Johnathan Mai

ITP368 – Assignment 3

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. **Enemies must be clicked to be killed.**
   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. It has 5 health and takes 5 clicks to kill.
   2. **MediumEnemy**: The medium enemy is a harder unit to kill than the small enemy. It will be encountered in later levels. It has 10 health and takes 10 clicks to kill.
   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. It has 25 health and takes a whopping 25 clicks to kill.
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. **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. **TowerDefenseAssignment3**: This is the main executable class of the assignment. It creates the scene and stage and is the host of all other classes. It also instantiates the GameController class.
2. **GameController**: GameController manages all nodes that appears in the game area. Upon instantiation, it populates a 2 dimensional grid of Landscape class, filling the grid with grass and a gravel path for the enemies to travel on. It then creates the base tower at the end of the gravel path, where if enemies reach, the user will lose “lives.” The Game Controller gives the user 15 lives in the beginning of the game.

After that, it creates the user interface in the top (and soon) bottom panel. These include labels and buttons that help the user throughout the game. On the top panel, there is the Rounds label, Lives label, and Money label. There is also the Start Round button and New Game button.

* The Rounds label indicates what round it is. There are 6 rounds total.
* The Lives label indicates how many lives the user has left. If it reaches zero lives, the user loses and must restart.
* The Money label indicates how much money the user has, but the money has no use as of now.
* The Start Round button begins the round and sends enemies towards the base. There are different combinations of enemies each round that gets progressively harder in round number. You cannot start another round during a round that you have not finished yet. After 5 rounds, the user wins.
* The New Game button starts the game over from scratch, at round 1.

1. **Handler Classes:** All of these handler classes are inner classes of the Game Controller class.
   1. **ButtonStyleHandlerPressed**
      1. This handler simply provides style for the button when it is clicked. It inverts the color of the buttons.
   2. **ButtonStyleHandlerReleased**
      1. Similar to the above handler, this handler changes the button colors back to normal when the user releases the mouse click.
   3. **EnemyFinishHandler**
      1. This handler deals with when enemies reach the end of the gravel path, AKA the base. This handler will take away lives if the enemy has health still. If it is the last enemy of the round, it will allow the user to move on to the next round. If the enemy takes away the last life of the user, the game is over.
   4. **EnemyClickedHandler**
      1. This handler deals with when enemies are CLICKED. Every click does 1 damage to the enemy. If it is the last health point of the enemy, it will kill the enemy, removing it from the game pane. If it is the last enemy killed of the round, it will allow the user to start the next round. If it is the last enemy killed in round 5, the game is won.