Software Requirements and Design Document

For

Group 5

Version 2.0

Authors:

Aiden Allen

Stefano Sanidas

William Couture

Rafael Cardoso

Aidan McGill

1. Overview (5 points)

At a top level our game is a tower defense game in which enemies will traverse across the screen and the user can place towers along this path in order to stop their progress. The game will feature a set of levels which will feature a variety of enemy types. Complementing this, certain towers will be categorized as well; DIfferent tower types will only be able to be placed on certain surfaces within levels.

Outside of our tower defense levels is an interactive menu in which the user moves a playable character to different on-screen doors to select menu options. Potential rooms / doors include a settings room, an exit door, and a shop room. Additionally, when selecting levels the user will have to select a small subset of their unlocked towers to bring with them into the level based upon the level's enemy type and surface. Completing level will reward the user with currency which can be used to unlock additional towers.This currency can be spent in the shop room to unlock new towers.

2. Functional Requirements (10 points)

1. Enemies should follow the designated path throughout each designed level. - **High Priority**
2. Towers should be able to selected and placed throughout the scene along the path - **High Priority**
3. Menu character should be controllable and able to select doors to level/menus - **High Priority**
4. Items and Character Stats should be able to be saved - **High Priority**
5. User is rewarded with menu currency at the completion of each level - **Medium Priority**
6. User is able to use menu currency to purchase new towers**. - Medium Priority**
7. Menu prompt appears to select towers when entering each level - **Low Priority**
8. Towers can be upgraded in-level - **Low Priority**
9. User can talk to NPCs in the menu. - **Low Priority**

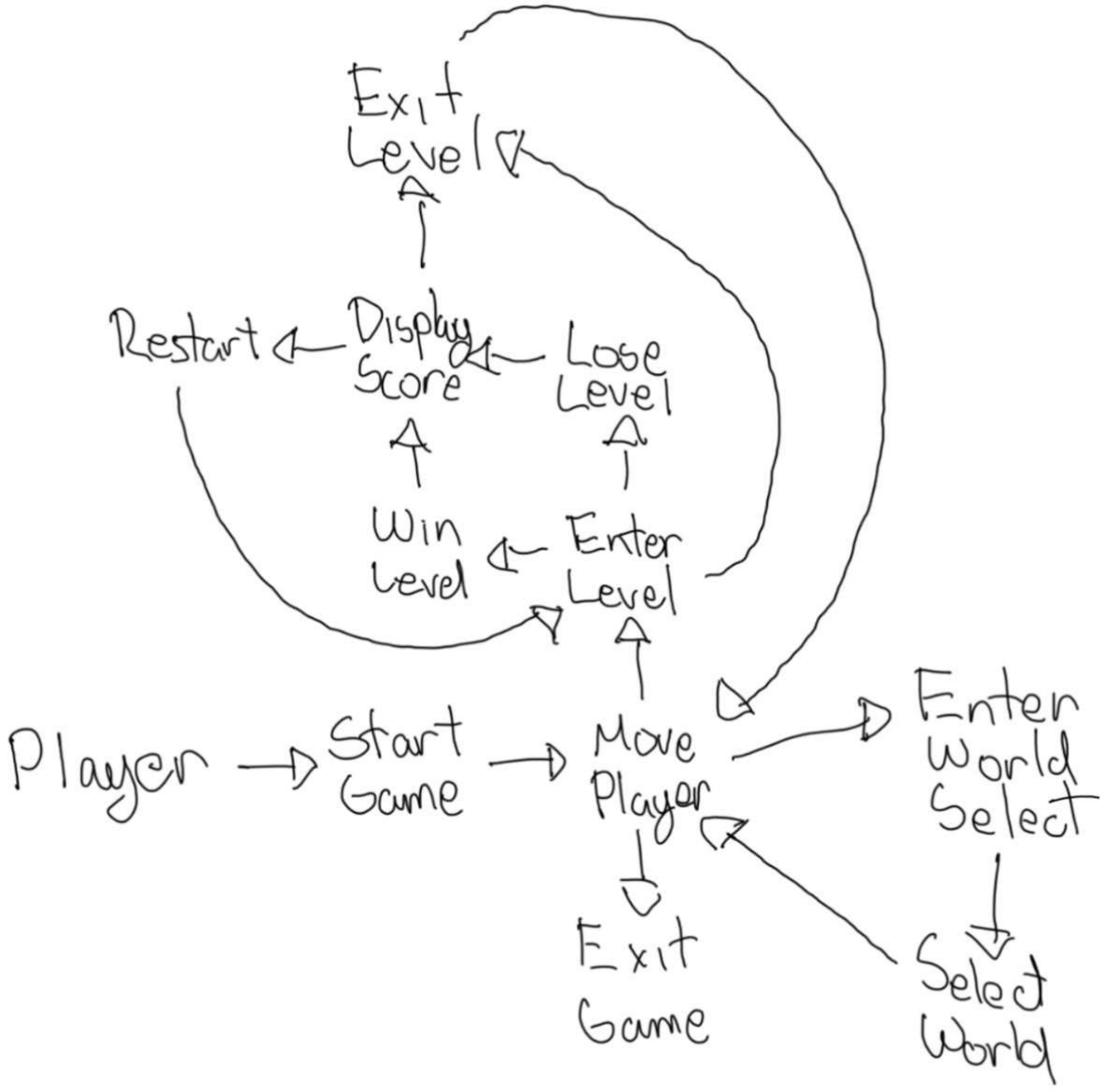
3. Non-functional Requirements (10 points)

