Text, calendar

Description automatically generated

Implementation Report

Juan Espares

44317962 | DECO1400 | 30/5/2022

**Implementation Summary**

The website was implemented with all pages consisting of the same navbar and footer. This HTML code was duplicated across all webpages except for the signup and login pages which had no footer. Starting with the navbar, it is fixed to the top of the page giving users easy access to different webpages of the website. This was done using “position : sticky” in CSS. Media queries were used to style different screen sizes for mobile and tablets. When the screen is minimized to a smaller width, the navbar menu items will disappear and a hamburger icon is introduced. Toggling this hamburger menu icon will enable the menu items to appear and reappear. When the menu items are displayed, it is repositioned into columns with “flex-direction: columns” since originally it is in “display : flex”. Javacsript (js) was used to toggle the hamburger icon by having an event handler that responds to the user clicking on it. When the user clicks and activates the icon, the ‘navbar-links’ class will be active and hence displayed to the screen.

The testimonials section in the home page was implemented using the idea of a carousel. 3 blocks are displayed at a time with each block consisting a review by a user. The left and right arrows on the side allow for navigating left or right to show more reviews. The user is also able to drag to scroll through this circular type data structure. It basically mimics a doubly circular linked list. The implementation of this feature relies mainly on js which adopted a swiper implementation using jQuery. The function initParadoxyWay() will always run since the HTML class .testimonials-carousel will have content (see appendix fig.1). This is a div wrapping all content including reviews for the testimonials.

Media queries were used to adjust the responsiveness of the webpages. This was implemented for the homepage with the navbar and footer across all pages. The footer was organized into 3 columns in HTML wrapped by a .footer-row class. This was displayed using flexbox with “flex-wrap : wrap”. The width is adjusted to 50% when the window is resized to smaller screens. This changes the displayed footer into 2 columns instead of 3 with the third column underneath (see comparisons in appendix between fig.2 and fig.3).

The login and signup pages were implemented with the same design template. A background image on the right with all the fields of the form to the right of the page. This is essentially a 2 column design. The Legends page which holds content is also designed with a 2 column design. The content pages (Legends, Loadouts and Mechanical Guide) have the same layout, with alternating left and right between images/videos and text. For large body paragraphs, the user has the option to toggle a ‘read more…’ button to extend the height of the container. Initially the height is only tall enough to view a few lines of the content. After toggling read more, the container is extended. This is achieved using js. When toggled, display will be set to inline to reveal the content. The user also has the option to switch back with read less… to minimize the content displayed back to original.

The boosting page is the only other page besides the main page which does not follow a 2 column layout. Instead, most of the content is positioned centrally. An accordion was used to implement the frequently asked questions section. It reveals information similar to the ‘read more/ read less feature’. This time, a plus and minus icon shows the state of each question block.

The rank selection in the boosting page was implemented with js to increase (+) and decrease (-) the rank value. It is ranged from 0 to 10,000 with increments and decrements of 100. The user can also type the number manually but it will be capped to 10,000 if the input number exceeds 10,000. jQuery was used for incerementValue() and decrementValue() where the + activates the incrementValue() and the – activates the decrementValue() on click (see appendix fig.4).

**Challenges faced / lessons learnt**

One of the main challenges I faced when implementing the website was styling of the signup/login pages. Prototyping them was easy but using CSS, I had to adjust the positions, width, height, margins and padding in order for it to look the way I wanted to on screen. I learnt to use flexbox to help positioning blocks much easier. However, this was a learning experience for me to utilize div containers properly within HTML.

I was new to javascript and was also not accustomed to write functions for event handlers. I had to get my head around arrow functions and function expressions with js to really maximise efficiency. I also learnt how to use conditional operators in JavaScript to implement an if true and if false block in one line (see appendix fig. 5).

Time was also another factor that I should’ve considered more heavily. As such, I was not able to implement 2 content pages and the contact page. However, these pages aren’t unique to the ones I’ve already implemented and follow the same layout/style only with different content. Styling took more time than I realized and making sure I was overriding the correct classes from the HTML block with similar HTML code was also something I found challenging. I have learnt that commenting and reusing of code should be done better the next time I develop a website to make life easier for myself; especially when there are multiple pages with similar styling.

Using media queries to adjust the responsiveness of the website was something I found tricky to implement. I had to use trial and error to figure out which div containers’ width I needed to adjust. I was not able to implement responsiveness for all pages especially the signup and login pages.

**Evaluation of Website (Nielsen’s Ten Usability Heuristics)**

The final implementation of this website has slight differences to the design but it still follows Nielsen’s 10 usability heuristics for user interface design.

There are hover effects for the navbar and footer which highlights what the user is about to click. This demonstrates “Visibility of system status” as feedback is given to the user instantly as they navigate along the website. Another example of this in the homepage is when the user hovers over one of the statistics section’s images. This dynamic effect relays to the user what they are doing to the website (i.e. hovering over a specific statistic). The navbar remaining fixed at the top of the screen allows for “Flexibility and efficiency of use”. It is easily accessed wherever the user navigates to across different webpages and in any webpage itself.

