Thobe Nation
Traditional Islamic Apparel Ecommerce Webstore

Final Project Report

TU 857
BSc in Computer Science Infrastructure

Ridoy Asaduzzaman

C18308501

Supervisor: Mariana Rocha

School of Computer Science

Technological University, Dublin

Date: 22/02/2023

Abstract

A small business's online appearance may significantly affect its performance regardless of the industry. Some businesses continue to ignore the fact that the majority of their clients browse their websites before making purchases, even in this day and age. A good online presence, especially a website, may be essential for boosting sales. Sometimes businesses are afraid to get online because they believe they lack the technical knowledge and website management skills. Additionally, businesses are worried about cost.

These statements are especially true for Traditional Islamic Businesses in Ireland. That is why it is imperative to create and raise the online standards of these businesses. This project will work with a small business owner and help them integrate their Islamic clothing business with technology. This project will feature a website which will give the owner the opportunity to introduce the brand to potential customers.

Declaration

I hereby declare that the work described in this dissertation is, except where otherwise stated, entirely my own work and has not been submitted as an exercise for a degree at this or any other university.

Signed:

_Ridoy_____

Ridoy Asaduzzaman

Date: 22/02/2023

Acknowledgements

I want to start by expressing my gratitude to my supervisor Mariana Rocha, whose time, assistance, support, and advice were crucial to the project's success. Without her direction, encouragement, and motivation, this project would not have been possible.

I also like to thank Jonathan McCarthy for his ongoing support, wisdom, and academic knowledge when he taught us the FYP module. I would also like to thank Patricia O'Byrne for her expertise as a second reader.

Furthermore, I would like to thank the teaching team at the DIT School of Computing. Your expertise, aptitude for instruction, and enthusiasm are unmatched and will always be valued. Last but not least, I want to thank my wife, Shaila Alam, for her continuous support during the academic year and my friends for always offering me their eager ears and eyes.

Contents

Table of figures

1. Introduction

This report outlines several development processes of a business-oriented e-commerce project. It offers a web application with a catalogue that will feature traditional garments that customers can view and interact with. Furthermore, a basket and checkout procedure involving electronic payment systems like credit cards, debit cards, and online merchants like PayPal. The goal of this website is for expat Muslims in Ireland to have access to traditional Islamic and cultural clothing and accessories from the comfort of their home.

1.1. Project Background

After conducting research using a survey (figure 19), which shows a significant lack of websites featuring Islamic and traditional Muslim clothing and accessories in Ireland. Therefore, there is a significant demand with low supply. Furthermore, research reveals that "80% of Irish consumers use the Internet to look for information about a business" and "73% are particularly frustrated by businesses that do not have a website at a minimum" (Murray, 2019). These facts are relatable as it is difficult and expensive to find traditional Islamic and cultural clothing online in Ireland. Accordingly, it is imperative to create this website and locate a supplier ready to collaborate to provide these items to Muslims in Ireland.

Moreover, after reviewing the current Islamic web stores, it is evident they lack individuality. These sites are built using Shopify and WordPress, which use prebuilt platforms, so most of the websites end up having the same layout and design, which doesn't allow them to be unique and stand out. This website will aim at solving a couple of different problems. Firstly, this website will give people access to difficult-to-source clothing and accessories which aren't usually available in Ireland. Furthermore, this website can help the supplier go from selling on WhatsApp and marketing with word of mouth to increase overall flexibility. It will also help the local supplier increase brand awareness and build credibility over customer reviews.

Moreover, the online website will have reduced prices and provide accessibility to products. Therefore, customers won't have to go into a physical store where prices will be much higher due to rent, utility bills, property taxes, etc. Traditional Islamic clothes and accessories provide us with a continual reminder of a simple, sustainable lifestyle. Another benefit is the ability to comprehend the value of conventional items and the history of the people who manufactured them. There are goods unique to each culture. These items are crucial for keeping culture and religion alive since they teach people about their heritage and Identity. 9 This website will be used to create a sense of belonging in the Muslim community.

1.2. Project Description

This is a modern e-commerce project that will focus on helping a small business that has no online platform. One of the primary advantages of a business having a website is that it gains more credibility. There are certainly a lot more suppliers with services similar to theirs available. Therefore, one of the greatest ways to stand out is to have a website that looks attractive and successfully communicates valuable information to the clients.

Without a website, customers could question the credibility of the company. They have the opportunity to make a fantastic first impression and convince potential clients that they are a respectable business by having a website.

This project will feature a website that will allow the user to interact with it by allowing them to log in and search for products by adding or removing products from their cart in a crud style manner. Furthermore, the user will then be able to enter their address in a form the admin will use to deliver the product the user selected. After that, the user can make a payment for the products. The admin will be able to view all the user information and have a dashboard to add/update and remove products.

