8/18/2020

Fishy Waters

Video Game Architecture & Optimisation

Teodor Grigor – Student ID 19412013
[COMPANY NAME]

Contents

1. About Game

Fishy Waters is classified as a sport-racing game. As the main description, it has the following sentence: “Catch fish, sell them for gold and upgrade your kit to catch even more fish.”. The game has as the main character a man on a boat and he attempts to catch fish to improve its tools or to extend the area where he can catch fish.

The game contains water elements as well as ground elements. The boat can navigate on the water and based on the user progress new maps are unlocked with new features as well.

In my version of the game, the main character is a person who wears red and it is on a boat made of woods. The boat can navigate only on the water surface, the same as in the original game and in some places are stones in the water, therefore, the player can not attempt to navigate through there with the boat because it is impossible.

2. Game Objectives

The objective is to catch as many fish you can to sell them and get some coins, coins that you can use to improve the boat tools as well as the in-game features such as increasing the map.

To catch a fish the boat should be near one (the fish can be recognized very easy on the water – it is represented as a small rectangle with a blue lighter than the colour of the water) and the mini-game will automatically pop-up in the middle of the screen. The mini-game is made of a random series of arrow keys and its objective is to let the player press the relevant arrow key with the one shown in the mini-game. The number of keys is randomized and it can be from a series made of 3 keys to up to 6 arrow keys. In case that the user fails to press the key then the fish will be lost.

The shop screen objective is to let the player know what features/tools can be bought with coins, therefore, he/she will know exactly how many fish should be caught to be able to buy that item.

3. Game Mechanics

In the original game the player can navigate with the boat on the water anywhere as long as that part of the water is unlocked. In my game the principle is the same – the boat can not navigate outside of the water (the player can not attempt to navigate on the ground – in the game is represented as some nuances of green).

The mini-game automatically pop up when the user is near a fish – this means that the fish should be in the next square, a square that is near the boat location. The mini-game is automatically generated every time when it pop ups therefore the player can not attempt to memorize the sequence of the keys and the order that they appear in the mini-game because it won't help him/her to catch the fish. If the player attempts to press an arrow key then if it is the right one the key will change otherwise the mini-game will automatically close due to the wrong key pressed – this event is also happening when the player fails to press the requested arrow key.

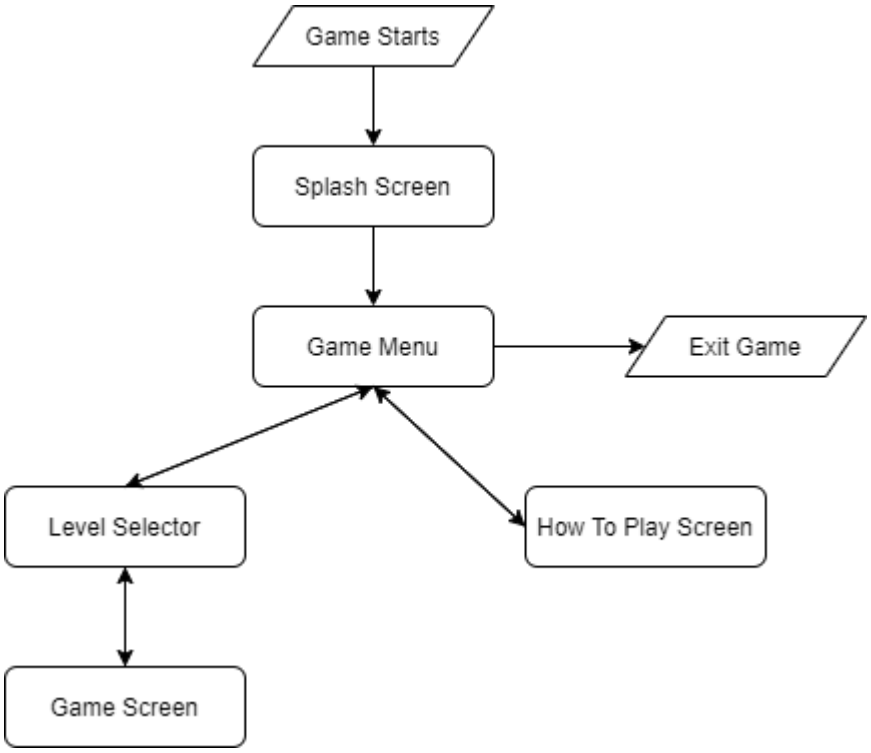
For the shop front the buttons are colored based on the player's coins as follows: if the player is able to buy that item (tool, upgrade or feature) then the button will be colored with green otherwise it will be red. The player can buy the entire map – this means that the map will be opened therefore the player can navigate in new areas to catch the fish, the player can increase the boat speed up to 6 times – this means that the player will be able to navigate on the water faster than at the beginning. The last available item is the player boat which can be upgraded therefore an advanced image of boat will appear (not available in the current version). It is to be noted that all the purchases made can not be restored or to get the coins back if you want to downgrade – this option is not available either.

4. Screen flow diagrams, Characters, NPC, Background design

The assets used in the game are created using the RectangleShape, therefore, all of them are primitive but the overall game design is of high quality – all the sprites were created by me so there is no need for referencing.

