**Part1-1**

**DNS purpose**

The Domain Name System (DNS) converts domain names that are easy for people to remember, like www.example.com, into the IP addresses that computers use to identify each other on a network, such 192.0.2.1. Users will benefit because they no longer need to memorize the IP addresses to access websites and other online resources.

**DNS types**

* **Recursive resolver**: This type that the clients (such as web browsers) typically interact with. The responsibility of the recursive resolver is to convert a domain name to an IP address by sending requests to other DNS servers, if necessary, and returning the result to the client.
* **Root domain**: Its top-level domain of the DNS hierarchy. It is represented by a dot (.) and contains information about the top-level domains (TLDs) such as .com, .org, and .edu.
* **TLD (top level domain)**: TLDs are the highest level of the domain name hierarchy, such as .com, .org, and .edu.
* **Authoritative**: Authoritative DNS server is a server that has the original and most recent information about a domain name. It is the final authority for a domain name, and it provides the IP address associated with that domain name.

Domain names are managed and organized into a hierarchical structure, with each level of the hierarchy separated by a dot (.). TLD (top level domain) like .com, .org and .jo. Second level domain like google, Facebook and yahoo. Third level domain like www.

**Part1-2**

**Communication protocols:** Are protocols that allow the client to communicate with the server by client request and server reply answer. Example of these protocols: HTTP, HTTPS, TCP, UDP, SMTP, POP, and many others.

**Server hardware:** Server hardware refers to the physical components that make up a server, such as the processor, memory, and storage. The hardware cannot do tasks without software.

**Operating system:** Operating systems like Windows, Linux, and macOS provide the underlying infrastructure that allows the server hardware to operate. They manage the resources of the server and give the web server software a platform to run on.

**Web server software:** Such as Apache and IIS, receives and processes requests from clients and sends back the appropriate response. This software is responsible for handling the details of the HTTP protocol.

**The relation between them:**

All these elements must be set up and optimized to function together during the website design and publishing process to deliver the website to users. The operating system must be set up to handle the server's resources in a way that ensures quick and reliable access to the website, and the web server software must be configured to route requests to the appropriate resources on the server.

An example to explain more. When a user requests to access a website, the browser sends a request to the server using the HTTP/HTTPS protocol. The server then uses the web server software to process the request. The operating system then managed the server's hardware and resources to complete the request and deliver the user's requested website. FTP protocol is responsible for publishing files.

**Part1-3**

**The impact of common web development technologies and frameworks on website design, functionality, and management**

Many common web development technologies and frameworks have an impact on website design, functionality, and management. here are some examples of them and the impact:

* **HTML and CSS**: Without these two technologies, web developers are unable to create the structure and layout of a website. While CSS is used to specify how a website should look, HTML is used to specify how a website should be structured. They make it possible for web designers to create websites that are both pleasing to the eye and useful.
* **Integrated Development Environments (IDEs):** Such as Microsoft visual studio and many others. These provide a lot of features to help the developer in writing and editing the code like auto completion and debugging the code.
* **Responsive design**: Responsive design: By making the website responsive, the developer ensures that the website looks and works well on a variety of devices, including desktop computers, tablets, and smartphones. Making a website responsive is important because this process leads to more people accessing the website because of the website’s flexibility on different devices.
* **Front-end development frameworks and libraries:** These frameworks like jQuery provide a structure for handling interactions and creating user interfaces for websites.
* **Back-end development frameworks:** Frameworks like Ruby on Rails and Django give a structure for developing server-side functionality and managing database interactions on the website.
* **Content management systems**: Software platform that enables users to create, edit, collaborate on, publish, and store digital content. CMS are typically used for enterprise content management (ECM) and web content management (WCM).

**Web development technologies and frameworks that I have used to develop my website.**

I used many technologies and frameworks while working on this project. These technologies and frameworks were important to help me to achieve my website goals.

1. More than one language is used in this project. The languages are HTML and CSS.
2. **Web browser**: Such as Chrome, Microsoft edge, and Firefox. The browser's main responsibility is to interpret and make understandable the information it needs to display.
3. **GoDaddy**: This website allowed me to buy a domain name for my website.
4. **PHP**: Language that is used in the back end of a website. It’s responsible for what we call it in series, behind-the-scenes functionality of the site. It refers to the server-side of the application, and it handles things like data storage and manipulation, user authentication, and server-side rendering of the front-end.
5. **Free hosting**: It’s a service which provides free hosting on a web server.
6. **Bootstrap**: Bootstrap is a web development framework that is open-source and free. It offers a collection of syntax for template designs to make the web building process for responsive, mobile-first websites easier.

**Part1-4**

**Search engine optimization**

Text, photos, forms, links, and more can all be found on a web page. The user visits a website to read, watch, listen, or perform tasks like making a purchase or signing up, among other things. Search engine optimization helps the user find what he/she is looking for online. Also, it improves the website so that it appears more prominent when the users search on Google.

