

GPG210

Technical Design Document



Written by **Prieance Sagar, Nemah Irshad and Dylan Suaris**

SUBMITTED TO: Anna Tookey & Ahmed Gamal Fouad

Table Of Contents

[Version History](#)

[Overview](#)

[Purpose:](#)

[Scope:](#)

1. [Deliverables](#)
 - a. [Game Deliverables](#)
2. [The Program](#)
 - a. [Game Description](#)
 - b. [Functions \(How will it work in terms of programming\)](#)
 - c. [MVP](#)
 - d. [Minimum Specifications for game to run on](#)
3. [Charts](#)
 - a. [UML diagram](#)
 - b. [UML Use Case Diagram](#)
 - c. [Project Timeline \(ClickUp\)](#)
 - d. [Task Breakdown](#)
4. [Technical Descriptions](#)
 - a. [Tools](#)
5. [Extra Features](#)

Version History

Version #	Date	Reason	Editor
1.0	12/June/2021	First Draft	Dylan
1.5	16/June/2021	Additional Information/ Format Edit	Nemah
2.0	22/June/2021	Added UML Class Diagram Added UML Use Case Diagram Edited program functions	Nemah
2.5	1/August/2021	Added AI Diagrams	Nemah

Overview

Purpose:

This document will be a reference to maintain a proper schedule and have a mockup of the game that everyone can look back to and understand what is needed. It will be a way for everyone to be on the same page.

Scope:

This program will be a 2D top down action RPG where the goal of the player is to help the villagers in the town, bond with their familiar, defeat the necromancer and find a way to return home. The game features multiple combat levels, a leveling system, a reputation system, and gear upgrades.

1. Deliverables

1.1. Game Deliverables

- Must be able to interact with NPC's and Familiar
- Must be able to Engage in combat
- Must be able to Move in the correct manner

2. The Program

2.1. Game Description

A top down 2D action RPG game, where the player must defeat enemies while bonding with their familiar. They attempt to rid a town of undead and find out why they are here with the help of a familiar that resides in this strange world.

2.2. Functions (How will it work in terms of programming)

- Movement script: to move player
- Enums to control AI's states
- Triggers to indicate when the player interacts with NPC's in the game for either combat or general purposes.
- Attack script to use in combat when fighting enemies
- Dialogue

2.3. MVP (Minimum Viable Product)

- Animated 2D Characters and Environment
- Interacting with NPC's and AI
- Engage in combat
- Familiar and Enemy AI

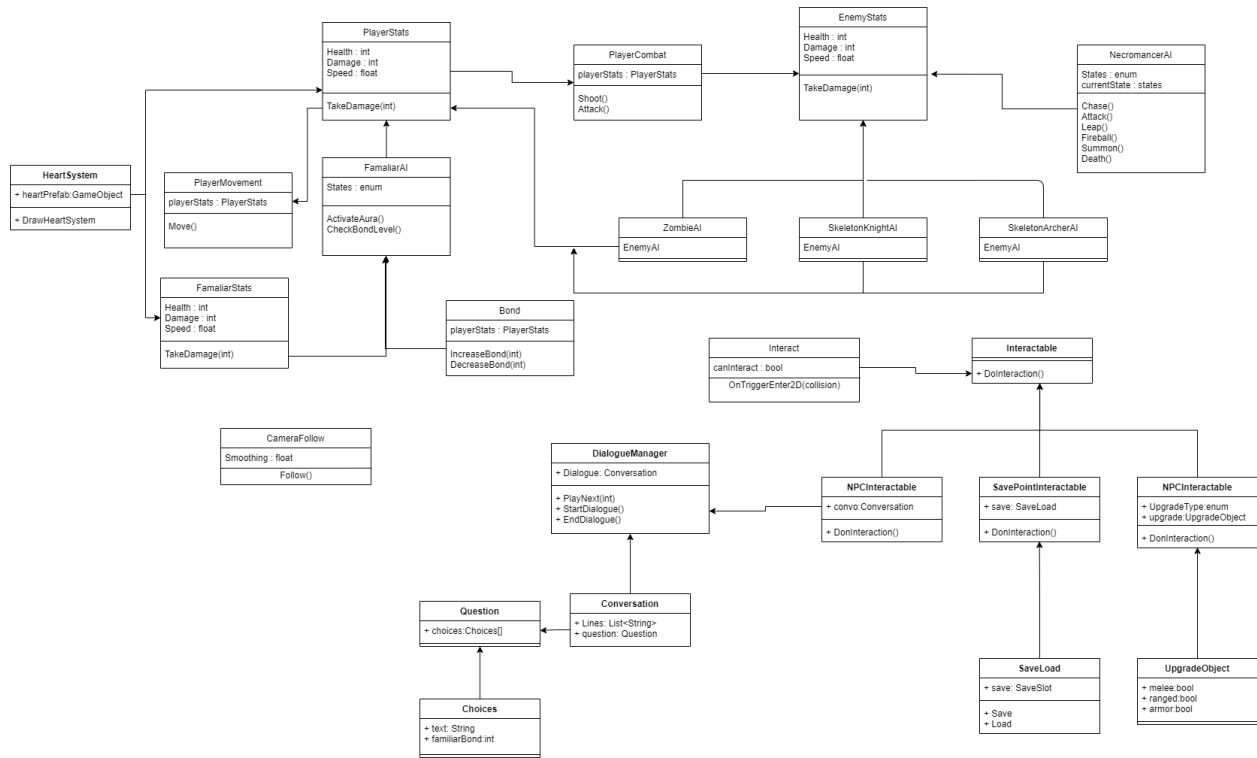
2.4. Minimum Specifications for game to run on

