

A5 Project Proposal

Title: First-Person Shooter Game Engine

Name: Zac Joffe

Student ID: 207118120

User ID: zmjoffe

Contents

1	Project Outline	2
1.1	Purpose	2
1.2	Background & Motivation	2
2	Technical Outline	3
2.1	Objective 1	3
2.2	Objective 2	3
2.3	Objective 3	3
2.4	Objective 4	3
2.5	Objective 5	3
2.6	Objective 6	3
2.7	Objective 7	3
2.8	Objective 8	3
2.9	Objective 9	3
2.10	Objective 10	3
3	Objectives	4
	References	5

1 Project Outline

1.1 Purpose

To create a 3-dimensional environment that can be explored via a first-person perspective. The player is equipped with a firearm, which can be used to defeat dynamically-placed enemies in the environment.

1.2 Background & Motivation

I've always been interested in creating a 3D game engine using OpenGL. First-person shooters have been one of my favourite game genres for well over a decade. The project is far more focused on the graphics engine as opposed to the game itself. The player shall be able to move around the world (as described by the objectives) and shoot enemies, but the gameplay will be simple and serves as an interesting context for the implementation of the graphics components.

This project is challenging since it employs a wide range of graphics concepts. Nearly all concepts in the pre-ray tracing part of the course will need to be employed, and implementation of most objectives are non-trivial. The project is also highly extensible; the ten objectives provide a solid foundation for further graphics and gameplay extensions. An example of possible extensions ¹ include more complex scenes, improved physics modeling, and hierarchical modeling of enemies for more realistic model movement and variable damage based on hit location. The foundation of this game engine can be used to implement clever enemy AI, such as that found in *F.E.A.R.* [1].

Through this project, I will learn how to create a basic 3D game engine which can be used to create a basic first-person shooter. It will also aid in further developing my understanding of the raster graphics pipeline taught in lectures and in assignments 1, 2, and 3.

¹I don't necessarily plan to implement any of these for the subjective marks. The point is that the project has great expansion potential beyond the specification outlined by the ten objectives.

2 Technical Outline

This section outlines technical details associated with each of the ten objectives. See section 3 for an abbreviated list of objectives.

2.1 Objective 1

This objective is about modeling the area where players can move around. This will be a simple scene with a flat, rectangular floor and four surrounding walls.

2.2 Objective 2

The game will launch in a main menu that is visually distinct from the normal gameplay. The menu will have a button that lets players start the game. Implementation of this feature can be done with a finite state machine with two states — one for the game, and one for the main menu. This state machine can be implemented with the state with polymorphic rendering methods, as described in *Game Programming Patterns* [2].

2.3 Objective 3

[3]

2.4 Objective 4

2.5 Objective 5

2.6 Objective 6

2.7 Objective 7

2.8 Objective 8

2.9 Objective 9

[4]

2.10 Objective 10

3 Objectives

- 1: Model the scene for the player to move around in.
- 2: Main menu user interface which the player will interact with to start the game.
- 3: Texture mapping.
- 4: Particle system.
- 5: Synchronized sound for player actions.
- 6: Static collision detection of surrounding environment.
- 7: Dynamic collision detection for bullets hitting enemies.
- 8: Physics engine with friction and gravity.
- 9: Shadows using shadow volumes.
- 10: Keyframe animation using linear interpolation.

References

- [1] Jeff Orkin. *Three States and a Plan: The A.I. of F.E.A.R.* 2006. URL: https://alumni.media.mit.edu/~jorkin/gdc2006_orkin_jeff_fear.pdf.
- [2] Robert Nystrom. *Game Programming Patterns*. State. Chap. II. URL: <https://gameprogrammingpatterns.com/state.html>.
- [3] P.S. Heckbert. “Survey of Texture Mapping”. In: *IEEE Computer Graphics & Applications* (Nov. 1986), pp. 56–67.
- [4] Franklin C. Crow. “Shadow Algorithms for Computer Graphics”. In: *Computer Graphics (SIGGRAPH ’77 Proceedings)* 11.2 (1977), pp. 242–248. ISSN: 0097-8930. URL: https://www.cs.rpi.edu/~cutler/classes/advancedgraphics/S13/papers/crow_shadows_77.pdf.