

**Semester Project 2**

**Of**

**Remano Victor**

**Final technical report.**

# **Introduction:**

## **Brief:**

The goal of the assignment was to create a snakes and ladders style board game with a game of thrones theme. The game needed to be functional and aesthetically pleasing. The game required a minimum of 3 pages:

- Character select page
- Board game page
- Winner page

The game needed to include 10 icons representing the characters or house sigils as well as a board to play on. The final would have the token of the winner and this page should also include an animation if time allowed.

The game did not need to be responsive as this would be more complicated than needed.

## Body:

### INDEX:

For the design I've opted to go with a darker theme. The general feel of the series is set in medi-evil times and is quite a dark setting both in atmosphere and in lighting. With this in mind I've kept the game images darker except for tokens as my goal was to make these stand out against a dark background.

The game layout is built with bootstrap which allowed me seamless placing of elements onto the document instead of using CSS-grid and having to fine tune elements to align them. Bootstrap was also a quicker option and less time consuming.

The color contrast over the entire page is AA compliant even with the video backgrounds.

The home page has a minimal layout so as to not overcrowd the visitors view. While the video background is fluid and non invasive. On this page you will find instructions on how the game is played as well as a link to the character select page.

I've chosen to go with the font Hodor for headings. I feel this font sits really well with the theme. For the body I have chosen to go with a sans serif font face called Josefin sans. The combination between Hodor and Josefin brings a well balanced feel to the website.

### CHARACTER SELECT:

The design of the character select page, in keeping with the index page, is simple.

The character tokens are arranged into 2 rows of 5. I used image masking in photoshop to achieve the round shape. They also have a width of 100% and a hover effect to let users know they can be interacted with. When clicked, a fetch statement is run in javascript via a function that takes the character name as a parameter. The unique character id associated with each character is then concatenated to the fetch API address and thus gives me the information of the particular character. The information is then displayed in a modal which can be closed by simply clicking outside the modal box. The information in the modal is created through javascript to streamline code and produce a less cluttered HTML. Thus making the code easier to read. I opted with a modal as this had a better effect and brought focus to the information that was being displayed. Inside the modal will be a button with another function attached to it. This button will be used to select a character. The character name and token is then saved into session storage to be used at a later stage. When 2 characters have been chosen, the page automatically moves onto the net page which is the boardgame page.

## BOARD GAME

On the board game page is where the real magic happens. On this page is where all the javascript runs. First thing we do is take the information saved in session storage and place them into relevant divs to display player one and player two.

The players and traps are stored as objects inside a store file. The players have key value pairs for tile number, turn, token and title. The movement of the tokens is achieved by adding a dice roll number which we got from a `math.random` method to the player tile number. This will then move the player forward and vice versa if the player lands on a trap. We can then place the token inside the tile with an ID equal to the number.

Traps are triggered when a players' tile number matches the trap number. This sets off a function which triggers the respective trap object

properties associated with the specific trap. I have achieved this by using a switch statement. I did have an if else statement at first, but later changed it as i found that switch statements, although they do the same thing, look neater and are easier to read, especially when working with many lines of code.

The tile design on the board game is just an outline with a transparent background. This gives a fluid feel to the board and makes it seem more a part of the entire website instead of a board that seems out of place

The player tokens are displayed at the top of the page next to the dice. These are fetched from session storage. When it is a theor turn, the players' token will have a border color which indicates visually who's turn it is. The border is clearly visible against the rest of the page.

The current player is determined by running a function that checks for a boolean within the player object in the store.js file and then decides from there. When withe turn ends, the same function then switches the boolean values thus ending the players turn.

When a player reaches tile number 30 or above, a pop up will announce the winner and the page is automatically redirected to the next page.

## WINNER PAGE

After reaching the finish tile, players are redirected to the winner page. On this page, the winners token is displayed. The token is rotated using CSS keyframes. I have also included a webkit for safari and firefox for the animation. This is to ensure that the animation is rendered across all browsers. The content on this page is again minimal and is mostly rendered using javascript. The name of the player is fetched from session storage and displayed within the paragraph.

## Conclusion:

In conclusion, I personally feel that I have learned an immense amount in a very short time. This assignment had taught me to think critically as well as logically. At first the task seemed huge and I did not know where to start. But after thinking logically, I was able to break down the tasks that needed to be done. I also used a trello board to help keep track of what needed to be done and what the deadlines were for each feature.

I am extremely impressed by what I have achieved through this assignment. I am aware that although my code is efficient, there are always better ways of doing things. As such, I will continue to revisit my code and improve on it as time goes by.