- OS: Windows Vista or greater
- Processor: 2 Ghz
- Memory: 2 GB RAM
- Graphics: 256 mb video memory, shader model 3.0+
- DirectX: Version 10
- Storage: 500 MB available space

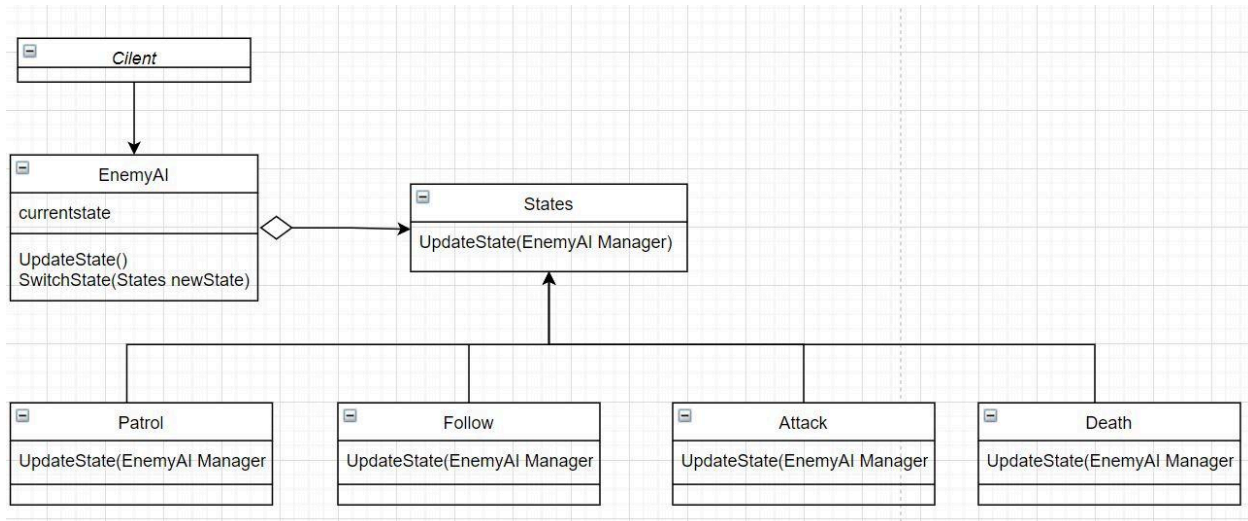
3. Charts

3.1. UML diagram

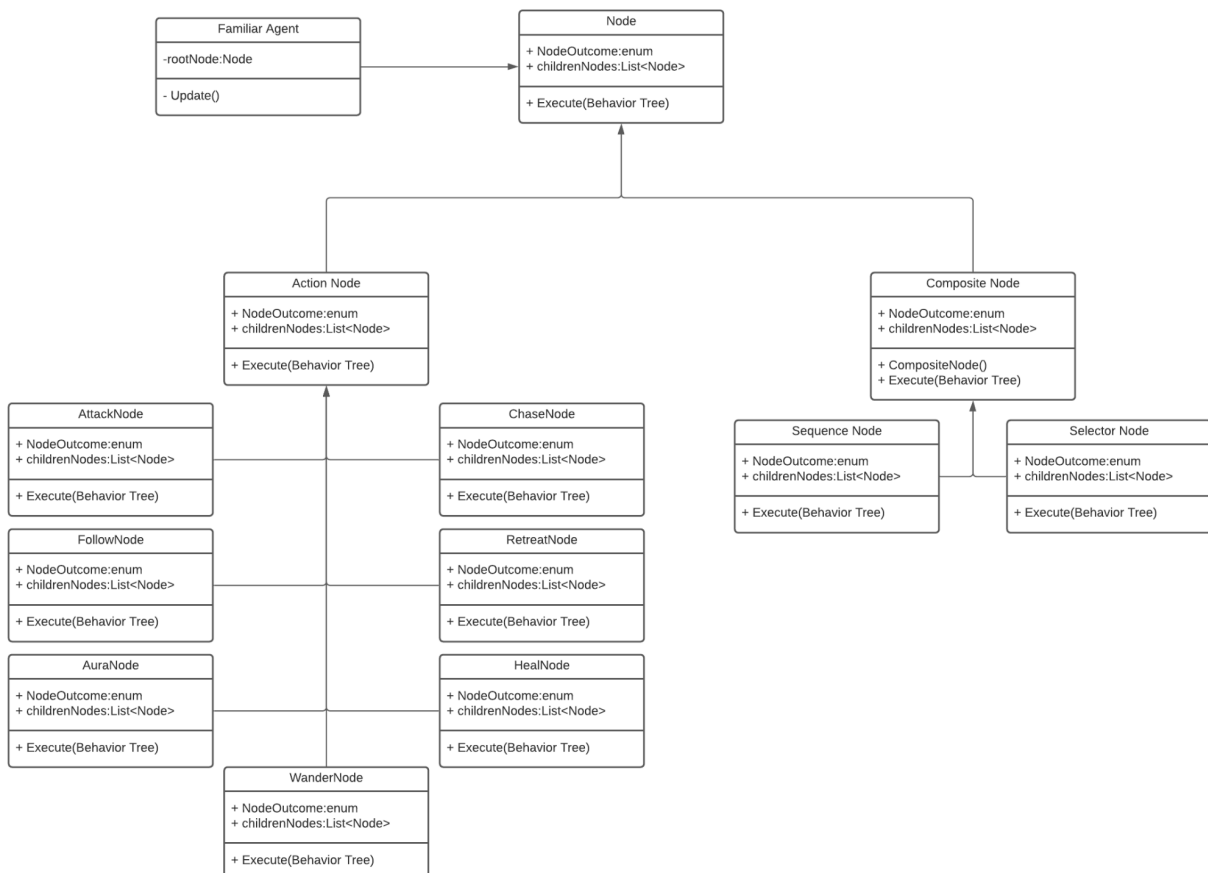
<https://app.diagrams.net/#G1toLwjCEXopDhqqdbtjMKXdRH1aVI0gAQ>