The layout of the website as mentioned above follows either a 3 column where the content is positioned centrally or a 2 column where the content will alternate left and right as the user scrolls down the page. This is to avoid clutter on the webpage and makes it look organized. This follows “Aesthetic and minimalistic design” according to one of Nielsen’s usability heuristics. In addition, the read more / read less feature and the accordion in the Boosting page hides information and only shows them when needed by the user. Furthermore, the color scheme of orange, grey and red is consistent throughout the website. Both of these design features of the website also follow “Aesthetic and minimalistic design”.

The footer contains social media icons which are recognized by the majority of the population. This follows “Match between system and the real world” as these icons are recognizable to the users. The target audience of this website also will recognize the icons used when selecting a platform in the Boosting page. These icons are universal to the gaming community which therefore follows this usability heuristic.

Lastly, Signup and Login pages’ forms are designed to guide the users on what they should be filling in for each field. The use of universal icons for each field follows “Match between system and the real world” and the placeholder values follows “Error prevention” as it not only tells the user what to input but also restricts the user in entering the incorrect information (i.e. CCV should only be 3 numerical characters).

**Conclusion and future improvements**

Overall, the website was implemented the way I designed it to according to the high fidelity prototypes. The homepage was almost identical to the prototype except for how the testimonials were implemented. An improvement for this would be to implement a backend side where the reviews are stored and more can be added by the user. This can be through a separate page where a user can send in their own review.

One of the functions that I would have wanted to implement was the scroll feature for the navbar; where the navbar changes colour to a lighter more transparent colour when the user scrolls. This was not successfully implemented due to time constraints.

I would have also like to have added more webpages including the Loadouts and Mechanical Guide pages. These pages follow the same layout as the Legends pages and are not so unique. However, the content in them would be updated frequently as the game changes every season which occurs every 3 months. These pages would have up to date information regarding the game. The Contact page was also not yet been successfully implemented.

The boosting page did not have the rank icons successfully implemented as proposed in the prototypes. These would have added another heuristic of “Visibility of System status” which is a good indicator for users what type of rank they are purchasing based on the more recognizable rank icon.

The checkout functionality for this website could be further developed which finalizes the users’ purchase. This will require more backend work which I have not yet learnt how to implement.

I believe the website itself has a lot of potential to be deployed and use for the real world. I have learnt a lot about front end development and realized it takes a lot of work and experience to produce a high quality product- one that is useable in industry.

**References**

1. Nielsen, J. (1994, April 24). 10 Heuristics for User Interface Design. Retrieved from Nielsen Norman Group website: https://www.nngroup.com/articles/ten-usability-heuristics/
2. Flaticon. (2013). Flaticon, the largest database of free vector icons. Retrieved from Flaticon website: <https://www.flaticon.com/>
3. Apex Legends Wiki. (n.d.). [Review of *Ranked*]. Retrieved May 5, 2020, from Apex Legends Wiki website: <https://apexlegends.fandom.com/wiki/Ranked>
4. w3schools. (2019). HTML Color Picker. Retrieved from W3schools.com website: https://www.w3schools.com/colors/colors\_picker.asp
5. Mozilla Foundation. (2019, May 29). MDN Web Docs. Retrieved from MDN Web Docs website: <https://developer.mozilla.org/en-US/>
6. Coyier, C. (2018). A Complete Guide to Flexbox | CSS-Tricks. Retrieved from CSS-Tricks website: <https://css-tricks.com/snippets/css/a-guide-to-flexbox/>
7. Davison, E. (2022, March 26). Best team comps in Apex Legends. Retrieved May 29, 2022, from Dot Esports website: <https://dotesports.com/apex-legends/news/best-team-comps-in-apex-legends#:~:text=Valkyrie%2C%20Gibraltar%2C%20Ash&text=This%20team%20composition%20has%20become>
8. Naiya, D. (2022, April 18). 5 best Apex Legends weapon loadouts for Season 12. Retrieved May 29, 2022, from www.sportskeeda.com website: https://www.sportskeeda.com/esports/5-best-apex-legends-weapon-loadouts-season-12

**Appendix**

*// Source Code from https://drive.google.com/drive/folders/1x8XkvHZfuz-Z-Xuab45gcEE3s0\_ILpqp*

*//all ------------------*

function carousel() {

"use strict";

*if* ($(".testimonials-carousel").length > 0) {

var j2 = new Swiper(".testimonials-carousel .swiper-container", {

preloadImages: false,

slidesPerView: 1,

spaceBetween: 20,

loop: true,

grabCursor: true,

mousewheel: false,

centeredSlides: true,

pagination: {

el: '.tc-pagination',

clickable: true,

dynamicBullets: true,

},

navigation: {

nextEl: '.listing-carousel-button-next',

prevEl: '.listing-carousel-button-prev',

},

breakpoints: {

1024: {

slidesPerView: 3,

},

}

});

}

}

*//initialise*

$(document).ready(function () {

carousel();

});

*Fig.1 Carousel Javascript implementation*

*A picture containing company name

Description automatically generated*

*Fig.2 Three Column footer*

*A picture containing graphical user interface

Description automatically generated*

*Fig.3 Two Column Footer*

**

*Fig.4 Increment and Decrement using jQuery for rank selection (+ -)*