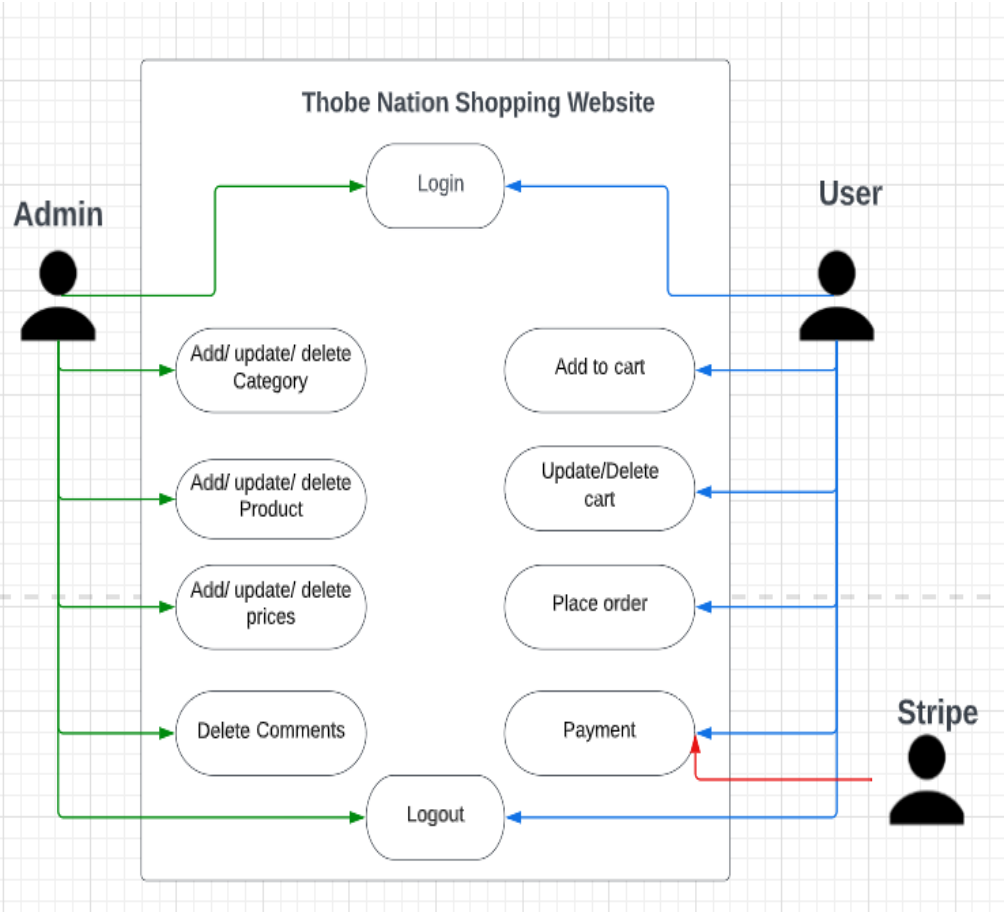


Figure 1 The use case diagram above shows in full the functionality interactions that both the admin and the consumer may have with the software.

1.3. Project Aims and Objectives

- The main aim of this project is to develop a cross-platform web application to aid prospective Irish Muslims in purchasing traditional Islamic clothes.
- Getting in touch with suitable suppliers of Islamic clothes to collaborate with.
- As there is a need for Front-End of experience; therefore, some research and experimenting will be necessary.
- To learn how to build the font-end, back-end and how to use frameworks.
- A good understanding of MongoDB is required as it will be utilized as the database foundation.
- Getting in touch with students to model the clothes for the website.
- Making the website cross-platform to it can be used on phones.
- Understanding how to deploy the website and what platform to use.

1.4. Project Scope

The scope of this project is to allow users to interact with it by enabling them to log in and search for items by adding or removing products in a crud-style fashion from their basket. It is designed for a single admin user and will give them the opportunity to input their address in a form that the admin will utilize to deliver the goods they choose. The user may then purchase the goods by making a payment. In addition to having a dashboard where they can add/update/remove goods, the admin will have access to all user information.

This project scope assumes that the user knows how to use a computer and phone to access the website and make online purchases. The project only caters to customers who have a basic understanding of computers and online payment systems. Furthermore, the project is designed to cater to a single seller.

1.5. Thesis Roadmap

Chapter 1, Introduction:
This chapter provides a summary of the project, as well as background information and problems encountered throughout the project.

Chapter 2, Literature Review:

This chapter will cover the research required to complete this report. First, a description of similar websites currently on the market will be provided. Nielsen's Heuristics will then be used to assess these systems. Thereafter, different technologies relating to front-end, back-end and Database will be

researched based on advantages and disadvantages. Additionally past projects which have used similar technologies will be researched.

Chapter 3, Experiment Design:

This chapter discusses the design method used for this project. The comparison of various software approaches will be the initial step. Following an evaluation of their merits and cons, the one most suited to this project will be chosen.

Chapter 4, Experiment Development:

This chapter will provide an explanation of how the primary system components were created. The configuration of the database, server, and clients will be detailed briefly first. Individual characteristics will then be described.

Chapter 5, Testing and Evaluation

To make sure that the website is optimized for functionality and usability, this chapter will look at black box testing, usability testing, and mobile device testing.

Chapter 6, Conclusions Future Work:

This chapter will provide a summary of how the entire project was produced, information regarding work that has to be done in the future, and a personal reflection on the experience.

2. Literature Review

2.1. Introduction

The research that is necessary to finish this report will be covered in this chapter. First, a summary of comparable websites currently available on the market will be given. Then, these systems will be evaluated using Nielsen's Heuristics. The benefits and drawbacks of various front-end, back-end, and database technologies will be examined. Additionally, similar technology-using projects from the past will be investigated.

2.2. Alternative Existing Solutions

2.2.1 Nielsen's Heuristics

Nielsen's Heuristics assessment is a problem-solving technique used to examine usability concerns in user interfaces. This method is a quick and straightforward way to evaluate usability in interfaces. There are ten guidelines for assessing the usability of website interfaces. The quote, *"Even the best designers, produce successful products only if their designs solve the right problems. A wonderful interface to the wrong features will fail." epitomises this assessment by Jakob Nielsen. The points awarded for the guidelines are based on my judgement.*

1. **Visibility of system status:**
   The system should always keep users aware of what is going on by providing suitable feedback in a timely manner.

2. **Match between system and real world:**
   Instead of system-oriented jargon, the system should speak the users' language, using words, ideas, and concepts that are recognisable to the user.

3. **User control and freedom:**
   Except for regulations that go against the system or conflict with some feature, the interactions must provide users the flexibility to choose the actions they deem fit.

4. **Consistency and standards:**
   To reduce user confusion, utilize the same terminology across the system. Users should have no uncertainties regarding the meaning of words, images, or symbols while interacting with a product.

5. **Error prevention:**
   A smart design should always keep issues and errors at bay.

6. **Recognition rather than recall:**
   The user should not have to recall all the system's operations or functionalities. As a result, always leave minor reminders of information that might help consumers navigate the designs.

7. **Flexibility and efficiency of use:**
   Both beginner and expert users should benefit from the designs.

8. **Aesthetic and minimalist design:**
   Interactions should only convey necessary information. Graphic components that are unwanted and can overload and distract consumers.

9. **Help users recognize, diagnose, and recover from errors:**
   Designs should assist users in identifying and resolving potential issues and faults.

10. **Help and documentation:**
    Even though it is preferable if the system can be utilized without documentation, assistance and documentation may be required.

Aljawdah Attire

https://aljawdahattire.com (Islamic Men's clothing website)
"A small, traditional attire brand striving to bring you culture with quality"

This is a Shopify-powered website focusing on two main pages: "Home" and "Catalogue". It is a simple website where customers can browse through a small range of traditional Islamic clothes, select the size and colour, and then add the product to their basket. After this, the customer can check out using the shop-pay system.

Details and Features:

· Allows the user to browse through a collection of clothes.

· Allows the user to select size and colour of clothes.

· Allows the user to add to basket and pay through multiple payment methods.

· Allows for a contact page where the customer can contact the admin.