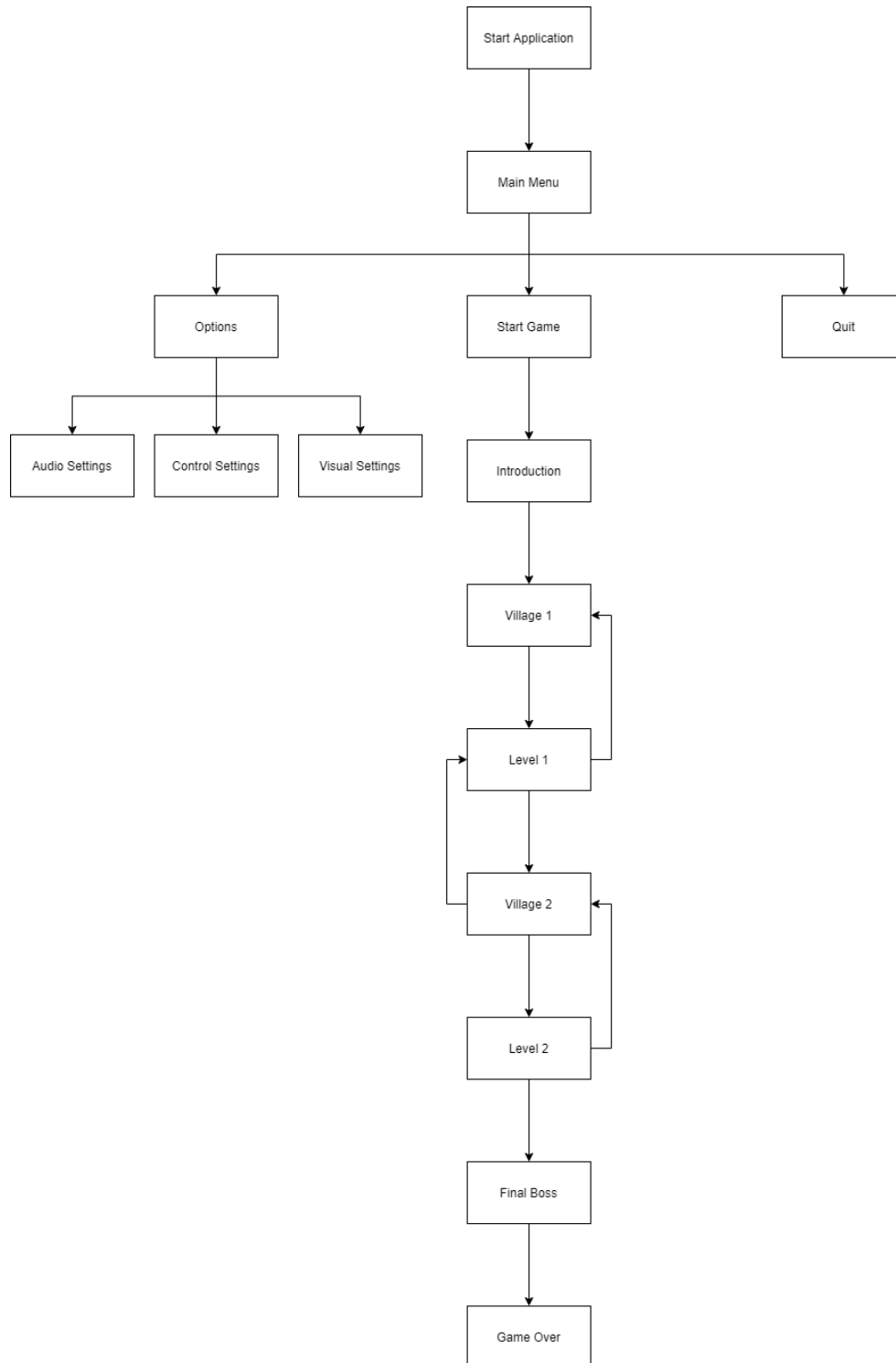
3.2. UML diagram (State Machine)



