

FALL '18



SHIELDS & SNAKES
BOARD GAME

HALLDOR VALUR STEFANSSON
NOROFF
Front-end Developement

Table of Contents

Introduction	2
Organization.....	2
Method.....	4
The Website	4
Colour Scheme for the Website	4
Font Family	5
Name of the Game	5
Logo	6
Character Page	6
Characters	6
Functionality	8
Design	8
The Game Board.....	9
Spaces and Traps.....	9
Functionality	10
Design	10
Game Pieces.....	10
Functionality	11
Design	11
The Dice.....	11
Design	11
Functionality	12
Winning Screen.....	12
Functionality	12
Design	12
User testing	13
Rules	14
Results.....	14

Game of Thrones Report

Introduction

The aim for this project was to create an **online board game** inspired by **Game of Thrones**, meant to include a character page, a board game page and a final page using **HTML, CSS and JavaScript**. My goal was to create a **traditional board game** which could be enjoyed by most age groups. I wanted all pages to be clear and understandable, and the design to represent the darkness and mystical elements of the show, while placing emphasis on key elements, such as buttons etc. The whole project is available on GitHub.com at;

<https://github.com/doristef/SemesterProject2>

Organization

In the beginning stages of the project I aim to get familiar with the show, get some ideas, **sketch up** (*image 1.1 to 1.8*) my pages and visualize the overall theme of the website. I will also write down a **list of the functions** I need, and then slowly work through them. I plan to use **Illustrator** for all drawings/images, **Visual Studio Code** for all programming and **Adobe Color CC** to choose colours. My **primary goal** is to create a functional page for choosing a character and having them move across my board. For that I have to write the following **functions**; choose characters, throw dice, move character, board trap, as well as design icons for these elements. However, I will continue to review all my designs throughout the process and do not intend to put too much time into design in the beginning stages. My **main goal** will be functionality and thereafter I will finalize the design. Other elements I will include in my game are e.g. main logo and favicon.



