Implementation

In order to fulfill the requirement UR9, we've made alterations to the ObstacleType and Boat classes. In the software given by the previous group, obstacles could be one of three different types, each doing different damage amounts to the player and having a unique speed. In our new program, we have added five new options to the ObstacleType enum. The difference is that these new "obstacles" do zero damage to the boats and can be easily identified as power-ups by their different textures. We then altered the "checkCollisions" function in the class Boat. Now, when a boat collides with an obstacle, the game checks what kind it is, obstacle or power-up with a switch statement. Then in the case that it is not a proper obstacle, and the player has no shield, it will apply the damage and speed decrease. Otherwise the program will apply the corresponding effect.

As was planned out in the concrete architecture, a new class named DifficultySelectScreen has been added which implements the libGDX screen class. This class acts in a similar manner to the main menu and boat selection screens. It contains a set of buttons, each of which takes the user to a different screen. In the previous software, the BoatSelectScreen passed an enum "BoatType" that would be then used in the following screen (MainGameScreen), but with the new requirement UR11 to allow the user to select a difficulty, the BoatType enum will reflect both the type of boat and the difficulty chosen. Therefore, we don't want to pass an enum after the BoatSelectScreen. The subsequent code we implemented instead passes a string called "type" with values such as "fast", "endurance", "strong" and "agile" through to the DifficultySelectScreen. It is then in this class that the difficulty is chosen and the combination of that with the type can be used to pass a BoatType enum to the MainGameScreen just as in the previous implementation. The difference being that the BoatType enum now features 12 possible enumerations which are combinations of boat type and difficulty (i.e. FASTEASY, AGILEHARD).