Nielsen's heuristic analysis

Exam	Points Available	Points Awarded	Comment
Visibility of system status	2	2	Customer is notified of purchase.
Match between system and real world	2	2	Customer can browse through without ad pop ups
User control and freedom	2	2	Customer Can add or remove items without interference
Consistency and standards	3	3	Website follows convention on select-cart-pay which is easy to understand
Error prevention	3	1	Error messages very vague and not targeting problem
Recognition rather than recall	3	2	Some pages don't have type and style of clothing on it.
Flexibility and efficiency of use	2	2	Easy to navigate for first time user.
Aesthetic and minimalist design	3	3	Design is quite minimalist and accurate.
Help users recognise, diagnose, and recover from errors	2	1	Error messages can be quite vague and not accurate to a specific problem.
Help and documentation	3	2	There is a contact form but no live support.
Total	25	20	

Libaas Boutique

https://libaasboutique.com/ (Traditional Women's Clothing website)
"Don't miss out on thousands of super cool products & promotions"

This is the most famous traditional website in Ireland, a WordPress-powered webpage which features an extensive range of clothing items and jewellery. The website's key feature is that it allows visitors to browse through a range of items before selecting the size and colour after which they can add the items to the basket and then check out with either cash upon delivery or PayPal.

Details and Features:

· Allows the user to search through a range of clothes.

· Allows users to go through offers, recommendations, and best sellers.

· Allows the user to have a Wishlist for personalised products.

· Allows the user to chat with a support bot.

Nielsen's heuristic analysis

Exam	Points Available	Points Awarded	Comment
Visibility of system status	2	1	Customer is notified of purchase. Some links on the page have no content.
Match between system and real world	2	1	Customer can browse but every page change has an unneeded marketing pop up
User control and freedom	2	2	Customer Can add or remove items without interference
Consistency and standards	3	3	Website follows convention on select-cart-pay which is easy to understand
Error prevention	3	1	Random parts of the site lead to anonymise sites.
Recognition rather than recall	3	3	Shows name types and styles of clothes on page.
Flexibility and efficiency of use	2	1	Vast number of options can confuse a first-time user

Aesthetic and minimalist design	3	2	Design is cluttered, and some pages have no content
Help users recognise, diagnose, and recover from errors	2	2	Errors are easy to recognise by customers
Help and documentation	3	3	There is live support available as well as 9am-6pm phone support
Total	25	19	

Based on Nielsen's Heuristics assessment results. Libaas Boutique is more user-friendly however, it is lacking in terms of user security as the site does random external website links. Aljawda Attire, on the other hand, comes in a close second in terms of design but feels a lot more secure. Both will be utilised to take some inspiration for this project.

Comparison of other Irish Islamic sites:

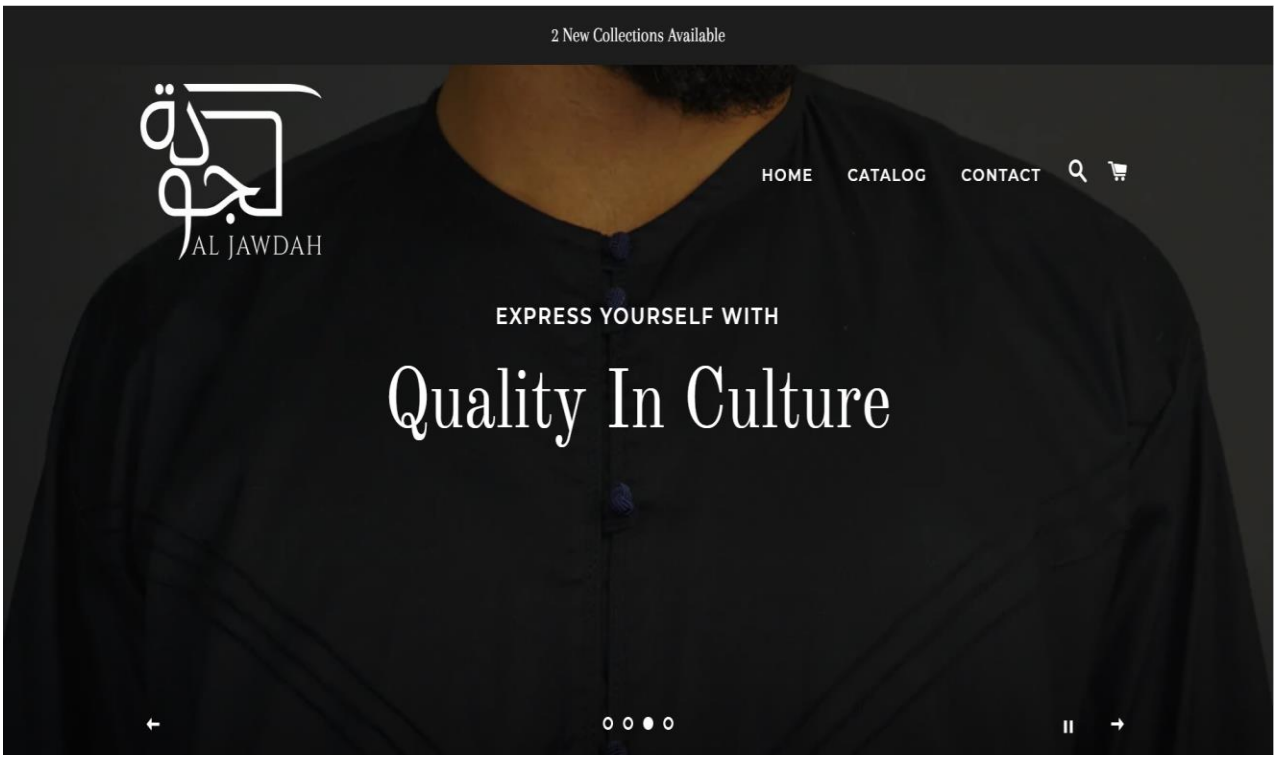


Figure 2 - Home page
Shopify (n.d.)

