|  |
| --- |
| Advanced Games Programming |
| CGP600 – AE1 |
| Game Design Report |

|  |
| --- |
| Alexander Pearson  Q13253760 |

Contents

[Introduction 2](#_Toc528839904)

[Game Design 3](#_Toc528839905)

[User Stories 4](#_Toc528839906)

[Software Design 5](#_Toc528839907)

[Testing Plans 6](#_Toc528839908)

[Work Breakdown Structure 7](#_Toc528839909)

[Gantt Chart 8](#_Toc528839910)

[Reflection 9](#_Toc528839911)

[Bibliography 10](#_Toc528839912)

[Appendix 11](#_Toc528839913)

# Introduction

# Game Design

The idea which David and I came up with for our game is based on the game of Dodgeball. The core mechanic will involve throwing balls at enemies with intent to deal damage or knock them off platforms, all while they attempt to do the same to you. Victory is achieved by reaching the end of the level having beaten all the enemies and not dying. The theme of the game will be that of fire and hell, taking inspiration from games such as DOOM (ref), with the game being set over an endless pool of lava and the enemies being fire hurling demons.

Graphics?

# User Stories

Together David and I worked through our game idea and wrote out a list of user stories (ref appendix) to determine all the necessary aspects of our game that needed designing. The core of the game is of course the player which has the most components, largely focused around user input.

Core requirements/Alternate ideas?

# Software Design

The most important aspect of the game that needed designing was the player. Most of the player design was done by David, beginning with basic movement (ref appendix). As shown in the class diagrams (ref appendix), the player class is derived from the character class, a generic class that has movement capabilities. The enemy class also inherits from character as they too move around the level like the player. The movement itself will use vectors and then matrices to determine the direction of movement and then the translation required for that movement.

Equation?

Another key part of the player movement that we have chosen to implement is jumping. David designed his implementation using a counter system to time how long the player would remain in the air, before having them drop back down. I however would use a rough implementation of gravity in which I would give the player a vertical velocity, upon jumping, and have this decrease over time due to a deceleration by gravity. This would create a much more natural looking jump that would follow a realistic arc in which the player reaches a point of having no velocity, up in the air, before falling and speeding back up again.

Camera, wvp matrix, Matrices, constant buffer, vertex shader, lighting?

The core mechanic of the game is the throwing of fireballs, which is done by both the player and the enemies. David’s design (ref appendix) outlines, how they will work, but a key difference between the two is the requirement to fire. The player will have a mana resource, represented in the UI (ref appendix) whereas the enemy will only be able to throw on a cooldown. The mechanics of this are shown in the flow chart for the enemy (ref appendix). We chose this method since the player needing to pick up mana refills makes them explore the level more, whereas the enemies won’t need to leave their designated zones.

Mana pickup/mana bar

Demon AI?

Particles?

Menus?

# Testing Plans

Initially I planned to include a plan for some ad-hoc testing in my appendix. However, as I worked through my design I decided that planning it would be a waste of resources. Instead it would be easier to test components on the fly as they are implemented. I did however create a white box test play (ref appendix). (ref?) I plan to use this to ensure that the most integral parts of the code do what they are intended to once the full game has been completed. It is important to test this since any ad-hoc testing would only test the components individually, not when combined as a full unit with others. I also created a black box testing plan (ref appendix) which I will use after all other testing is completed. (ref?) Using this I can have external testers test the game and ensure all the features work as intended on the surface, all ready knowing that it works fine in the code.

# Work Breakdown Structure & Gantt Chart

The WBS (ref appendix) was made by David and shows all the tasks necessary to completing the game. Most of the tasks are derived from the user stories (ref appendix) or from the breakdown of the user stories in the software design. David also put together the Gantt chart (ref appendix) which shows how log each task will take and when they will be completed. Using the Gantt chart, I can see the critical path that must be taken when developing the game to ensure that the most important tasks are completed on time, to not set the project behind.

# Reflection

Working on this design with David showed a lot of strengths of our ability to design together. We created some detailed designs which I believe will be very useful when it comes to developing the game.

What we learnt?

I was also reminded how important it is to backup our work to not lose anything important. I have created a GitHub repository (ref) to hold my project in to ensure safe backups as I develop the project.

Some of the weaknesses of my software design is the

One problem that we identified in out work was the lack of detail in the Gantt chart. After looking through the tasks laid out in the WBS (ref appendix), David set to work revamping the Gantt chart to include much more detail. This led a more in-depth Gantt chart (ref appendix) which shows all the necessary tasks that would go into developing this game.

One of my weaknesses which hindered our progress throughout this assignment is my time management. I could have made more effort to structure when I would do tasks and plan how long they would take. Since I had a partner, it made it more important that I complete tasks when necessary and in some regards that was not the case here. In the future I will use what I have learnt from the WBS and Gantt chart in this assignment to help me better work on other assignments, without having timing problems.

Overall, I would say our biggest weakness in this assignment was our lack of collaboration. Our plan to complete the design was to allocate an even number of tasks to each of us and then design them on our own. We did this, and while we did collaborate on some tasks, I now wish we had done so on more. A lot of the software design for this gam is intertwined and if I had a chance to do this project again, I would try to work on more design as a team rather than hoping to put it all together individually. This would allow for not only a more coherent design, but for us each to improve upon aspects that maybe we think are not up to standard.

# Bibliography

# Appendix