Image 1-1



Image 1-2

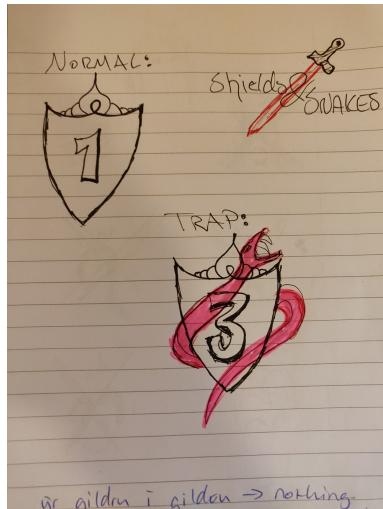


Image 1-3

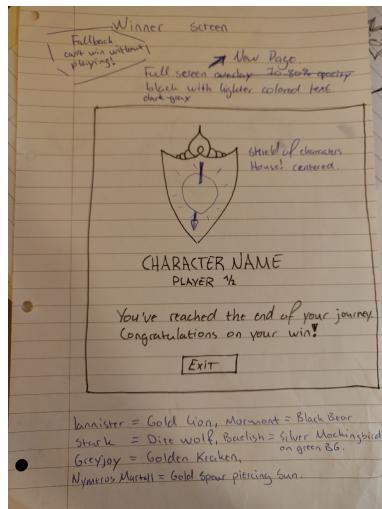


Image 1-4

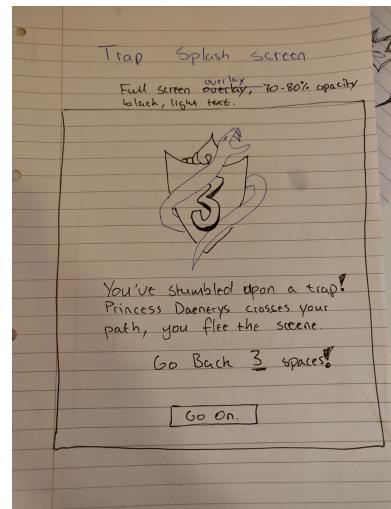


Image 1-5

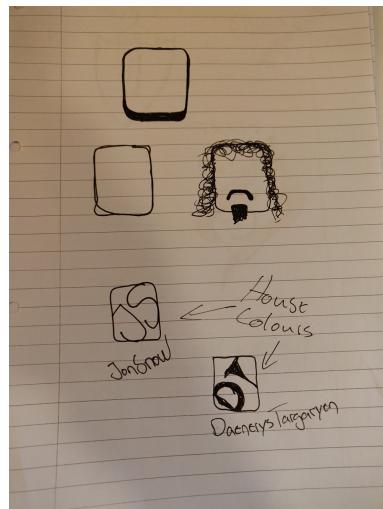


Image 1-6

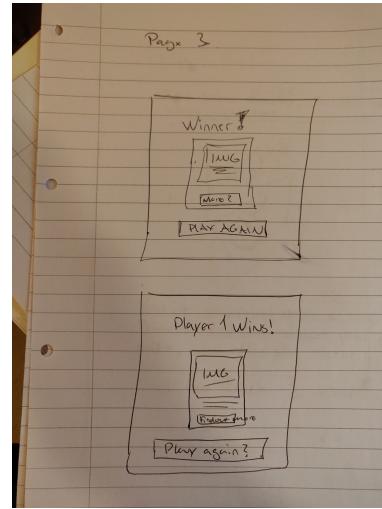


Image 1-7

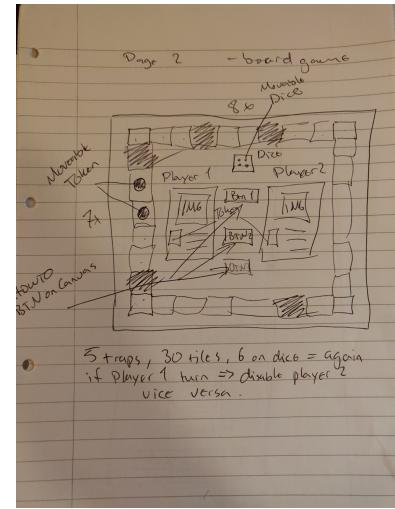


Image 1-8

Method

To start the project, I wrote down my ideas, listed some key elements and started sketching the layout of my board. I spent some time watching various tutorials on **Lynda.com**, read about game programming at **scotch.io** and **stackOverflow.com**, as well as reviewing similar websites suggested by **Google**.

I knew I would underestimate the amount of code I would have to write, as well as how time-consuming it would be, so I wrote my code in chunks, to be able to reach certain milestones. This proved to be extremely beneficial and kept me motivated through the entire process, as well as providing better overview and organization. **My milestones** were to write the following functions; choose two players, create the board, throw the dice, add random traps, move players on the board and redirecting to the winning page. **Additional functions** were to alternate between different characters before starting the game, seeing additional information for each character, play game splash/overlay screen after choosing two players, splash/overlay screen when landing on a trap, on-screen messages for smaller activities (e.g. throwing a six, throwing a second six, landing on a trap after throwing a six), on-screen indicator (i.e. colour) to differentiate between active and inactive players, different image for traps spaces on the board. I stuck to my plan and used only **Visual Studio Code** for all my programming.

For my design I created a logo for the front page, favicon, character face images, board spaces (including traps), throw dice button and other buttons and splash/overlay screen images (e.g. shield with swords). All images and icons were created using **Illustrator**, while I used various programs to find my colours.