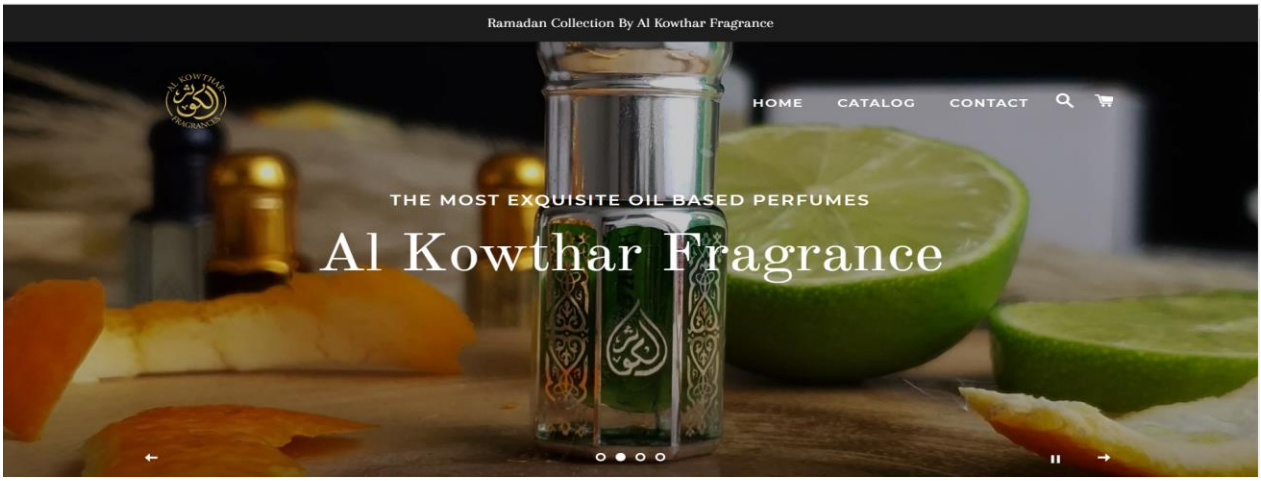


Figure 3 - Home page
Annaqah (n.d.)

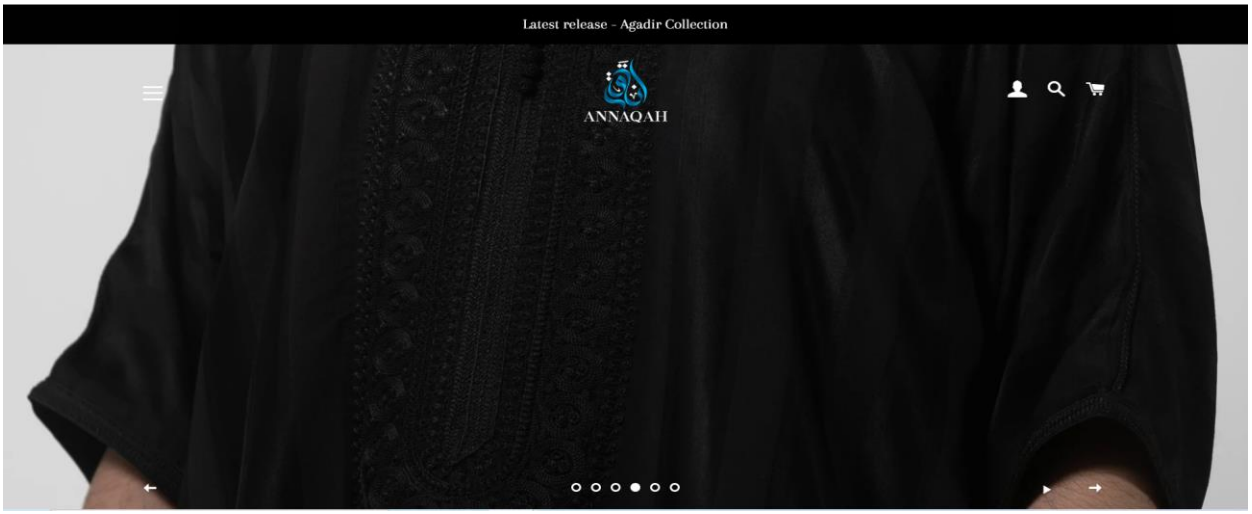


Figure 4 Alkowtharfragrance (n.d)

Look at the sites, and you'll see they bear a strong resemblance. Each website offers design templates, which are often mistaken for designs. Templates provide a cookie-cutter approach to web design, and don't allow for creativity. This is a problem. Templates are often chosen as a way to save money and quickly get a site up and running. That's OK, but the problem is that many sites begin to resemble others on the web.

To make the customizable elements that one wants to include on a website, the website for this project will be made using code. As a result, total control over the site's appearance and functionality exists. Making a custom website enables one to produce exactly what is desired. No restrictions will exist on functionalities, scalability, or application integrations.

2.3. Technologies Researched

This section will evaluate a few different technologies in front-end, back-end and database selection. Thereafter, weigh their benefits and drawbacks, and choose the most appropriate for this project.

2.3.1 Front - End

- **JavaScript**

  Advantages:
  As a newbie, JS is the easiest language to learn. It does not require any installation or configuration. This is a very powerful language. With JS, practically anything is possible. JavaScript has a wide range of uses, including creating complex web pages, mobile apps, and even robots. Developers prefer using JavaScript in conjunction with HTML and CSS to create dynamic web sites and is very fast on the client-side browser. Furthermore, JavaScript makes it simple to alter or add variables at runtime or create code that generates code. The creation of scripts or templating engines benefits greatly from this freedom.

  Disadvantages:
  JavaScript's aggressive type coercion is a serious issue. This occurs because of the fact that some data that is input into the code may have distinct meanings in this language. For instance, the symbol "+" can indicate adding two numbers, adding a number to text, or adding a text to a number. Moreover, the debugging feature, which is essential for any developer, is absent from JavaScript. This makes detecting mistakes extremely difficult, especially because the browser does not display any errors.

JS sidebar.js ×

```
1   import { getPlaces } from './dataService.js';
2
3   function renderCities() {
4     // Get the element for rendering the city list...
5     const cityListElement = document.getElementById('citiesList');
6
7     // ...clear it...
8     cityListElement.innerHTML = '';
9
10    // ...and populate it, one place at a time using forEach function
11    getPlaces().forEach((place) => {
12      const cityElement = document.createElement('div');
13      cityElement.innerText = place.name;
14      cityListElement.appendChild(cityElement);
15    });
16  }
```

Figure 5 JavaScript function to render cities. Team, D. (2019).

- **Dart**

Advantages:
There is a lot of documentation available for this language because Google is working on the Dart interpreter, all of the language's characteristics are described in depth. This language has a very high level of performance, Dart-written programs typically run quicker than JavaScript-written programs. Moreover, High-quality real-time apps may be made using Dart, a dependable programming language. This programming language is object-oriented and includes inheritance, interfaces, and optional typing.

Disadvantages:
Dart is a fairly new programming language so it lacks strong development communities. In the market, it is seldom utilized. Thus, if you search for a career in this field, it will be more difficult to locate an employer hiring a Dart programmer. Furthermore, as the language is still having development updates regularly there is a danger that the API may change or that things won't be properly or fully documented if you begin writing in Dart right away.