The SE has 3 missions to do:

1. Crawling
2. Indexing
3. Delivering search results

Briefly, the process of gaining access to online web sites is known as crawling. Delivering search results entails analyzing a user's search query to determine the best results from the index.

**Indexing**

The index functions as the book's back. A person might search the textbook's index to locate the pages on a certain topic if they wished to focus on that subject. In a similar way, web pages are indexed.

While crawling the web, SE compiles a database of pages it finds, which is then utilized to create the index. Every website that SE crawls has its HTML parsed to remove all the links. SE does this to save the URL that each link points to as its destination.

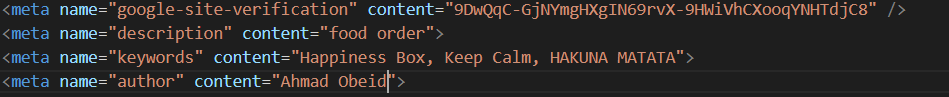
In addition to the other associated metadata that SE has gathered about the page, search engines keep many of these word-URL connections for each URL that is indexed. This procedure is used to choose which URLs appear in the search results.

**The techniques to improve the ranking and indexing of the website.**

* Full of unique, high-quality content on the website.
* Update the content regularly.
* Pick keywords to target.
* Construct backlinks to the website.
* Submit a sitemap to each search engine.
* Get rid of duplicate content.
* Developing on-site SEO.

The steps I followed to develop the SEO for my website:

* I opened google search console.
* I put the URL of my website and then got a meta tag.
* Then I copied it and put it in home page code with keyword



* Then I verified my website URL
* Then I went to xml-site map to get my website sitemap
* I put it in google search console and then in the direct manager in public HTML.

**Part1-5**

**Front-end**

The front-end of a website refers to the region that visitors can interact with. It oversees the website's layout, functionality, and interactivity. The front-end technologies are HTML, CSS, and JavaScript.

**HTML** (Hyper Text Markup Language) is a markup language used to design the web. It builds the structure, layout of the web, and the meaning of the various elements on the page.

**CSS** (Cascading Style Sheets) is a stylesheet language used to control the appearance of web content, including font, color, image, and layout.

**JavaScript** is a programming language that is used to make web pages interactive. The developer through JavaScript can add behavior to the elements and make them move or change when a user interacts with them.

**Back-end**

The back end of a website is responsible for what we call it in series, behind-the-scenes functionality of the site. It refers to the server-side of the application, and it handles things like data storage, user authentication, and server-side rendering of the front-end. **PHP**, **Python**, and **Java**, as well as databases like **MySQL** are considered as back-end technologies.

**How the front and back-end relate to presentation and application layers.**

A website's presentation and application layers are formed by its front end and back end. Users communicate with a website's presentation layer, which also renders the user interface. The application layer controls the logic and operations of the website and collaborates with the presentation layer to display the required information to the user.

Let’s say that you're using the restaurant's website to make a food order. You enter your credit card information on the front-end of the website (payment page), while in the back end, the application takes that information to send you an email order confirmation after storing it in a database (which is built on the server). To conclude, the presentation layer is front-end and application layer is back-end.

**The differences between online website creation tools and custom-built sites with regards to design flexibility, performance, functionality, User Experience (UX) and User Interface (UI)**

**Custom-built sites**

Custom-built websites are those that are written entirely from scratch and by hand in coding languages like HTML, CSS, and JavaScript. Because the website developer is free to make any adjustments, he/she desire, these websites offer more flexibility in terms of design and functionality. But compared to using a website building tool, build a website from scratch takes more time and effort and needs a higher level of knowledge and skill.

**Website creation tools**

Websites creation tools are software platforms that enable developers to create and publish websites without having to write any code. These tools frequently offer a range of pre-designed themes and functionalities and have an intuitive user interface. Some well-known website creation tools: Weebly, Wix, and Squarespace.

**The differences between online website creation tools and custom-built sites with regards to:**

1. **Design flexibility**: Since custom-build sites allow the developer to customize the website in any way, he/she wants, the developer will have more options and controls than they would in website builder tools. Moreover, in website creation tools, the design capabilities are usually less and there may not be as much customization on the website.
2. **Performance** Custom-build website beat website creation tools in terms of performance. Custom websites can be optimized to function according to the demands of the website because they are created from scratch. In addition, because website creation tools are made to be simple to use rather than to increase performance, tools ability to improve website performance is poor.
3. **Functionality**: Both custom websites and website creation tools can offer a wide variety of abilities and features. However, one benefit of specifically designed websites may be the ease with which they may be integrated with other systems.
4. **UX & UI**: User interface (UI) and user experience (UX) are significant aspects to consider while developing a website. The quality of these components will rely on the tool or platform being used as well as the knowledge, experience, and abilities of the developer or designer. Both custom-built sites and website creation tools can offer good UX and UI.

