Engineering 1

Group Assessment 2

Implementation Document

Cohort 2

Team 11

b) Explain how your code implements your architecture and requirements (incorporating your recorded changes for Assessment 2). Briefly explain any significant new features, e.g. non-primitive data types, significant algorithms or data structures. Give a systematic report of any significant changes made to the previous software, clearly justifying each change, and relating it to the requirements and architecture by pointing to relevant class names and requirement IDs. Note that, if a change has significant side effects, it needs a solid software engineering justification. State explicitly any of the features required for Assessment 2 that are not (fully) implemented. (10 marks, ≤ 4 pages)

Implementation

Difficulty

UR_LEVELS refers to the new requirement that the game must allow players to choose between different levels of difficulty: Easy, Normal and Hard.

There are 3 different difficulty levels implemented into the game. They are Easy, Medium (Normal) and Hard. The difficulty you wish to play on can be selected in the settings screen on the main menu and then that difficulty is implemented when you start a game.

When changing between difficulty settings three values are changed to different values depending on the selected difficulty level. BOAT_TARGET_SPEED which represents the AI boat's maximum speed and is set to the lowest value when easy difficulty is selected, is increased for medium and is highest when hard difficulty is selected. OBSTACLE_COUNT which represents the number of obstacles that are spawned randomly throughout the map, this is set to the lowest value (50) when on easy and is increased on medium to (100) and set to the max value (200) when on hard. The final variable changed when altering difficulty is POWER_UP_COUNT, this represents the number of powerups spawned in a game and is set to its highest value when on easy and is subsequently decreased when increasing difficulty to medium and hard.

Values for this are stored in Difficulty.java but used where necessary, including in AIBoat.java, and BoatRace.java.

Game Save

UR_SAVE denotes the requirement to provide facilities that would allow players to save the state of the game at any point and resume the saved game later

Power-ups

UR_POWER_UP is a new requirement to implement five power-up packs, which could be found floating down the river and be picked up by boats to improve some of their characteristics

There are 5 unique power-ups implemented into the game, they are Repair (refills a portion of the players lost durability), Stamina (refills a portion of the players lost stamina), Boost (Gives the player a temporary speed boost), Time (reduces the time taken on the current leg) and Teleport (jumps the player forward a set distance). When a boat collides with one of the powerups they collect it and its related effect is immediately applied to the player's boat. Each powerup has a unique look so that players can tell which powerup they are collecting.

Power-ups are implemented as an extension of the MovableObject class and their collision is implemented using the CollisionObject class. In this way power-ups act as an obstacle as they are spawned randomly throughout the map and float around with random direction until they are collided with by a player. The main difference is that when collided with instead of reducing the player's speed and durability instead they help the player.

This was implemented in the Powerup.java file.