Johnathan Mai

ITP368 – Assignment 2

Design Document – Tower Defense

**Entities**

All entities are subclasses of the class, Image. The Image class provides basic member variables to load in an image and create an image view at a certain point on the pane.

1. **Landscape**: The landscape class is to be inherited from. Its children classes will create the background “world” of the tower defense game (i.e. the grass, the path the enemies will travel on, etc.). Landscape provides a square polygon for the children to fill in and place whatever will inhabit that “square” in the grid of the game.
   1. **Grass**: The Grass class represents a tile on the grid where the user will be able to build towers to defend against the enemies. The class could also fill in arbitrary structures, such as trees, that will prevent the user from building structures.
   2. **Gravel**: The gravel class represents a tile on the grid where enemies can travel. Enemies will start at a certain gravel tile and move towards the **base**, only on the gravel path. The gravel tile will not allow structures to be built on it.
2. **Enemy**: The Enemy class is the parent class of all enemies. For now, there are 3 types of enemies that inherit. Enemies, controlled by the AI, are designed to move towards the user’s base on the gravel path ONLY. When they reach the base, they will begin attacking the base until the base loses all health. At that point, the user loses. All enemies will have properties for damage dealt, attack rate, move speed, and health. Also, they are all *minions* from the film Despicable Me.
   1. **SmallEnemy**: The small enemy is the most basic unit. It is the weakest (meaning it deals the least damage and has the least health). It will be encountered in earlier levels.
   2. **MediumEnemy**: The medium enemy is a harder unit to kill than the small enemy. It will be encountered in later levels.
   3. **BossEnemy**: The boss enemy is the hardest unit to kill at the moment. It will definitely have higher health and damage, although fewer in numbers.
3. **Tower**: The Tower class is the parent class to all towers that the user can create. Towers shoot projectiles at enemies when they are in “sight,” with the sole purpose of killing them. Towers will cost currency to create and upgrade. Towers are stationary and can only be built on the grass. Like the enemies, the towers will have a damage and attack rate, as well as a *range* where they can “see” enemies. Certain towers are better with certain enemies, so it is up to the user to discern which towers to build from the beginning.
   1. **RapidTower**: Rapid towers are faster attacking towers that deal less damage. They will also have less range than other towers. When there are few units with high health, it will fail.
   2. **SniperTower**: Sniper towers are slower attacking towers that deal very high damage. They have great range, but when there are many units, it will fail.
   3. **SplashTower:** Splash towers are medium attacking towers that deal medium damage. SplashTower has a special feature: splash damage. Splash damage means that wherever the projectile lands, it will deal damage to enemies within a certain radius. The projectile also be aimed towards the enemy at the moment it is fired; thus, if the enemy moves from that point far away enough, it will miss. This is in contrast to the rapid and sniper towers where their projectiles will follow the enemy until it reaches the target.
   4. **Base**: The base is what the user wants to protect at all costs. It will have an attack as well, probably similar to the splash tower without the “splash.” It will have a health very high, where enemies will eventually chip away at.
4. **Projectile**: The projectile is a lone entity that is solely the image of a projectile that is shot from a tower. The projectile has no properties otherwise.

**Other** **Classes**

1. **TowerDefenseAssignment2**: This is the main executable class of the assignment. It creates the scene and stage and is the host of all other classes.
2. **GameController**: GameController is in charge of populating the pane. It manages and populates a 2 dimensional grid of Landscape classes, an ArrayList of towers, an ArrayList of enemies, and the **User**. It gives the animation of randomly waddling around the “map” to the enemies and the movement of the user as well.
3. **User**: The user is a single character for Assignment 2 that the user can control. The User is a child to the Enemy class, so it has similar characteristics to enemies. The user can be controlled, however, by using the mouse to click where on the map the user wants to go.