The backgrounds for all the screens are made of rectangles with the window size and with a default colour. The only screen that has a different background that can be seen with multiple colours is the game screen where the map is the background – the map is made of water, which is blue, grass, which can be of 2 colours of green –light green and a darker one, stone, which is made of 3 layers of grey, the fish, which is the same colour as the water but it is lighter, the harbour, which is coloured with yellow and the boat.

The boat contains orange and brown- for the boat, and red and pink for the human body, which can be seen on the boat.



5. Pseudo design

```
Run the game
If the game started then
    The splash screen will be shown
If the splash screen was shown for 2 seconds then
    The game menu will be shown
If the game menu is shown then
    If the Exit button is clicked then
        The game is closed
    If the How to play button is pressed then
        The Instruction screen is shown
    If the Play button is pressed then
        The Level selector screen will be shown
If the instruction screen is shown then
    If the Back button is pressed then
        The game menu will be shown
If the level screen selector is shown then
    If the Back button is pressed then
        The game menu will be shown
    If the Start New Game button is pressed then
        The game screen will be shown with the user progress reset to default
If the game screen is shown then
    The elements are shown in the game window
    If the arrow keys are pressed then
        If the next position is water then
            The boat will move into that direction
    Fish are generated at all the time
    If the player is near a fish then
        The fish minigame will start with random keys
    If the boat is in harbour then
        The shop screen will pop up
    If the back button is pressed then
        The level screen will be shown
```

6. Implementation of game

In the game were implemented the following items:

- A background – the background is based on colours
- A graphical character – the main character, the boat
- The boat can move by pressing the keyboard keys
- NPC are included – the fish
- The main character can interact with other NPCs – shop screen, fish
- In the game is present a coins system
- The main character can fully interact with the NPCs
- The game contains different features as well as in-game upgrades and new maps therefore it can be counted as more than 1 level

7. Testing

Objective	Test Case	Test steps	Expected result	Actual result	Pass/Fail	Pass/Fail	Pass/Fail	Pass/Fail	Pass/Fail	Pass/Fail	Pass/Fail	Pass/Fail	Pass/Fail	Severity	%Pass
Detect button click	Button	Attempt to click on the button	The click on the button should be detected only if the button was clicked	The button behaves as expected	Pass	Pass	Pass	Pass	Pass	Pass	Pass	Pass	Pass	N/A	100%
Read the content of a file	Files	Using the methods from the fstream attampt to read a file	The content of the file should be read as it is	The file was read properly	Pass	Pass	Pass	Pass	Pass	Pass	Pass	Pass	Pass	N/A	100%
Scene selection	Scene manager	Attempt to change the scene and show it	The scene should be changed without any errors and on drawing the selected scene should be drawn	The scene manager behaves as expected	Pass	Pass	Pass	Pass	Pass	Pass	Pass	Pass	Pass	N/A	100%
Level selection	Levels Screen	Based on the user progress the levels are shown properly	Only the levels unlocked should be shown in green	The levels screen behaves as expected	Pass	Pass	Pass	Pass	Pass	Pass	Pass	Pass	Pass	N/A	100%
Boat Movement	Boat	Using the arrow keys the boat should move in the selected direction	The boat should move without any problmes if in that direction the tile is water	The boat movement behaves as expected	Pass	Pass	Pass	Pass	Pass	Pass	Pass	Pass	Pass	N/A	100.00%
Fishing Mini-Game	Mini-Game	The minigame specifies randomly a key and the user should press it before it changes	If the game is completed the coins should increase otherwise the fish disappears	The minigame behaves as expected	Pass	Pass	Pass	Pass	Pass	Pass	Pass	Pass	Pass	N/A	100%
Buy items	Shop Screen	The user attempts to buy an item	The item should be bought only if the player has enough coins	If the user has enough money to buy that item more than once the game autobuys it until the player can not buy that item anymore	Pass	Fail	Pass	Fail	Fail	Pass	Fail	Fail	Pass	Critical	30%

8. Algorithm Optimisation Strategy

In the game, I have used different classes for each object. It can be seen that the game source code is structured in-game package and library package. Under the library package, there can be found multiple classes that were created with the purpose to create different UI elements such as the buttons, the level button, the font manager, the load image.

Under the game package can be found the classes for the game scenes and the classes for the character, NPC, game map. The fishy waters map is loaded from a .map file – the file can be viewed as plain text, into a 2D array. The map size is 40x40 and it is used in the game later on accessing the memory from pointers.

The boat movement is made by checking the targeted tile when the player attempts to move only therefore the game is moving smoothly without any bugs due to low memory. Since all the sprites are created using the SFML library – the RectangleShape, the sprites are loaded at the drawing time only.

9. Evaluation and future developments

During the development of the Fishy Waters assignment, especially the game part, I encountered some problems with the classes that were saved using pointers, but I managed to fix them without too much trouble. I can confirm that at this moment, after creating the game, I can manage and work with pointers at ease without doing basics errors.

For future developments, it would be better if the game will have new assets for the ground elements such as trees, woods, houses or even humans or animals. Also for the fish, it would be better if each fish will have their sprite and animated. At the moment the fish is represented as a light-blue rectangle therefore this games demonstrates more the mechanics used then a game that can be launched on the market.

Another improvement could be a bigger map with areas with their features such as different types of water – where the boat can not move very faster, or areas very dangerous with sharks, where the boat should have high resistance so it can survive to shark attacks.