Building a Game

Project Plan

Romina Petrozzi

CS3810 - BSc Final Year Project

Supervised By: Julien Lange

Department of Computer Science
Royal Holloway, University of London

Abstract

The game I am developing is a 3D platformer set in forest ruins, where the player controls an axe-wielding hero tasked with rescuing trapped animals guarded by goblin enemies. Points are awarded based on the number of enemies defeated, and to clear each level the player must rescue all the animals. The level designs will take inspiration from Super Mario 3D World, with different obstacles and enemy situations to challenge the player.

For the development of this game, I have chosen to use Unity game engine. ‘Game engines are platforms that make it easier to create computer games. They allow you to integrate and combine into single unit individual game elements such as animations, interaction with the user, or detection of collisions between objects’ (Barczak M, Woźniak H, 2019). There are many game engines available for 3D development, with notable examples including Unity, Unreal Engine, and CryEngine. As Barczak and Wozniak (2019) explain, these engines provide reusable components, allowing developers to focus on gameplay and design rather than redeveloping fundamental systems. Unity stands out as the best choice for my project due to its powerful capabilities, user-friendly interface, and well documented resources. According to Christopoulou and Xinogalos (2017), Unreal Engine 4 and Unity are the two most developed engines, with Unreal Engine being more suited to experienced users providing remarkable graphics, while Unity is more suited to beginners with a large asset library and simpler user interface.

My motivation for this project stems from my interest in virtual reality (VR) development. By working on a 3D platformer, I am building essential skills in 3D game development that will transition well into VR projects. Unity, being a widely used engine for VR applications, is a strategic choice to help me prepare for a future career in the VR industry.

1 Timeline

I decided to partition the project into two halves. The first half (Term 1) will primarily focus on enhancing understanding through literature reviews, acquiring and familiarising with the data. The latter half (Term 2) will concentrate on an in-depth analysis of the data, the application of data science techniques, and a review of the findings.

1.1 Term 1

Week 1: — • Learning and understanding the fundamentals of Unity game engine
Week 2: — • Finalise the core game concept
Week 3: — • Research into game development patterns and produce short report on them
Week 4: — • Prototyping core mechanics such as movement and collisions
Week 5/6: — • 3D graphics and animation: Finding and using assets, researching and implementing animations, beginning user interfaces
Week 7/8: — • Game architecture: Implementing design patterns
Week 9: — • Initial playtesting and refinement of core gameplay mechanics : Bug fixing and gathering feedback from users
Week 10/11: — • Work on interim report and presentation

1.2 Term 2

Week 1-2: — • stuff here
Week 3-4: — • stuff here
Week 5-6: — • stuff here
Week 7-8: — • stuff here
Week 9-10: — • stuff here
Week 11: — • stuff here

2 Risks and Mitigations

Any project you choose to undertake will inevitably encounter risks and require mitigations. I will start by discussing the general risks and their mitigations and then move on to those specific to my project.

2.1 Hardware Failure

Hardware failure

2.2 Poor Estimation of Tasks

Realistic time management is

2.3 Uneven Balance Between Report/Code

The code is

2.4 Machine Learning Risks

With machine lea

2.5 Overhead of Chemistry Knowledge

As a compute

2.6 Computational Efficiency

It can be the ca

2.7 Conscious Experiments

Experim

Acronyms

- CSP: Crystal Structure Prediction.
- DNN: Deep Neural Network.
- FUSE: Flexible Unit Structure Engine.
- ML: Machine Learning.
- SE: Software Engineering.
- TDD: Test-driven Development.

Glossary

GitHub A cloud-hosted extension of a Git version control system with a web GUI.

Overleaf An online LaTeX editor providing features such as history/version control and collaboration.

VESTA A 3D visualization program for structural models, volumetric data such as electron/nuclear densities, and crystal morphologies.

References

1. Hirsch, E., Graybiel, A.M. and Agid, Y.A. (1988) ‘Melanized dopaminergic neurons are differentially susceptible to degeneration in parkinson’s disease’, Nature, 334(6180), pp. 345–348.

2. Chapuis, S. et al. (2004) ‘Impact of the motor complications of parkinson’s disease on the quality of life’, Movement Disorders, 20(2), pp. 224–230.