**Part1-7**

The tools, software, management services, and technology selected to build a custom website may differ. The specifications for one Web site will be different from the specifications for another Web site. As there are several factors that could influence the selection process, these include:

1. The type of website being built (e.g., restaurant website, e-commerce website, web application, etc....)
2. The skills and expertise of the development team
3. The target audience for the website

**Technologies, management services, tools, and software’s justification**

1. **HTML & CSS:** These two programming languages are regarded as essential elements in the creation of every website. HTML must be used by any developer who wishes to organize the content and design of a website. Additionally, CSS must be used by the developer to control and alter the layout properties. Websites that combine HTML and CSS are aesthetically pleasing, well-organized, and responsive. Additionally, as HTML and CSS are supported by all web browsers, websites created using these languages will be usable by a variety of users.
2. **JavaScript:** Without employing JavaScript, the developer can produce structured, responsive, and visually appealing webpages. However, the developer must employ JavaScript if they want a dynamic user interface, interactive elements, improved accessibility, and the use of JavaScript libraries and frameworks to better codebase maintainability.
3. **PHP:** there is no actual website without a back-end (server-side languages such as PHP). Because every website’s project must have requirements, functions, and logic, and these cannot be implemented without PHP for back-end development.
4. **XAMPP:** if the developer wants back-end development, then he/she needs XAMPP for the database to create in result a functional website which meets project requirements.
5. **VS code:** The code must be written in an editor by the developer. A well-known editor that is accessible on Windows, Linux, macOS, and other operating systems is Microsoft's Visual Studio Code (VS code). It contains a number of capabilities, such as code completion and debugging assistance to find the code problems, that make the process of developing websites easier.
6. **Font awesome**: font Awesome is a popular icon font library that can be used to add scalable vector icons to a website.

**Part1-8**

There are many tools and techniques that are available for designing and developing a custom-built website. Here are some examples of many:

* **Integrated Development Environments (IDEs):** Such as Microsoft visual studio and many others. These provide a lot of features to help the developer in writing and editing the code like auto completion and debugging the code.
* **Design tools such as Adobe XD and Figma:** these tools help the developer in designing wireframes or prototypes for a website. Personally, I used Figma to design wireframes for my website.
* **Front-end development frameworks and libraries:** these frameworks like jQuery provide a structure for handling interactions and creating user interfaces for websites.
* **Back-end development frameworks:** frameworks like Ruby on Rails and Django give a structure for developing server-side functionality and managing database interactions on the website.
* **Deployment & Maintenance tools:** like Cloudways which helps the developer to ensure that the website functions correctly.
* **Web hosting services:** the infrastructure for hosting and running the website on the internet is provided by these services like AWS and Google cloud.

**Website Vs design document + Evaluation for the development process**

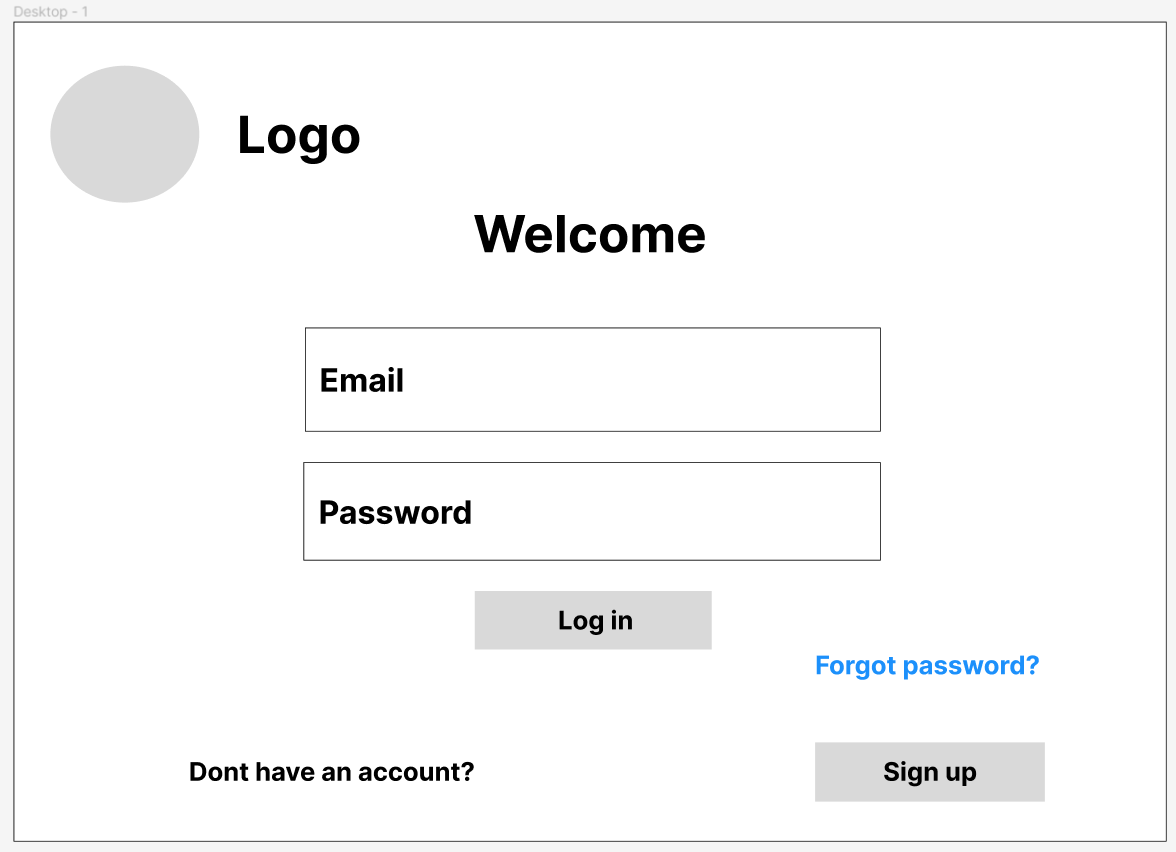
**Making decisions in website pages’ implementation**