3.3 UML diagram (Behaviour Tree)



4.4 UML Use Case Diagram

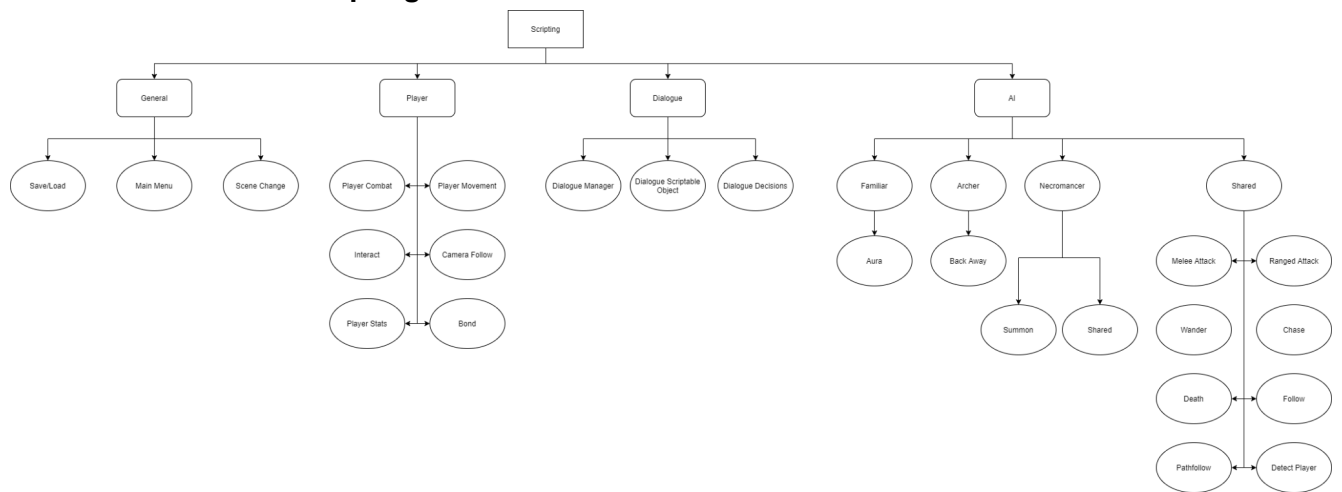


3.3. Project Timeline (Trello)

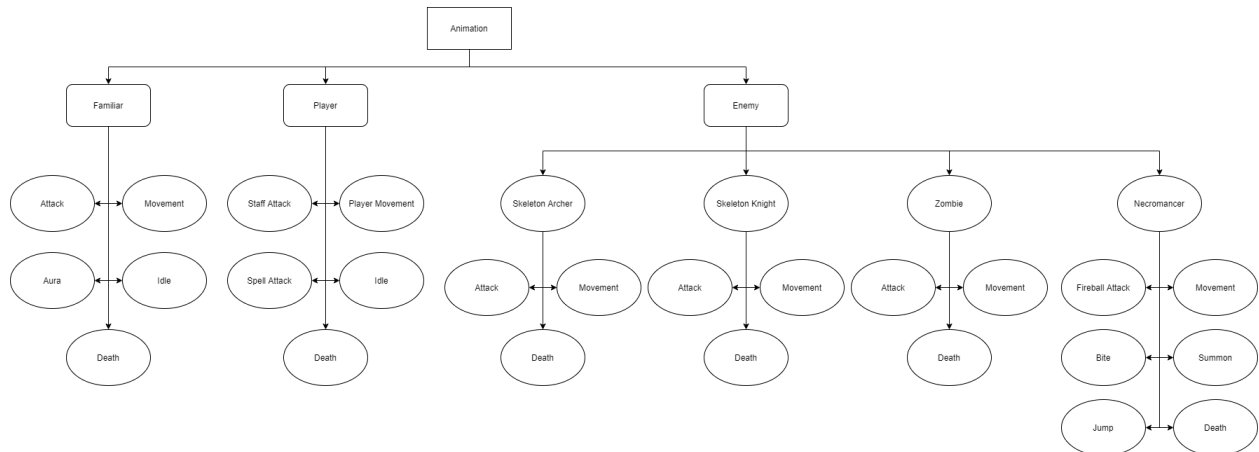
<https://trello.com/b/c5KFEwsD/familiar>