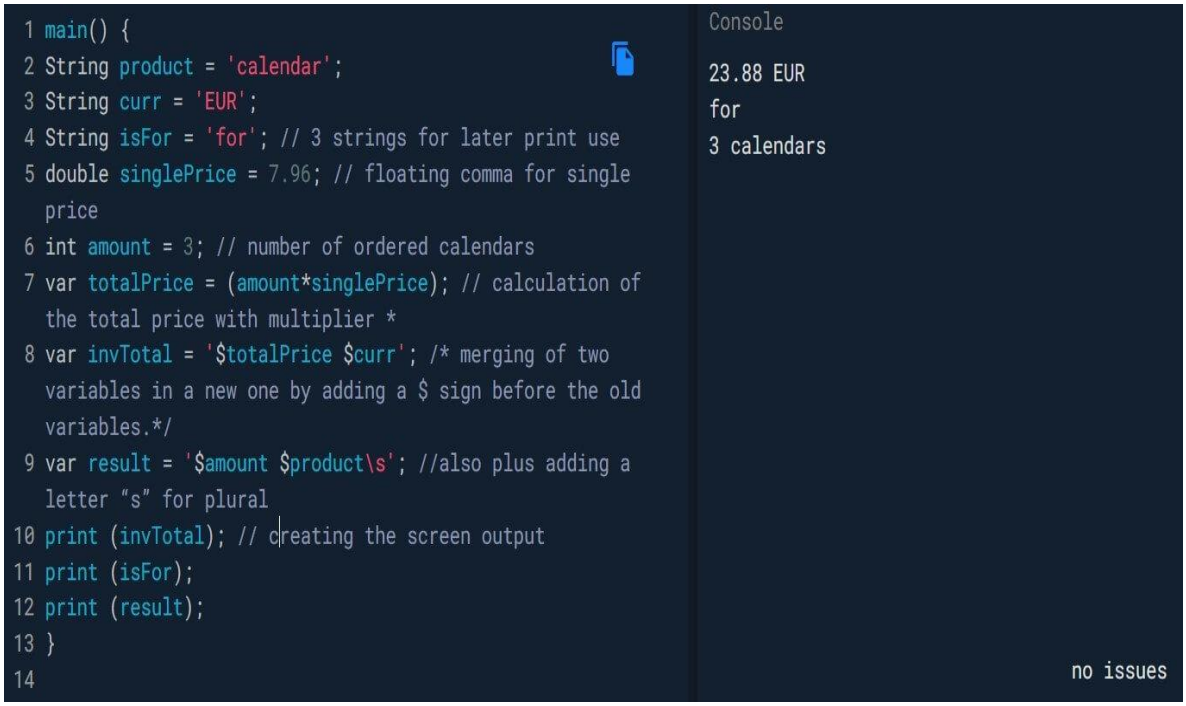


Figure 6 Multiplication calculation using Dart - Nuraly, A. (2020).

- **TypeScript**

Advantages:
Major advantages of JavaScript are carried over into TypeScript, but it also offers further advantages due to static typing and other TS-specific ideas. They are especially helpful for dispersed teams working on the same project with big codebases. Strong static typing is now available in TypeScript. Once declared, a variable can only accept specified values and cannot change its type. Developers are informed of type-related errors by the compiler, preventing them from entering the production stage. As a result, the execution of the code runs more smoothly and with less likelihood of mistake. Classes, interfaces, inheritance, and other class-based object-oriented programming (OOP) principles are supported by TypeScript.

Disadvantages:
There is always the possibility of strange type conversions occurring at runtime since typescript is ultimately converted into untyped JavaScript. Learning about types and various TS-specific notions requires time and effort on the part of the learner. Therefore, if an organization intends to switch to TypeScript and has JS developers who have no prior experience with TS, they won't immediately reach 100% productivity.

```
/// <reference path="./typings/zim/index.d.ts" />

// This version uses globals - good for stand-alone ZIM / CreateJS

var scaling:string = "fit"; // fit scales to fit the browser window
var width:number = 1024; // can go higher...
var height:number = 768;
var color:string = green;
var outerColor:string = dark;
var frame:Frame = new Frame(scaling, width, height, color, outerColor);
frame.on("ready", function() {
    zog("ready from ZIM Frame");

    var stage:Stage = frame.stage;
    var stageW:number = frame.width;
    var stageH:number = frame.height;
```

Figure 7 Making GUI frame using TypeScript - AltexSoft. (2016).

Front-End language for this project:

After researching and weighing the benefits and drawbacks of JavaScript, Dart, and typescript, this project will utilize JavaScript for the project's front end as it offers a wide range of tutorials, flexibility and compatibility that is required for the project.

2.3.2 Back – End

- **JavaScript**

  Advantages:
  Developers can work on both the front end and the back end by running JavaScript on the server. They will have a far better overall understanding of the project and eliminate the need for unnecessary communication across teams. A single developer may implement each feature from beginning to end. As a result, the project's business logic between the client and server APIs is considerably more consistent. Furthermore, JavaScript also offers a wide range of frameworks.

Disadvantages:
NPM itself, because to its irrational popularity, may occasionally be a problem. Every day, dozens or even hundreds of new packages are pushed there so the application can crash if it doesn't recognise these.

```
const http = require('http');

const hostname = '127.0.0.1';
const port = 3000;

const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('Hello World\n');
});

server.listen(port, hostname, () => {
  console.log(`Server running at http://${hostname}:${port}/`);
});
```

Figure 8 Configuring server using JavaScript - Team, D. (2019).

- **Python**

Advantages:
Having syntax similar to English, Python is a high-level programming language. Because of this, reading and comprehending the code is much simpler. No matter how many mistakes the program has, Python only displays one. Therefore, Debugging is easier. The vast Python standard library contains practically all the functions required for your work. Therefore, you are not dependent on other libraries.

Disadvantages:
Python's poor pace is partly a result of its dynamic nature, which requires it to perform more work when running code. Python is thus not employed in projects where speed is a crucial component.

```
class GameObject:

    def __init__(self, image_path, x, y, width, height):
        # Player image import and resize
        object_image = pygame.image.load(image_path)
        self.image = pygame.transform.scale(object_image, (width, height))

        self.x_pos = x
        self.y_pos = y

        self.width = width
        self.height = height

    # Draw the object by blitting it onto the background(game_screen)
    def draw(self, background):
        background.blit(self.image, (self.x_pos, self.y_pos))

```

Figure 9 Game object class in Python - TechVidvan Team (2019).

- **PHP**

Advantages:
The fact that PHP is available to everyone is one of its most important benefits. It is available for free download from an open source. Due to its great degree of adaptability, individuals may easily utilize it to integrate its features with those of several other programming languages. Furthermore, PHP also features an integrated database connection that aids in tying databases together.

