Edinburgh Napier UNIVERSITY

Coursework Report – Part 2

Domenic Cassisi
40457750@live.napier.ac.uk
Edinburgh Napier University - Web Technologies (SET08101)

Abstract

This coursework report discusses the implementation of the game idea planned in phase 1. The HTML pages were implemented almost exactly as planned. However, the actual game implementation differs a lot from the concept. It lacks in story, complexity and functions. The project was also influenced by the current corona situation which led me to leave Scotland.

Keywords – Web Technologies, Adventure Game, HTML, CSS, JavaScript

1 Implementation Overview

In this chapter I would like to briefly discuss the implementation of the project. The HTML pages and the game are presented separately.

1.1 Website

As originally planned, I have developed 3 pages: Home, Game and Contact.

The Home page is the most "complex" one which is usually presented first to the user. Its task is to explain the type of the game. The Home page looks almost exactly as I planned it in the first half of this coursework. I just added one more section to the site.

The Game page is simple and implemented as planned. Besides header and footer, it just contains the game container with the WebGL context.

The contact page slightly differs from the planned version. Rather than using a layout that presents everything in one row, I used multiple rows for different (semantic) sections.

1.2 Game

The game did not become what I first planned to develop. The game lacks in size, story and complexity. However, it also contains a lot of the fundamentals required to build a more complex game.

Game World I did build a simple 2D tile based game world using the map editor Tiled. This map was then exported as JSON file and loaded into the game. Actually I planned to create a bigger game world.

Player Movement Player movement has been implemented as planned. The player can walk in all four directions and even diagonally where always a proper walk animation is played. However, I am missing collision detection with objects in the game world. There is only collision detection with the game world's boundaries.

Story The story has been kept short and simple. However, it can easily be extended and made more complex if necessary. The story is defined in an own file (constant) which is then loaded as (JSON) object into the game. The story consists of story points whereas each has its own id, text and options. Each option references the story point that comes next.

2 Things to improve?

If I had more time, I would work on a more complex story first. Beside that, I would extend the game in functionality by adding collision detection, other ways of interacting and more. The game world itself could also become bigger.

Furthermore, I would want to better organize my code. Since this was the first time working with any kind of JavaScript framework, there is for sure a better way to structure and organize the game code.

3 Challenges

The biggest challenge has been working with JavaScript. Even though I used a framework for the game which made things a lot easier, it was still confusing sometimes. In particular, the fact that JavaScript does not care about types made the start a little difficult for me.

On the other hand, this assignment allowed me to better understand the world of web development, especially client-side development. I will for sure make use of the learned knowledge and technologies in future projects.

4 Overall Situation

Due to the Corona situation I had to leave Scotland and return to Germany. For a long time it was unclear whether we could even finish the modules. In the end, a solution was found, but a lot of time was lost in between, which of course had an impact on the project result. I still hope that the result is acceptable.

5 Frameworks and Assets

Frameworks and Tools:

- Tiled to build a 2D tile based game world.
- Phaser3 for game canvas, sprites, animations and player movement and sound

- jQuery for easier interaction with the DOM. Used for smooth scrool effect.

Assets and Tutorials

- OpenGameArt for graphics and sound
- FlamingText as logo generator
- YouTube for a simple basic tutorial on how to use Phaser