3.4. Task Breakdown

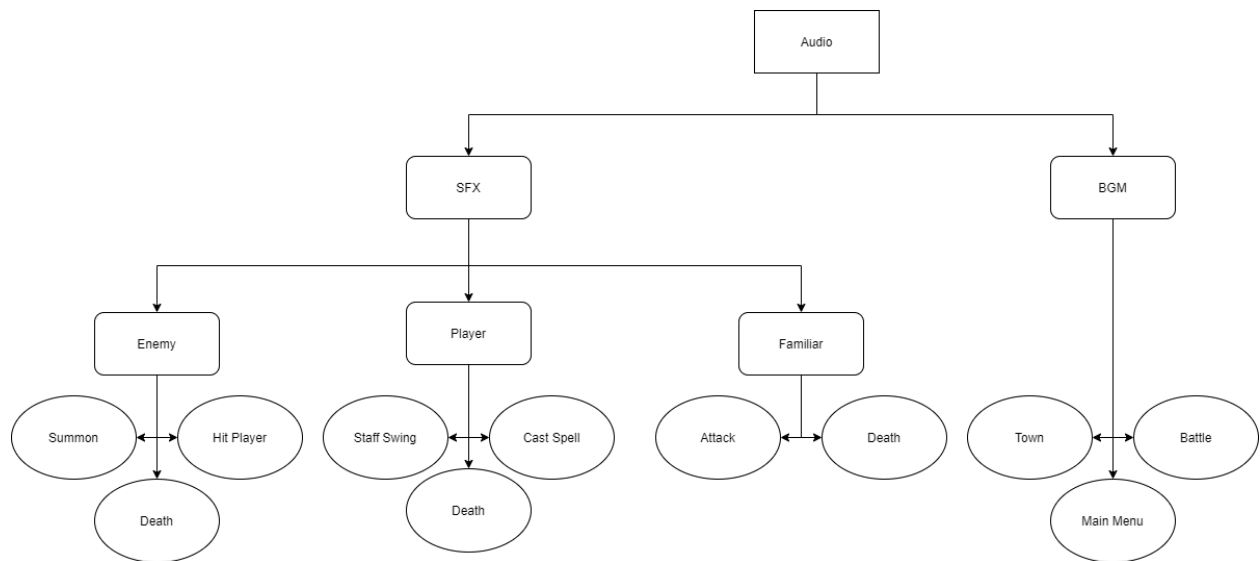
3.4.1. Scripting



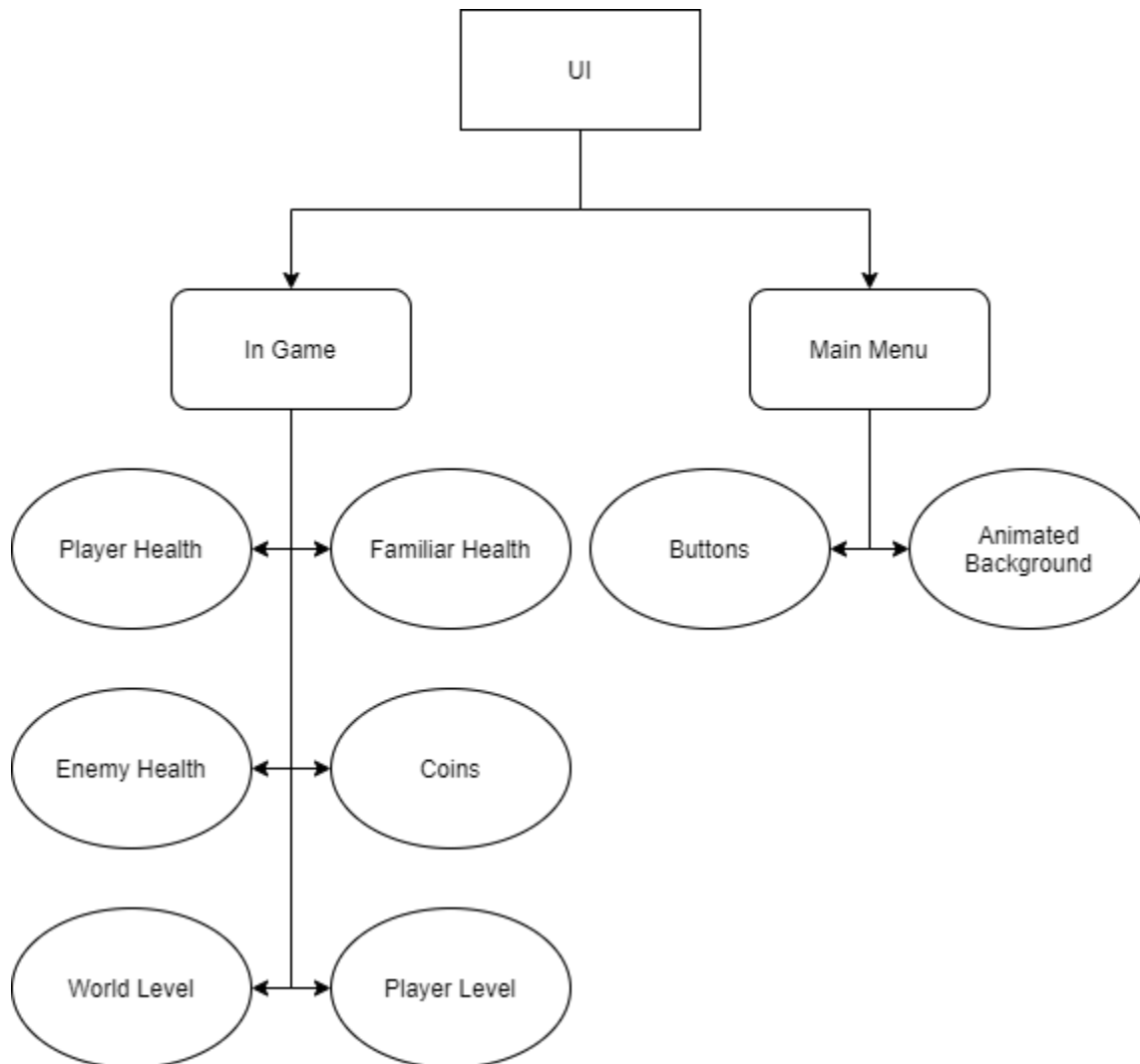
3.4.2. Animation



3.4.3. Audio



3.4.4. UI



4. Technical Descriptions

4.1. Tools

- **Visual Studio**

This will be our main coding software which allows us to code in C# and develop the program.

- **Unity**

This is where we can set up the game design and see how it will run. We will also be using the tile map editor to create our levels.

- **Trello**

This will be used to plan the sprints

- **Diagrams.net**

This will be used to create the task breakdown

- **Google Docs**

This will be used to document the HDD, GDD, and TDD

- **GitHub**

This is a development software which backs up your programs and code and even allows you to return to earlier revisions of your project.