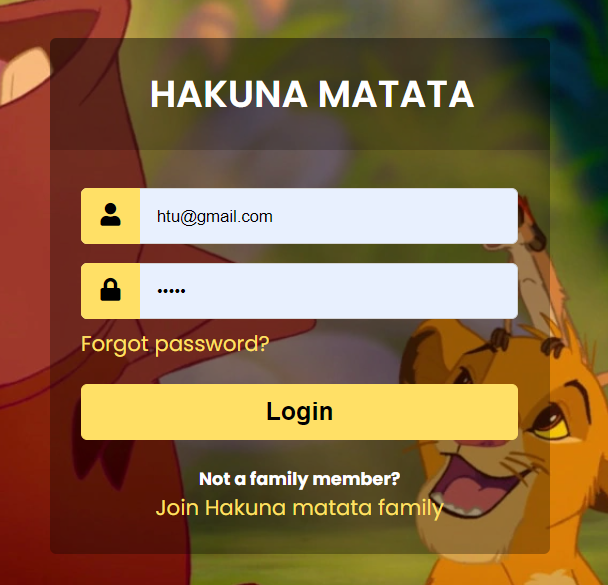
After I wrote the design document for my website, I started to build the website itself. Any website needs a front-end for pages layout, and back-end for the website requirements and pages functions. So, in this section we will track each part in the design document and compare it with website front and back-end implementation decisions and see the similarities and differences. Moreover, I will give the reasons for the exit differences during the comparing and evaluating process. In addition, I will mention the challenges that I have faced.

**Website pages and functions against design document**

I wrote a goal, scope, and functional requirement for my website in the design document, and I drew a sitemap. The website functions and pages are similar and accomplished what I wrote in the design document for this field only. But according to the wireframes, not all the pages on the website are similar to their wireframes. There are few differences between them, and I will track each difference page by page.

**Signup and login page**

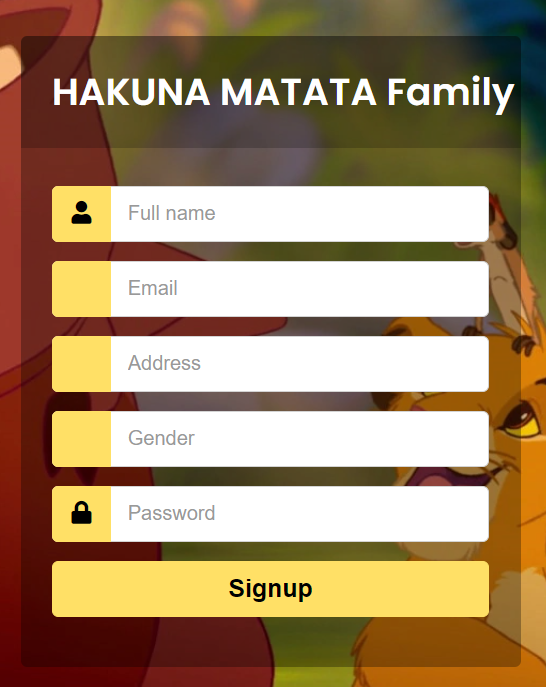




We notice that in the wireframe there is a logo, but on the login page there is not. Instead of the logo, I put a background for hakuna matata movie. I did the change in the development process because I felt that if the customer goes to the login page, he/she will feel like it really is a Hakuna matata restaurant and the memory of the movie might come back to him/her. So, in the result I will attract and catch the eye of the customer to my website.

Graphical user interface, application

Description automatically generated



The difference is the same as the login page, but in signup wireframe there is a back button to let the customer go back to the login page, and there is no text field for the gender. But, on the page there is no back button, and the customer can see the field where he/she can fill the gender. I removed the back button, so the customer can only get in the login page through signing up.