Disadvantages:
PHP can be unreliable and occasionally makes mistakes. It may result in consumers having access to inaccurate information and expertise. It is not feasible to alter or modify the web applications' core behavior since PHP bans it. Frameworks for PHP operate differently from those for other languages that are comparable. Its features and performance might therefore deteriorate.

```php
<?php
$str_isset = "";
$bol_isset = isset($str_isset);

If ($bol_isset){
   echo "The variable is set";
}
else {
    echo "The variable is not set";
}
?>

```

Figure 10 Setting up variable in PHP- Prasanna (2021).

Back-End language for this project:

Following investigation and consideration of the advantages and disadvantages of JavaScript, Python, and PHP, this project will use JavaScript for the project's back-end since it provides the ease of access to work on the front and back-end seamlessly and will also keep the API more consistent.

2.3.3 Frameworks

- **VueJs**

  Advantages:
  The data binding functionality of VueJS allows users to edit or assign values to HTML properties, modify the style, and assign classes by using the binding directive v-bind. In addition, Vue makes use of template syntax based on HTML. With vue-assistance, switching between pages is simple. All DOM modifications are absorbed by the virtual DOM, which essentially replicates the primary DOM component, and are made available as JavaScript data structures.

  Disadvantages:
  The Chinese language used in the framework's code presents challenges for non-Chinese developers. Furthermore, Vue.js does not currently support a huge number of plugins because it is a new technology and is still in its early phases of development. Moreover, code that is too

flexible will have more mistakes and inconsistent behaviour. Project delays are brought on by an increase in errors and anomalies.

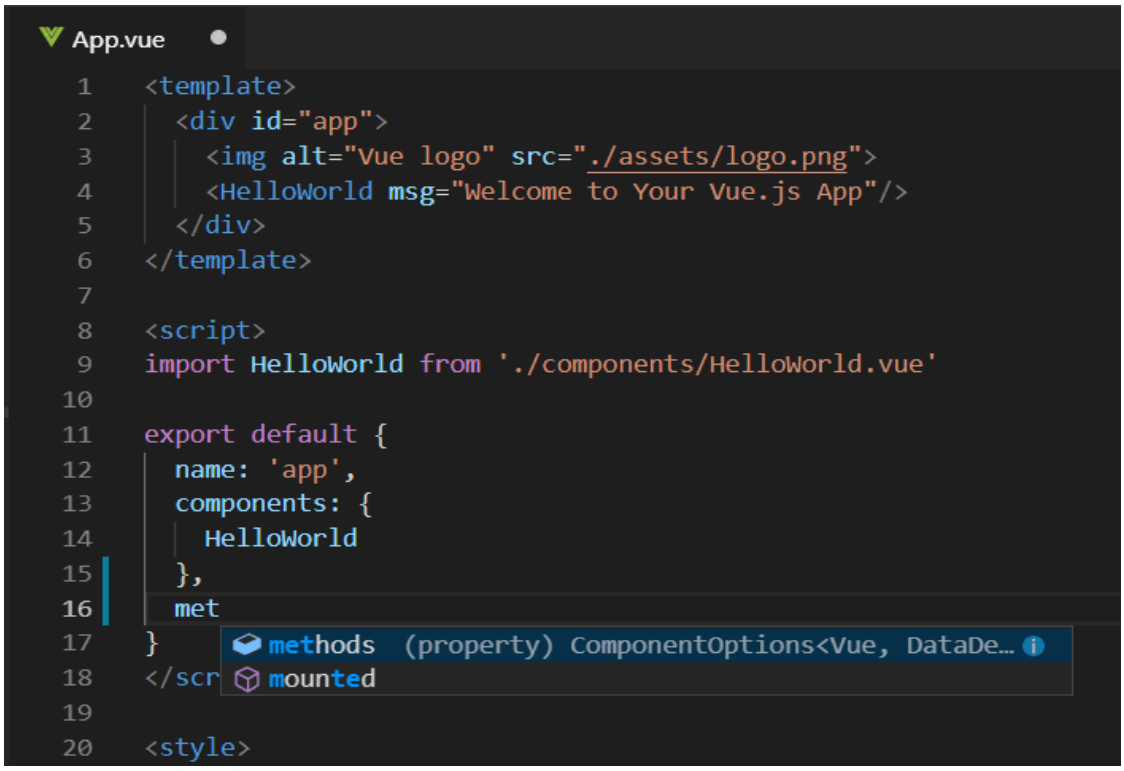


Figure 11 Initialising component in Vue.js - editor (2022).

- **ReactJS**

Advantages:
React is a package used for building web apps that enables programmers to make quick single-page programs and user interfaces. With virtual DOM, developers may make changes to the app's tiniest parts without affecting the whole thing. One of the elements that affects Reacts appeal is reusable components. They greatly simplify the development process by allowing developers to reuse parts in subsequent projects rather of having to create the same code from start for identical tasks. The fact that ReactJs is an open-source project may be credited with the library's significant community support.

Disadvantages:
There just isn't enough time to provide thorough documentation because of the volume of changes and new releases. The only resources remaining for developers are bare-bones text guidelines. Due to the library's constant evolution and the resulting need for developers to continuously learn new mechanics or processes, development speed is quite slow.

```
import React, { Component } from 'react'

export default class GameCharacter extends Component {
    constructor(props){
        super(props);
        this.state = {
            firstName: 'Rahul'
        }
    }
    render() {
        return (
            <div>
                <h1>Hello {this.state.firstName}</h1>
            </div>
        )
    }
}
```

Figure 12 React code to print first name - Bhatt, (2020)

- **NodeJS**

Advantages:
Nodejs enables non-blocking I/O systems, which let you handle several requests concurrently. Users may modify and further enhance Node.js to suit the users' unique needs. Furthermore, both client-side and server-side may be created with Node.js. Node.js is fortunate to have a vibrant development community that constantly works to improve it.

Disadvantages:
Nodejs lacks a robust and extensive library system in compared to other programming languages. Furthermore, Node.js's constantly evolving API is a serious drawback as well. Moreover, Performance becomes a problem when there is a lot of calculation involved.