1. Will be distributable for the windows operating system.
2. The average frame rate must be at least 30 frames per second.
3. Minimum hardware requirements for the game should be accommodating of most users.

4. Use Case Diagram (10 points)

*This section presents the **use case diagram** and the **textual descriptions** of the use cases for the system under development. The use case diagram should contain all the use cases and relationships*

between them needed to describe the functionality to be developed. If you discover new use cases between two increments, update the diagram for your future increments.



Move Player: Instead of having a traditional menu system, the game uses an overworld in which the character moves around in order to pick the level.

5. Class Diagram and/or Sequence Diagrams (15 points)

Class Case Diagram:
https://lucid.app/lucidchart/35f9560a-94e6-4074-80b2-094435291818/edit?viewport_loc=-3188%2C-292%2C5587%2C2636%2C0_0&invitationId=inv_fd6d7607-b1cd-4af0-b1db-01e871629ef5

Sequence Diagram 1 - NPC Dialogue
https://lucid.app/lucidchart/7daabcbb-ce0d-45c8-ae26-e72cdf98dff0/edit?viewport_loc=-121%2C-29%2C1939%2C1525%2C0_0&invitationId=inv_ba2bc10f-9b90-4742-930a-0284ade1c17e

Sequence Diagram 2 - Level and Tower Selection

https://lucid.app/lucidchart/934bb810-460f-462f-a8a0-a76b65c0d2bf/edit?viewport_loc=-181%2C177%2C2710%2C1278%2C0_0&invitationId=inv_9eb199d8-7d0f-4ebf-8bbc-5b51e6ef8cc3

Sequence Diagram 3 - Shop

https://lucid.app/lucidchart/48f8b5e1-4b94-44aa-9a6f-41d42311aaf8/edit?beaconFlowId=C022A0D9BB70F488&invitationId=inv_87ddf109-6156-43cc-b685-79a76429fc84&page=0_0#

6. Operating Environment (5 points)

The environment in which the software will operate is in a Personal Computer. The operating systems that our game is targeting are Windows 10, Windows 11, and Mac.

7. Assumptions and Dependencies (5 points)

Due to different members of the group using different operating systems, there could be situations in which a bug that happens in one operating system might not show up in the other and thus, make finding the cause of the bug harder. As a result, we depend on Unity's versatility on different systems in order to effectively continue working on the project.

Our project is entirely Unity-based, so we rely on Unity providing secure, stable releases of their software in order to complete development of our project.