Challenges Faced When Developing Finger Run

When I read the specification of the web page, the idea of building a game that makes people exercise their fingers came to my head. This was when the design of Finger Run’s page layout and coding began. At first, I had no idea how to approach the project. I started writing the pseudo code but I had to start again as the first pseudo code did not make sense, meaning I was unable to turn my idea into a basic process of how it works. This is when I decided to make research based on keyboard keypresses in JavaScript. I came across the ‘Eloquent JavaScript’[[1]](#footnote-1) book online, which demonstrated one of the simplest approaches to using JavaScript Event Handlers with keyboard keypresses. I followed the tutorial and applied it to my algorithm, which I have managed to complete after a second approach as it was making sense and appeared developable.

Furthermore, I started designing the basic layout of the page. When it comes to design, I work with colours as I develop the project. This is the reason why I decided to sketch the website. After I was done with the design aspect of the game. I began implementing the algorithm and plan. I have first struggled with making the keypresses work despite spending time on trying to understand how they operate. I looked at the tutorial again, read about Event Handlers on the ‘JavaScript and JQuery Interactive Front-end Web Development’ book, and watched some YouTube tutorials. After persisting for about a day or two, I managed to make one of the keypresses work by changing the attribute of image in JavaScript which affects the layout of the image with the new attribute. In addition to that, I also learnt about CSS transitions to show the movement of the ball.

After making the ball responsive to the keypresses of the different arrows, I used an if/else statement to give a player feedback based on their performance. This was not hard at all as I comprehend this part of JavaScript from previous assignments.

When I wrote the algorithm for the timer of the game, I planned to use an if/else statement but during the process of building, I decided to try a loop which did not quite work so I went back to my first approach and managed to get the timer working.

On the other hand, the red background (warning colour) of the timer did not change even after the player restarts the game. It only changes back to its original after refreshing the page. It took me quite a while to fix that bug. I checked online but it was difficult to find the direct solution I was looking for. This was when I decided to set the background colour to light grey in the play function. This means that the background will automatically change to light grey as soon as the game starts and the timer starts running. In contrast, its colour changes to red in the last three seconds of the timer.

Overall, this game forced me to think as junior programmer. I came up with the idea of how the game works so there were so many decisions I had to make such as the speed of the ball or lasting period of the game. Such simple decisions have been made during the process of building the game and this is why they have not been reflected in the algorithm.

1. Marijn Haverbeke, Elloquent JavaScript (Second Edition) (2014). [↑](#footnote-ref-1)