The Website

Colour Scheme for the Website

At the beginning of the project I briefly researched the show, which gave me the impression of a dystopian world from the middle ages filled with mythical creatures. Therefore, I felt that a **darker colour scheme** would better capture the shows medieval vibe. At first, I felt that a blue accent colour felt strong and fitted well into the website, however after adding in darker colours for the background it felt washed out. For that reason, I chose a **bright yellow** to accentuate key elements for the “character page” such as the “choose”

button. I still wanted to include the colour blue and went with a greyish-blue in various hues for the background. I found all my colours using **Adobe Color CC**, **ColourLovers.com** and **Palleton.com**, but I tweaked all my colours using **Illustrator**. I maintained the same colour scheme in all the pages for my board game; dark background accompanied by stronger and brighter colours to indicate which elements were the most important. These colours were bright yellow, deep red, light blue and apple green.

Font Family

The fonts used throughout the website are **Cinzel Decorative**, **Cinzel** and **Lato** found on **Google Fonts**. Cinzel Decorative was used for all headings (H1-6) throughout the website, as well as for on-screen messages in the board game. I chose this font as the designer, Natanael Gama, was inspired by first century roman inscriptions meant to represent the ancient history of the latin alphabet while being modern¹. Cinzel as only used for numbers on the board, as the numeric font from Cinzel Decorative was more roman inspired while Cinzel uses contemporary numbers which are easier to read². Lato was used for the body copy, it was designed by Lukasz Dziedzic³. This font was chosen because it is easily readable, does not steal attention but is still sleek and elegant.

Name of the Game

What's the name of the game? Does it mean anything to you? (Do I need to reference ABBA for this?). One of the biggest struggles for me was finding a name for my board game. This might be because I'm not familiar with the show nor the books. I finally decided on the name "Shields & Snakes" as icons for both shields and snakes are highly important and recurring through the game.

¹ Gama, A. (n.d.). *Cinzel Decorative*. Retreived from <https://fonts.google.com/specimen/Cinzel+Decorative>

² Gama, A. (n.d.). *Cinzel*. Retreived from <https://fonts.google.com/specimen/Cinzel>

³ Dziedzic, L. (n.d.). *Lato*. Retreived from <https://fonts.google.com/specimen/Lato>

Logo

For the board game's **main logo**, I contemplated an image of a big sword, independently standing straight up from the ground. However, after deciding on a name I felt that the main logo would better represent and increase memorability if it reminded players of the game. Therefore, I chose a logo similar to **the traps** and directly linked to the name "**Shields & Snakes**"; a snake wrapping itself around a large shield with a star. Further, the logo is closely linked to the image used for the **favicon**; a red shield with a dark star in the middle. I used the same colours for the logo, favicon and the board spaces to strengthen their resemblance. The logo is placed between a text; Shields, Snakes and the text is white in order to make it more visible.



Character Page

In this chapter I will discuss the characters chosen to represent players in the game, their function and design.

Characters

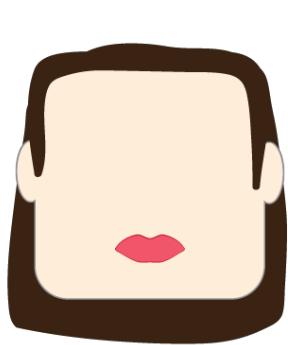
The following chapter will list the characters I chose, their design and function. Even though I have not watched Game of Thrones I'm aware of the show's popularity, therefore I instantly knew that I would have to create my own images. When creating the characters for my game I browsed through images of some of the show's main and/or most influential characters⁴. To represent them I created icons for their faces, emphasizing each character's main features to make them easily recognizable. I used these images for the character page, where players could choose their desired character to represent them in the game. The characters, and the character traits, chosen were **Jon Snow**; beard and curly hair,

⁴ Owen, P. (2017). *All 49 'Game of Thrones' Main Characters, Ranked Worst to Best (Photos)*. Retrieved from <https://www.thewrap.com/all-47-game-of-thrones-main-characters-ranked-photos/>

