6.0 Software Design Description Document

6.1. Introduction

6.1.1 System Objectives

6.1.2 Hardware, Software, and Human Interfaces

6.2 Architectural Design

The overall architecture of the project will be split between the user interface, the player controller, and the game manager. Graphics and physics systems will be modeled and controlled by the Unity game engine. The user interface will be presented in a menu, and will handle starting the game, exiting the game, or changing settings within the game. The player would be confronted with these menus when starting up the game program, and when pausing the game while it is running. The user interface will not relate to when the player is actually playing the game, as the player is only maneuvering through the level, and should not be bothered by menus. The player controller script will handle character movement and game mechanics that the player can perform. The way the character moves throughout the level will be managed within the player controller and all updates to how the player moves will be facilitated here. If the player has any changes with its overall mechanics, this will also be handled here. The game manager script will handle all other interactions outside of the player. All states of the level, as well as scene transition, and level interaction with the player, will be handled by the game manager script. In short, the game manager handles the current overall game state, including how the player has interacted with the level, and whether the player has won or lost at any given moment.

6.2.1 Major Software Components

6.2.2 Major Software Interactions

6.2.3 Architectural Design Diagrams

6.3. CSC and CSU Descriptions

6.3.1 Class Descriptions

6.3.1.1 Detailed Class Description 1

. . . 6.3.1.n Detailed Class Description n

6.3.2 Detailed Interface Descriptions

6.3.3 Detailed Data Structure Descriptions

6.3.4 Detailed Design Diagrams

6.4 Database Design and Description

We did not use a database for our project.