**Home & contact page**

Graphical user interface

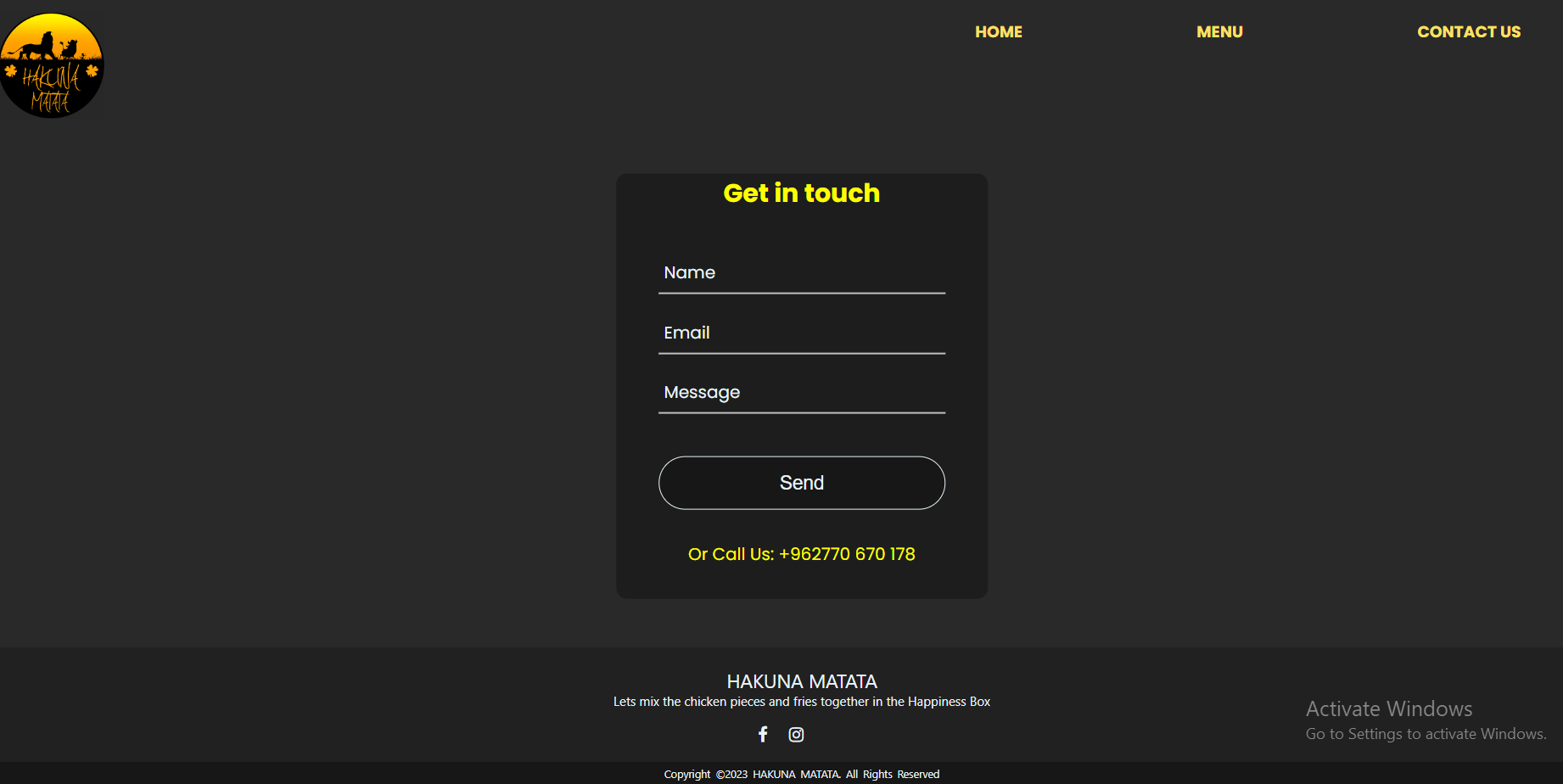
Description automatically generated with medium confidence



There are a lot of differences between the home page and its wireframe. Frist, I removed the about page and Instagram and Facebook’s buttons when I implemented the home page. I designed a responsive footer and I put the previous elements in it. I removed the image beside the title. Second, I removed the order now’s button because there is no need for it, the customer can order through menu. Finally, I add a login button to let the customer login before he/she can order. And when the customer logs in, the button will disappear and there will be logout button instead of it on the home page. The changes to the home page were for better user experience and better UI. There was no need for the about page, and the customer can read about the restaurant and get its social media through the footer. As for the login and logout buttons, I added them to achieve the website functional requirements and its sitemap.