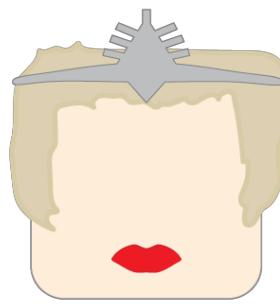
Daenerys Targaryen; icy blond hair, **Cersei Lannister**; short ashy blond hair and crown, **Arya Stark**; dark, slicked-back hair, **Jaime Lannister**; short hair with lush waves and scars, **Sansa Stark**; side swept red hair, **Tyrion Lannister**; thick messy hair and beard, **Theon Greyjoy**; short and ruffled hair, **Jorah Mormont**; fine, well combed hair and light full beard, **Joffrey Baratheon**; short ashy blond hair and embellished crown. Female characters were further distinguished from male characters by including bold red lips. All characters share the same skin colour, while other colours were chosen specifically for each character.

Other information which I included on this page was character names and a short introduction, i.e. description, of each character such as gender, title, birthdate, date of death, culture and which house they belong to. All information regarding the characters were fetched from the **API of Ice and Fire**⁵.

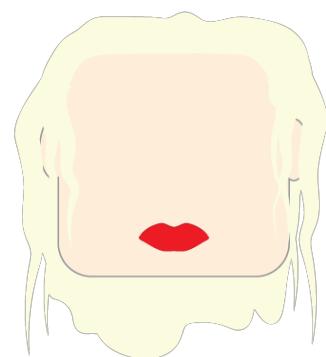
Character Images;



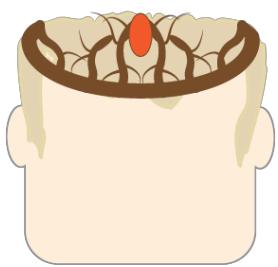
Arya Stark



Cersei Lannister



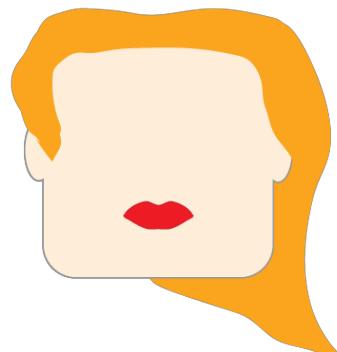
Daenerys Targaryen



Joffrey Baratheon



Jon Snow



Sansa Stark

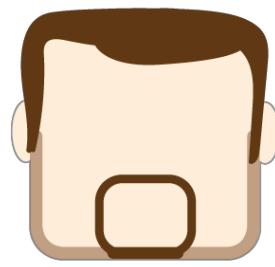
⁵ Skoog, J. (2018). *An API of Ice And Fire*. Retreived from <https://anapioficeandfire.com/>



Theon Greyjoy



Tyrion Lannister



Jorah Mormont



Jaime Lannister

Functionality

Two buttons accompany each character, “more” and “choose”. When first selecting a character, the whole field becomes darker while simultaneously adding a text indicating which players has been chosen (i.e. player 1). After choosing a second character the whole screen becomes dark and only displays **two new buttons**; “play the game” and “choose other characters?”. Furthermore, it is possible to **deselect** a previously chosen character.

Design

Each character is set up in a **bootstrap card**, the character names appear in the card header whereas the card body contains the information about the characters and is toggled by the more button. The text appearing in the body is split by bright yellow lines. The image and both the “more” and “choose” buttons are inside the card but outside the body. **The colour scheme** follows the design of the whole website, a dark background and a bright yellow accent colour.

The Game Board

In this chapter I will first discuss the creation of the board itself, secondly both spaces and traps, third representation of players and lastly the dice. **The board** itself was drawn inside **Canvas**. I opted for a traditional look and chose **a square**. After coding it for the first time, and getting it to work, I still had problems moving the players along the board. Then I realized that it would be easier to mark all the spaces and add them to **an Array**, that way I could access each space independently. I decided to first get my code up and running before finalizing the style for my board game. However, I did some preliminary sketches for each component, i.e. style of the tiles, colour, players and etc., throughout the whole process. The gameboard consists of 30 spaces, including five traps.

