LPOO - Final Project Intermediate Check-Point

1. Architecture Design
   1. UML
   2. Behavioural Aspects
   3. Design Patterns

* Singleton - This will be used in game class because it knows all but there should be only one game class accessible to all.
* Strategy – This will be used in different levels of game because the snake should have different behaviours.
* Template – This will be used to draw the world after it’s updated.

1. GUI Design
   1. Main Menu



Here we have a mock-up of the main

menu of the game. The game will start

when the player chooses one of the three

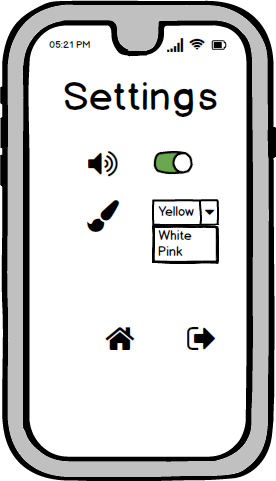
existent difficulties (beginner, intermediate

and impossible). In this menu we will also be able

to go to the settings or the scores menu, share

the game results on social media and exit the

game.

* 1. Settings Menu

In the settings menu we will have na option

to remove/add sound to the game. The player

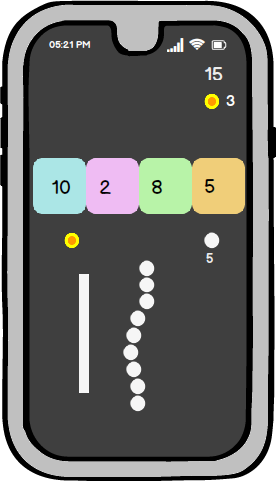
will also get to choose the color of the snake he

will be playing with.

After changing the settings, the player can return

to the main menu or just exit the game.

* 1. Game Mode



The objective of the game is to get the largest

Snake possible. When the snake collides with

a circle, the number of circles indicated will be

added to the snake body. Then, when colliding

with a square the snake loses the number of

circles shown inside the box. The player will

lose the game if the snake disappears (number

of circle = 0).

In the game, the player may also collect coins,

that will increase the final game score. The

number of coins gained and the current score

will be shown in the game.

* 1. Game Over



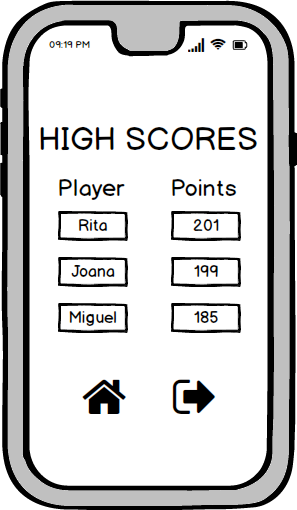
When the game is lost, the player can return

to the main menu, leave the game or play again.

the score will appear on the screen and a crown

will indicate if a new high score is reached.

* 1. Scores Menu



This menu will show the highest scores

obtained in the game. The player may

return to the main menu or leave the game.

1. Test Design – Unit tests
   1. Test collisions with the different objects of the game: squares, walls, circles and coins;
   2. Test if when the snake disappears (number of circles = 0), the game is over;
   3. Test if the snake gains the number of circles indicated when colliding with a circle;
   4. Test if the snake loses the number of circles indicated when colliding with a square and the square disappears;
   5. Test if the number of coins increases by one when colliding with a coin;
   6. Implement test for the diferente levels: beginner, intermediate and impossible;
   7. Test buttons behaviour (play, exit, settings, sound, select levels,…);
   8. Test snake movement;