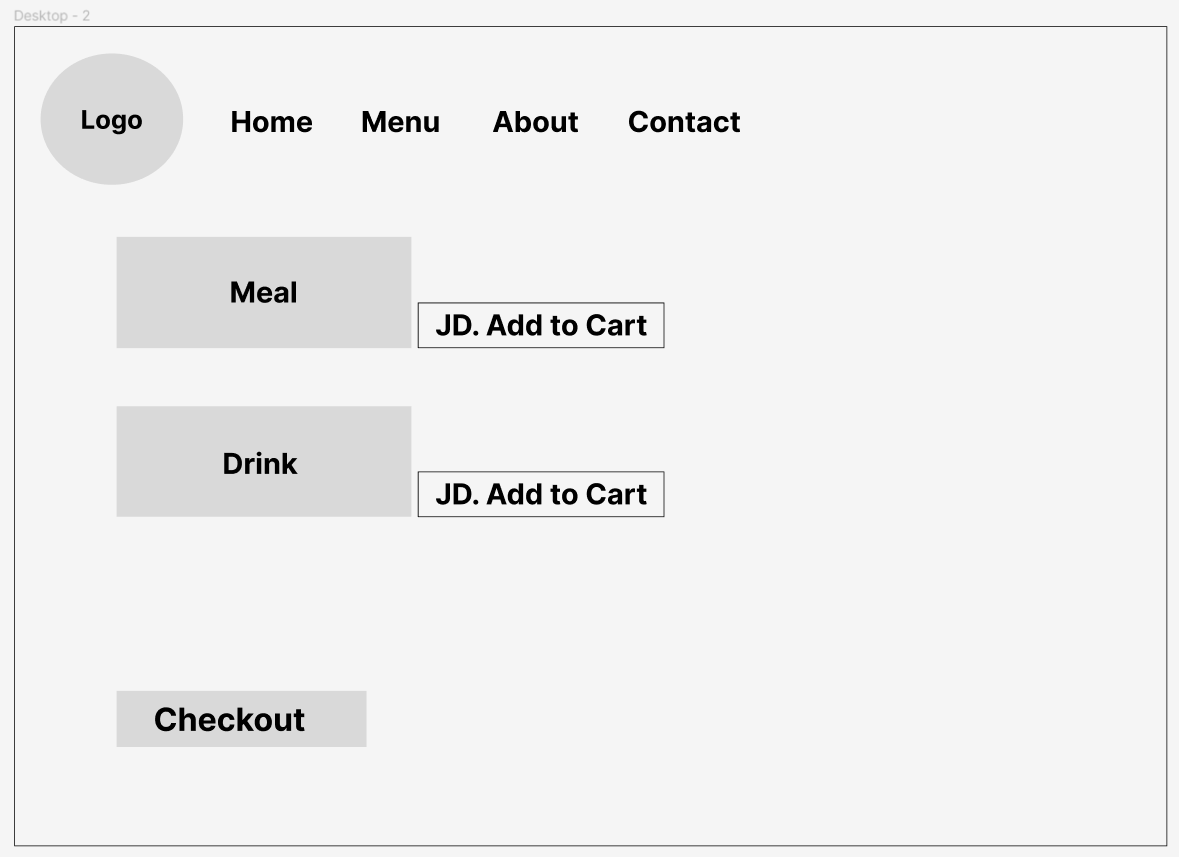
Graphical user interface, application, PowerPoint