Spaces and Traps

One of my ideas was to make all the spaces look as **shields**, with some decorations, and even choosing different colours for different space, such as the first and last spaces. The spaces that had traps have **snakes** surrounding them (which I sincerely hope exist in GoT). **The traps** will be placed randomly on the gameboard, changing with every new game. When a game piece lands on a trap a large image of the space, along with a short text describing the nature of the trap and where the game piece will land, will be shown on an **overlay/splash screen**.



Normal Space



Final Space



Trap Space

Functionality

The first function was to create the board, I used a square. I set up four sides from a layout Array which assigns length of each side, moreover each space is added to the space Array, including all relevant values. Second function is to shuffle the Trap Array using The Durstenfeld Shuffle, which treat one part of my Array as shuffle while the other is treated as unshuffled⁶. To shuffle, a random number is generated and then the place of the element at a random index is exchanged with the current index. The third function was to randomly add traps to the Space Array. I did not want the traps to be too close to the start nor the finish point, or side by side.

Design

The spaces were drawn as shields. Each shield has a dark outline and white numbering, to make them stand out and be easily viewable. I chose the colour red, in two different hues, for all spaces with the exception of the winning space. The last space is slightly larger than other spaces and was give the colour green in two different hues to make it clear where the game ended as well as for a more visual appealing look. Additionally, to differentiate between other spaces and the traps, a green snake is seen wrapping itself around a shield.

Game Pieces

After having played the game a few times, I saw that the movement of the game pieces was unsatisfactory as the pieces only moved once according to the dice, which made it difficult to follow along with the game and made it seem less realistic. I decided to add a delay after each movement according to the number displayed on the dice. Set Timeout did not work, but after having browsed through Google and StackOverflow I found others who had experienced the same problem. The trick was to create the movement inside a self-invoking (recursive) function, which means I had to rearrange and change my previously written code. In the end both game pieces move smoothly along the board in a more realistic manner. Further, to represent the game pieces an image, and name, of the chosen

⁶ Holst, L. (2012). *The Durstenfeld Shuffle*. Retrieved from <https://stackoverflow.com/questions/2450954/how-to-randomize-shuffle-a-javascript-array>

character appear in the left and right corner, each marked with a colour corresponding to the colour of the game pieces and the line appearing under the “throw dice” button.

Functionality

Two functions were needed for the game pieces. **First**, the function for **start character** which reads from the player variable and places the game piece on the right space. **Secondly**, the function to **move character** which takes in the dice number and moves the game piece accordingly on the board. Next the function writes to the player variable **the location** of the game piece then checks which space the game piece has landed on and later if this space is a **trap**, to which a trap message (overlay) will appear accordingly. Then it checks if the player has thrown **a six**, or a double six, and presents the correct message. As well as checking if the game piece has reached or moved past **the last space**, which calls for the winning function.

Design

For the **game pieces** I went with a **basic design**, to increase visibility as the spaces already have their own design. I felt that if I created a more complex game piece it would not be as easy to follow them along the board. For me **visibility** was more important than an elaborate design. I created a slightly **rounded square** that included a **white number** indicating which player had been assigned to which game piece. To further distinguish between the two, I chose two contrasting colours, a **bright yellow** for player 1 and for player 2 a **bright blue colour**. As mentioned before, these two colours can also be seen below the “throw” dice button and above the images, presenting the names of the chosen characters.

The Dice

Design

To draw the images used for the dice I used **Illustrator**, and to show a random dice inside the canvas I used **JavaScript**. The colours used were white and dark grey, for a classic look.

Functionality

I created **six images** of a dice, displaying numbers one to six. **Four functions** are needed to throw the dice. **First**, is to create **a button** for the players to click, as well as **an indicator** (a blue or yellow line), representing the players. **Second**, is to create a **random number** between one and six. **Third**, is to **display** an image of the correct dice and last is find **which player** is throwing and then to run the functions in the **correct order**.

