Design Document: Space Shooter

The premise of the game hasn’t changed drastically but has been fleshed out so the player has a goal to work towards. Also, some features had to be scaled back due to the difficulty of implementation. The premise of the game is now to reach the “safe zone” by avoiding the asteroids and enemy alien ships. The player can control the ship with arrow keys and shoot with the space button. There are up to 3 different levels, starting with the easiest level of just avoiding asteroids from level 1 up to having to avoid asteroids and aliens in levels 2 and 3. Though I originally wanted to have different types of enemies with different behaviors, due to difficulties of implementation, I wasn’t able to successfully program multiple types of enemies into the game.

This section will briefly introduce some main sections of the code. In the model package contains classes like GameObject, PlayerShip, Enemy, and so forth. GameObject is the base object class that all models derive from and contains attributes for position, imageFile, dimensions, and initialPosition. PlayerShip contains additional attributes like numLives, currentScore, and a ShootingObject (to support shooting abilities). Enemy contains a speed attribute and a method for movement. In the controller package contains classes such as Controller, PlayerController, EnemyController, ObstacleController, and Main. Controller is the base class that all controllers derive from and contains methods like setImage (to set up an ImageView), setViewPosition(to position the view), and resetPosition(to reset an objects position). The PlayerController contains methods such as setting up key listeners to control player movement, shooting the laser, and manipulating/displaying scores & lives. The EnemyController controls the enemy’s movements and will align them to designated areas in the scene. The ObstacleController is has code similar to the EnemyController. Main controls all the controllers and sets up the AnimationTimer to start/stop the game. The added Utility package is for utility classes such as dimension and position. The position class contains additional methods to help position objects for the various controllers.