Section n: Requirements diagram

### n.1 Use case Diagrams

Por finalizar.

### n.1 Textual Description of the use cases

**Unique identifier:** CU1. Play level

**Context of use**

⦁ We start the level.

**Preconditions and activation**

⦁ The user is running the game and has already selected the boat.

**Guarantees of success or post-conditions**

⦁ The level is loaded, and the player can play without issues.

**Main scenario**

⦁ The player will be able to move around using the movement keys.

⦁ Along the course, obstacles appear, and the player must avoid them.

⦁ The player's aim is to reach the finish line in the shortest possible time and ultimately to be the first boat to do so.

**Alternative scenarios**

⦁ The game might crash whilst the player tries to access the game.

**Unique identifier:** CU2. Choosing the boat

**Context of use**

⦁ The user chooses the boat by using the control keys from a menu that is displayed on screen. This implies having the same boat on all 3 levels.

**Preconditions and activation**

⦁ The game has been started.

**Guarantees of success or post-conditions**

⦁ The player chooses the boat they want to play successfully.

**Main scenario**

⦁ The system loads and displays the different boats on the screen.

⦁ Each one has different characteristics, varying in style and stats.

⦁ The player will choose a different boat depending on his preference.

**Alternative scenarios**

⦁ The user exits the game without completing the action of choosing a custom boat.

**Unique identifier:** CU3. Start game

**Context of use**

⦁ When starting the program.

**Preconditions and activation**

⦁ Being on the start screen.

**Guarantees of success or post-conditions**

⦁ Once the boat is chosen, the level screen is loaded.

**Main scenario**

⦁ The player chooses to start a new game.

⦁ The system loads the first screen where the user chooses the boat to play with.

**Alternative scenarios**

⦁ An error occurs, and the player cannot start the game or finally decides not to play and closes the program.

**Unique identifier:**CU4. Control boat

**Context of use**

⦁ The user wants to control the boat.

**Preconditions and activation**

⦁ The boat has been selected (CU2) and the level has begun. Both previous cases have been successful.

**Guarantees of success or post-conditions**

⦁ The boat movement matches they keys stricken by the user.

**Main scenario**

⦁ The user presses the assigned key in the direction in which they wish to move.

⦁ The system checks the validity of the movement and performs the movement if allowed.

⦁ The system shows the change of position according to the user´s decisions and the validity of them.

⦁ The system continues checking if the control keys are pressed and the validity of them until the level comes to an end.

**Alternative scenarios**

⦁ The move indicated by the player is invalid. The system makes no move, displays the game screen, and waits for a new move.

⦁ The system is not able to recognize the key pressed by the user, displaying no movement.

**Unique identifier:** CU5. Use Powerups

**Context of use**

⦁ During the race, power-ups ups will be available for grabs if the player collides with them.

⦁ Being in a race.

**Guarantees of success or post-conditions**

⦁ Player gets a Powerup to use during the race changing for a given time the way the boat behaves and the way the game is played.

⦁ The system detects the selected Powerup and applies it accordingly.

**Alternative scenarios**

⦁ Error in the detection of collision with the powerup.

⦁ The powerup is not applied correctly or applied at all.

**Unique identifier:** CU6. Play mini game

**Context of use**

⦁ If the boat is destroyed, the player will have a chance to come back to life by playing the Simon says mini game.

⦁ Starts after boat is destroyed.

**Guarantees of success or post-conditions**

⦁ If you pass the mini game, you will return to life.

**Main scenario**

⦁ The player´s boat has been destroyed and they are given an opportunity to return to the game as they left it.

**Alternative scenarios**

⦁ The player loses the mini-game and the race ends.

⦁ The system fails to start the mini-game and simply ends the race.