Winning Screen

In the following chapter I will discuss the design and functionality for the screen appearing after a player has won the game. When a player wins, a similar layout to the ones following a trap will appear. A large image of **a shield** with two swords crossing each other. I decided to **delay the transfer** to the winning screen a bit, as I feel the players need some time to realize where their game piece has landed before being shown the winning screen.

Functionality

To call out the **winning screen** after a player has landed on or moved past the final space, **one function** is needed. One function that is called with the player number (player1 or 2) which then writes to the **browser memory** (i.e. session storage) the players number, name and colour. Next, the browsers **window location** is refreshed redirecting the browser to the winning page. Once on the winner page, it reads the browser memory for the players number, name and colour, writing the results out. I also included a button “**Play Again?**” redirecting players to the main page.

Design

The **shield** appearing on the winning page shares its design with other shields presented in the board game. The shield represents all the players, I did have an idea to make individual shield to represent each house, but their design would have been too time consuming for such a small detail, although it would have been a nice extra touch. As previously mentioned, the shield is similar to the other shields in the game, but the image is much **larger**. I chose the colour **green**, in two different hues, to establish a connection to the

last space on the board. I added the image of two crossing **swords** in the middle of the shield. A white text with the winning character's name appears, as well as name of the player (e.g. Player 1) in the same colour previously assigned to them. The **yellow** colour of the “**play again**” button clearly stands out, while not taking too much focus of the shield.

User testing

I convinced some **family members and friends** to play the board game regularly from version 1. V1 looked very raw but was enough to test all the features. I also asked each person to send in their own **ideas** on how the game should look, such as what they felt was missing. **My mother** tested out the game for me, which proved to be very challenging, as she is not very **tech friendly**. However, I do feel it is important to receive feedback from an **older audience**, as it often helps me to notice which elements should be clearer. My 5-year-old son tested the game as well throughout the entire process, and boy did he enjoy it! He even challenged his mom and younger brother. He also pointed out that he wanted the game to be available on **mobile devices**. The game is first and foremost meant for desktop and most types of tablets, but maybe I make it responsive on mobile devices. It does work on mobile devices though, but the user needs to zoom-out the board page to have a clear view.

My **main obstacles** were to fix the “**double sixes**” function, but the wrong player was assigned the “has thrown a double” variable. I resolved this by making sure that the doubles variable (which notifies all double six throws) must be reset to zero following each player turn. Another problem I experienced was that all my **traps appeared side by side**, I was able to fix this by checking the spaces placed before and after a trap so that no traps would lie next to each other. Furthermore, the **trap overlay** was too big for a smaller screen but was resolved by switching to percentage sizes instead of pixels. Lastly, when **game pieces** were assigned to the same space, only one player would remain visible, this proved to be an easy one to fix by making sure that one would be placed slightly below the other which would in turn be place slightly above.

Rules

As a finishing touch I decided to include a page with the rules of my game. This page includes some of my icons, such as a red shield, trap shield and a dice. I felt it would be good for potential players to be able to read over the rules before playing the game or if anything becomes unclear.

Results

The game works **as planned**, players are able to choose between characters, game pieces move correctly across the board, all traps work as planned, when rolling a six the player get a second turn and once a player has reached the last space they are transferred to a winning screen. I created all the necessary logos, icons and images needed for a fun game reminding of Game of Thrones. I feel that I have **met all my goals** for this assignment. My board game works **for all devices** although it does look it's best on larger screens. I have received **positive feedback** from all of the people I asked to review my game, my target group seems to be a younger crowd and was a big hit with my five-year-old son. However, if I were to create another board game, I would do a few things differently. One thing I might have **done differently** would be to create different icons for "traps" that were good, i.e. pushing the game piece further along the board. All in all, I feel my board game is a good old classic, easily comprehensible and enjoyable for all ages.