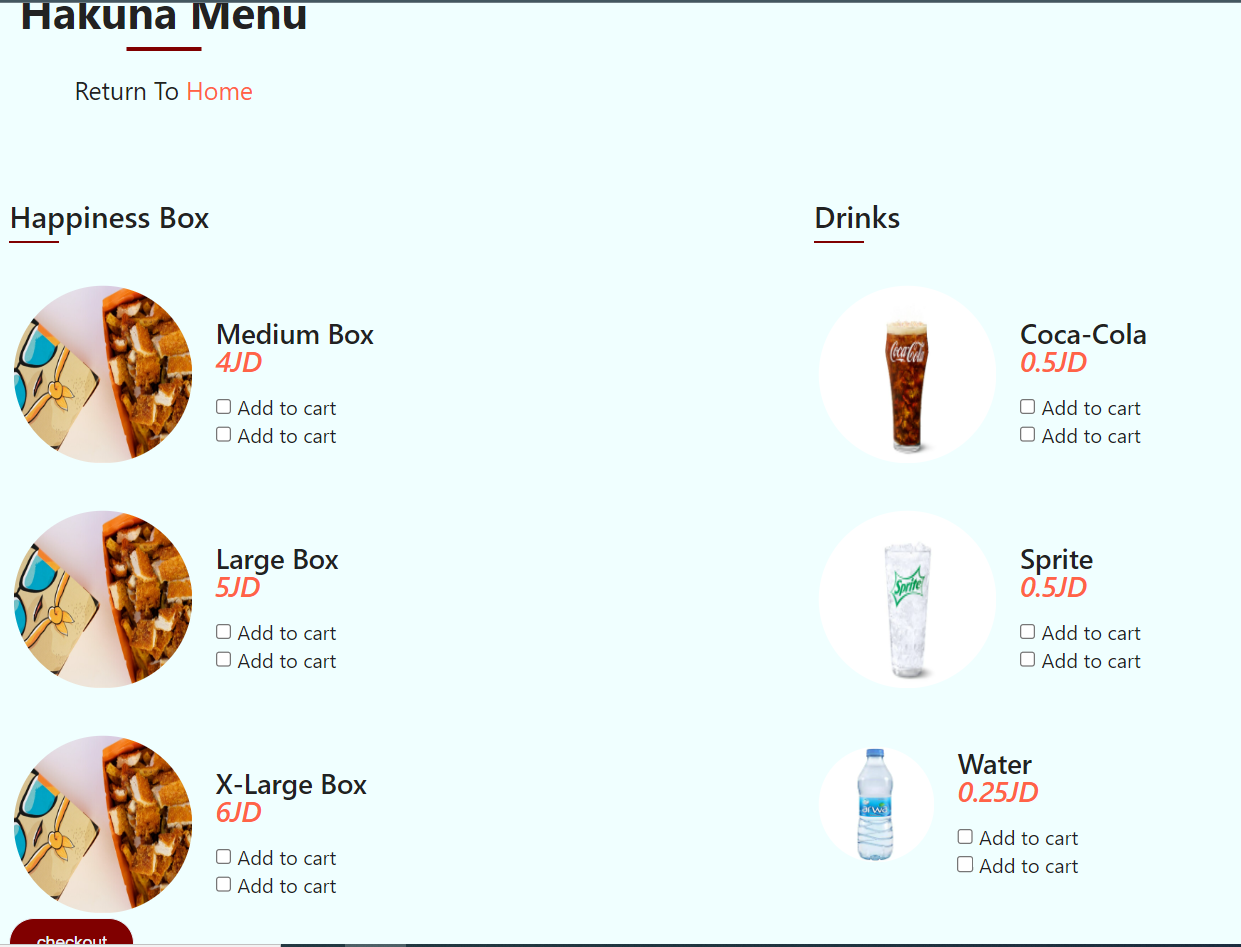
Description automatically generated



The differences between the contact page and its wireframe are similar to the home page. Also, when I implemented the contact page, I decided to remove the restaurant’s map and the image that contains the restaurant ‘s contact information. Instead of that, I added a form to contact that enables the customer to contact us in better user experience. In addition, I put our number.