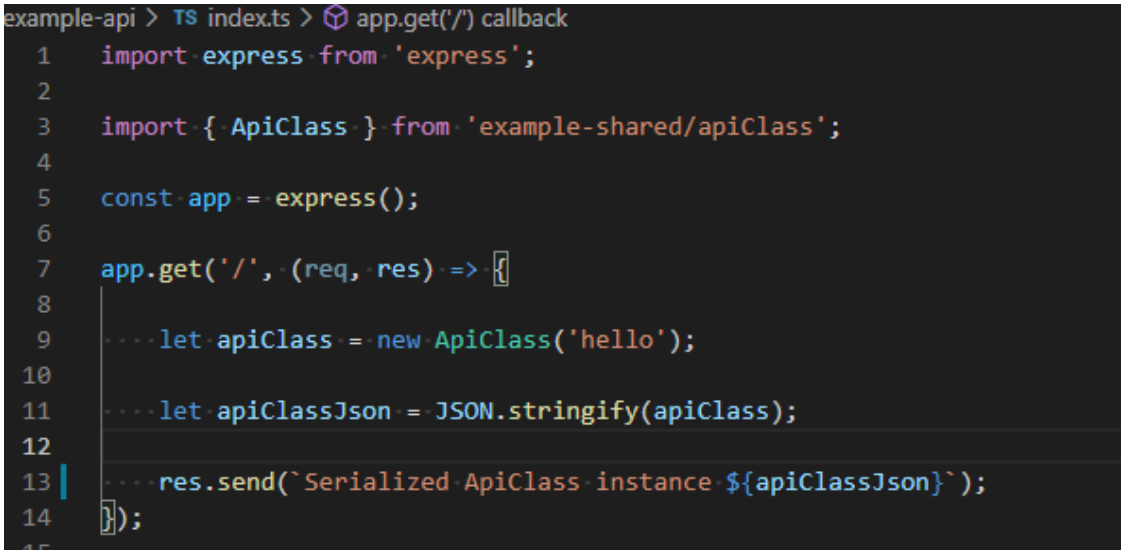


Figure 13 Running get request using Express - Express (2020).

The frameworks for this project:

This project will be using Node.JS and ReactJs for this project after investigating and considering the advantages and disadvantages of ReactJS, VueJs, and Node.JS. This is because Node.JS can run both the client and the server at the same time, connect to databases, and offers a variety of tutorials and documentation. Furthermore, Compared to using plain JavaScript, ReactJs makes it easier to create interactive user interfaces and online apps fast and effectively.

2.3.4 Database

- **MySQL**

  Advantages:
  With this database engine, it is possible to modify the tool's functionality and handle data from different table types by selecting from a variety of storage engines. It offers a ton of functionality even though it is a free database engine. A number of user interfaces can also be applied. It may also be set up to work with several databases, including DB2 and Oracle.

Disadvantages:

To get MySQL to perform tasks that other systems carry out automatically, such as making incremental backups, may take a significant amount of time and effort. Furthermore, XML and OLAP are not supported natively. Moreover, For the free version, support is offered, but it costs money.

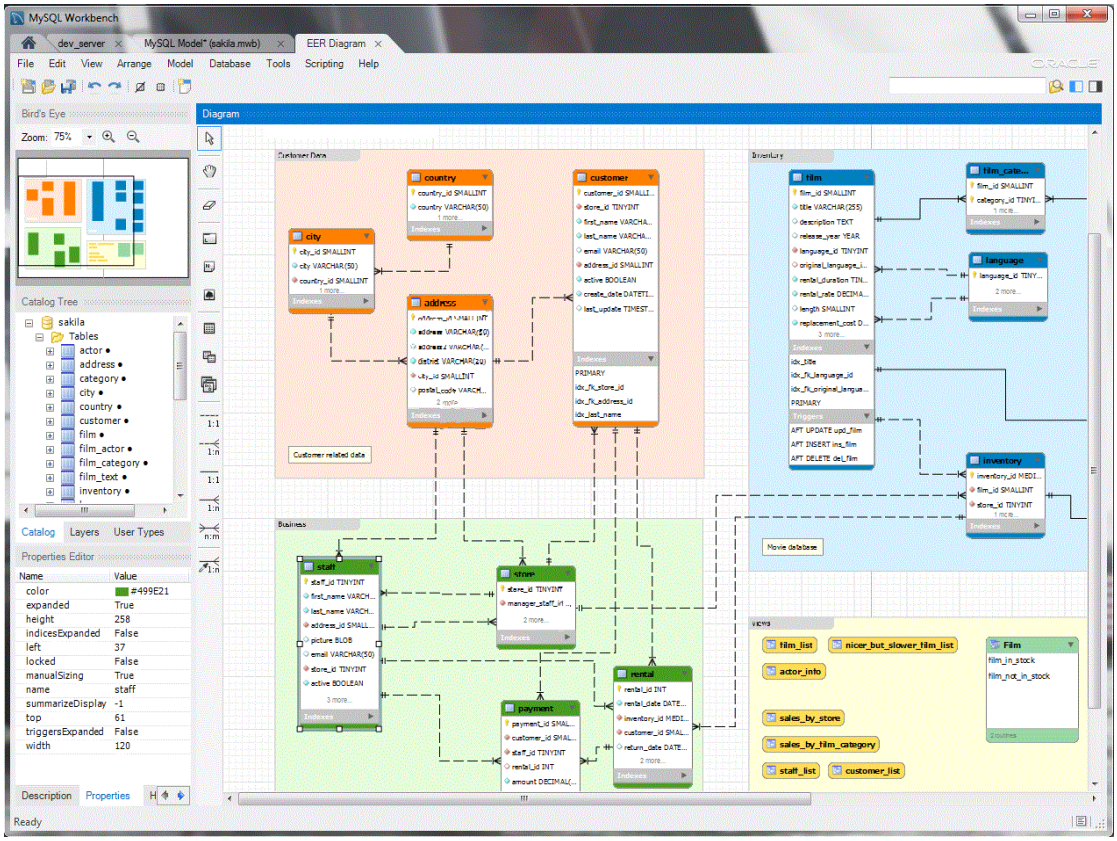


Figure 14 Tables in MySQL - MySQL (2021).

- **PostgreSQL**

Advantages:

Numerous settings, including virtual, physical, and cloud-based ones, can host this database management engine. Terabytes of data may be handled by this scalable database management engine. JSON is supported. There are several pre-set functions available. Also, there are numerous interfaces available.

Disadvantages:

You might find yourself looking things up online because the documentation is sometimes patchy. It can be difficult to configure. During large bulk operations or read queries, speed may be compromised.

