Seizonsha
Clinton P., Steven H., Alex L., Eric T., Sean K.

Architecture Doc

Module Breakdown

Game- The game class is the controller that has all of the game logic and rules. Also contains the game loop. The game class is a black box that provides abstractions to all game entities through which actions can be performed. For instance, the collision dectection is tile based and happens during the MoveEntity() method of the game class.

Level- Contains abstractions for the tiles and is what interfaced the tiles with the game object.

GameEntity- Abstract class which represents all objects in the game. Each GameEntity has an Update() method which is called during the update phase of the game loop and a Draw() method which is called during the draw phase. Also constains basic object information such as x,y coordinate, sprites, etc. The Update() method can be overwritten by child classes for child-specific reasons. All NPCs, effects, projectiles, etc. implement this class. The main Game class handles all of their functionality using polymorphism.

Interfaces

Equipable- Anything that can be equipped to a player's skill slots. Includes items, weapons, and skills. They must implement OnEquip(), OnUnEquip(), and Use().

Unlockable- Anything that can be unlocked from the skill tree. Must implement OnUnlock().

Spawnable- Anything that can be spawned in the game. Must implement OnSpawn(). All spawns are added to a spawn queue and are spawned at the beginning of the game loop within the Game class.

AI- Anything with an artificial intelligence. Must implement an AI() method, which is called during the UpdateAI phase of the game loop.

Classes

Player- Extends the abstract GameEntity class and provides additional functionality such as skill slots, a skill tree, and gamepad input.

Skills- Several skill classes, such as Gun and Bullet implement Equipable and Unlockable.

Collision- A class to handle collisions with an implemented Equals() method that is used to prevent duplicate collisions within a hashset.

NPCs- Any entity that implements AI.

Game Loop

Removal- All entities flagged for removal are removed from the game.

Spawn- Spawns all spawnables in the spawn queue.

Update- Updates all entities currently in the game.

Collision- All collisions that have been generated this loop are executed by calling the Collide() method of both entities.

Draw- Draws the level and calls Draw() on all entities in the game.

Other Notable Features

· Levels are generated using a 2D grid of numbers, where each number represents a type of tile. Within the Level class, there is a Dictionary that maps these numbers to sprites and properties of that tile.
· The game keeps track of the tiles that all entities are occupying at a given time. This allows entity collision to be tile-based efficiently in addition to regular tile collisions.