**Menu page**



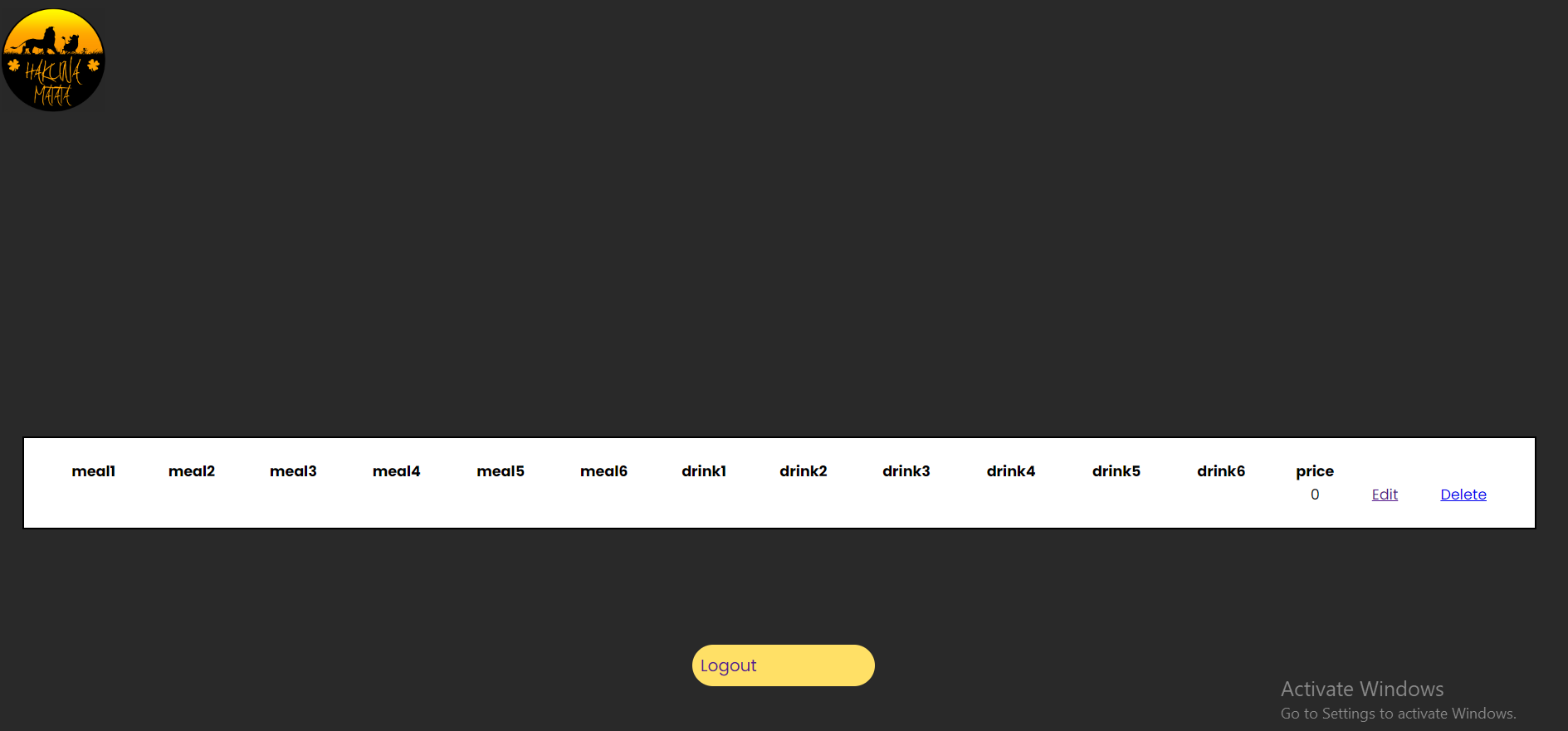


When I implemented the menu page, I removed the navigation bar and the logo and put instead of them (**Hakuna Menu**) and a link to return to the home. This change in my point of view will allow the customer to only fucose on the menu items and for clear and better user experience and UI

**Cart & edit page**

Graphical user interface, text

Description automatically generated



The Cart page and its wireframe are very similar.

A picture containing table

Description automatically generated

Graphical user interface, application, Teams

Description automatically generated

Also, the edit page and its wireframe are similar to each other.

I compared and evaluated the process of writing the design document against the website pages and functions implementation decisions. Also, there was regular testing to ensure that all the functions and features in the website work and meet the website goal and scope, and the results were positive. However, there is no development process that is zero-challenges or zero-future recommendations. There are a few things that I might add to get a better user experience. First, I might change the background to black one and add the website logo in the signup page, login page, and menu page to get all the website pages connected with the same theme. It’s not a large difference in the theme, but this recommendation might improve the UI and UX. Second, I could've added a button in the cart page to return the customer to the menu page for better UX. Third, make the contact page function in good way, but I did not do a back-end code for it, I faced a problem with it because it’s hard to do it actually which is enabling the customer to actually send a message to my email. Another two problems that I have faced are in the menu page, cart page, and edit page. In the menu page, I have limited the number of items that the customer can add, but he/she can still add with quantities from single item in the menu but not a lot of items. The cart and edit page before the back-end development were responsive, but after I implemented PHP for the website, they no longer become perfectly responsive to mobiles frame. I really worked hard and did my best to fix and deal with these challenges, but apparently, I did not achieve well solving for the challenges.

**References**

1. Lutkevich, B. and Burke, J. (2021) What is DNS? how domain name system works, Networking. TechTarget. Available at: https://www.techtarget.com/searchnetworking/definition/domain-name-system (Accessed: October 28, 2022).
2. A very complex question, with a complex answer. I hope it can clarify a few things...http://www.quora.com/what-is-the-purpose-and-relati... (no date) Quora. Available at: https://technologycareers.quora.com/What-is-the-purpose-and-relationships-between-communication-protocols-server-hardware-operating-systems-and-web-server (Accessed: October 28, 2022).
3. Search engines and how they affect your website (no date) Mailchimp. Available at: https://mailchimp.com/resources/how-search-engines-work/ (Accessed: December 20, 2022).
4. Michigan Technological University (2021) Five ways to improve your site's ranking (SEO), Michigan Technological University. Michigan Technological University. Available at: https://www.mtu.edu/umc/services/websites/seo/ (Accessed: December 20, 2022).
5. Jones, K. (2022) 11 Seo Tips & Tricks to improve search indexation, Search Engine Journal. Available at: https://www.searchenginejournal.com/11-seo-tips-tricks-to-improve-indexation/288521/#close (Accessed: December 20, 2022).
6. 21+ best web design tools that designers are using in 2022 (2022) The Official Cloudways Blog. Available at: https://www.cloudways.com/blog/best-web-design-tools/ (Accessed: January 14, 2023).
7. Javinpaul (2022) 10 best frontend and backend frameworks to learn in 2022, Medium. Javarevisited. Available at: https://medium.com/javarevisited/10-best-frontend-and-backend-frameworks-for-java-python-ruby-and-javascript-developers-cce3c951787a (Accessed: January 14, 2023).
8. Amsler, S. and Churchville, F. (2021) What is a content management system (CMS)? definition from whatis.com, Content Management. TechTarget. Available at: https://www.techtarget.com/searchcontentmanagement/definition/content-management-system-CMS (Accessed: January 14, 2023).
9. Front vs. back-end web development: Your ultimate guide (No date) Front vs. Back End Web Development: Your Ultimate Guide: Programming and Development Blog | Stone River eLearning. Available at: https://blog.stoneriverelearning.com/front-vs-back-end-web-development-your-ultimate-guide/#:~:text=What%20Does%20'Front%20End'%20and,the%20server%2C%20application%20and%20database. (Accessed: January 21, 2023).
10. M, B. (2022) Advantages of custom-built sites vs online website Builder Tools, Icecube Digital. Available at: https://www.icecubedigital.com/blog/advantages-of-custom-built-sites-vs-online-website-builder-tools/ (Accessed: December 5, 2022).
11. The difference between template websites vs custom made websites (2022) Manifera. Available at: https://www.manifera.com/difference-template-vs-custom-made-website/ (Accessed: December 5, 2022).