike bithell: 'the highs and lows of creative freedom are very addictive'. The Observer Retrieved from <https://search-proquest-com.ezproxy.bolton.ac.uk/docview/1721796971?accountid=9653>

EPIC GAMES. (1998) *Unreal Engine.* [Software] Unreal Engine 4.20.3 Cary, NC: EPIC GAMES.

<https://www.theseus.fi/bitstream/handle/10024/119612/Thesis%20-%20Toni%20Minkkinen.pdf?sequence=1>

<https://www.gamesindustry.biz/articles/2013-10-28-going-indie-im-more-scared-of-staying-in-aaa-right-now>