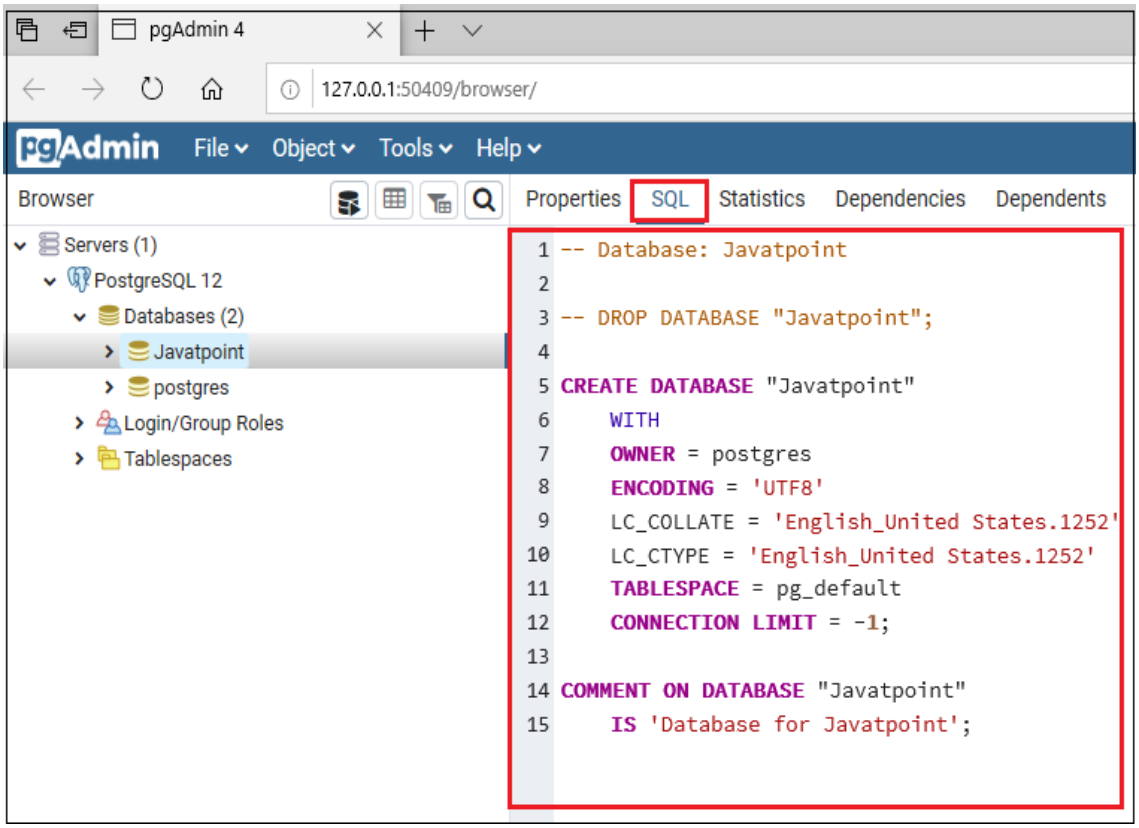


Figure 15 Initialising Database in PostgreSQL PostgreSQL (2017).

- **MongoDB**

Advantages:

Applications that employ both structured and unstructured data should use MongoDB. The database engine connects databases to applications using MongoDB database drivers, and it is incredibly flexible. It is quick and simple to use. Other NoSQL documents and JSON are supported by the engine. It is simple and quick to store and access data of any structure. Schema can be created instantly.

Disadvantages:

There isn't a query language for SQL. Although there are tools for doing so, using the engine requires an additional step. It can take a while to set up. The default settings lack security.

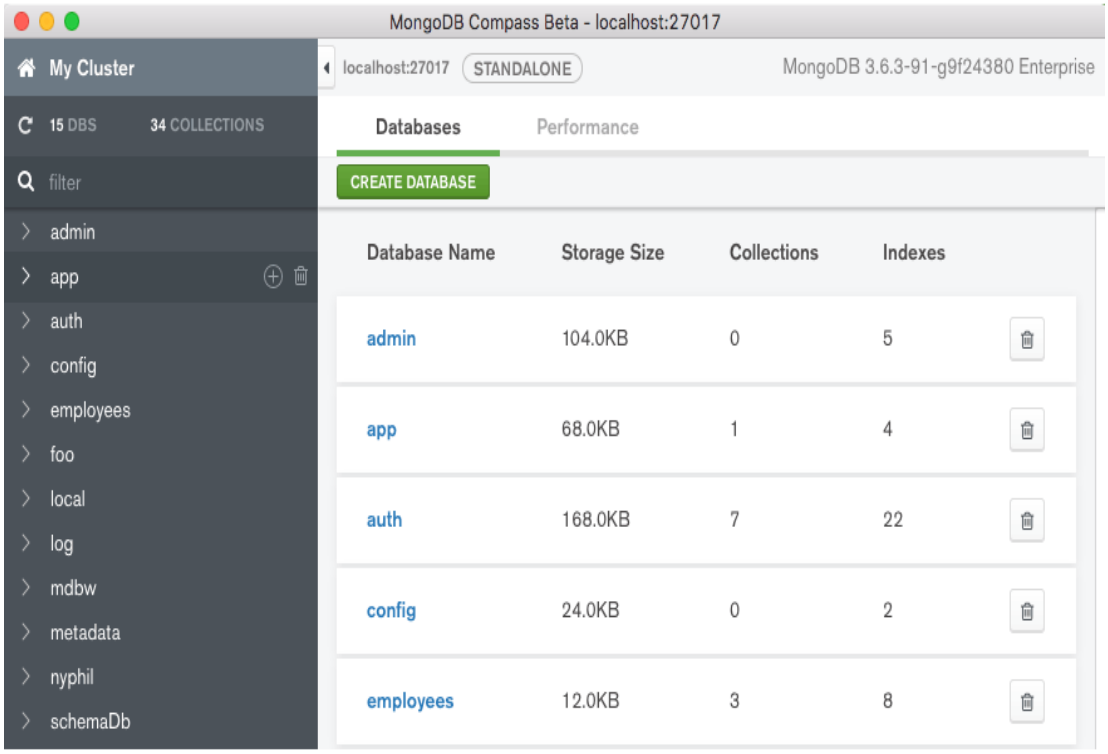


Figure 16 Databases in MongoDB - MongoDB (2022).

The Database for this Project:

Mongodb will be used for the project's database after research and evaluation of the benefits and drawbacks of Mongodb, MySQL, and PostgreSQL. Mongodb is a flexible database that functions exceptionally well with most programming languages. Additionally, there is a ton of available documentation and tutorials, making it quick and simple to use.

2.3.5 Payment Capture

The Stripe platform enables online financial management and the acceptance and processing of payments. One of the most widely used eCommerce systems is this one, which is used by software platforms, marketplaces, subscription businesses, and other online and offline merchants. Stripe is unique because it combines payment processing with gateway functionality. Due to the reasonable Stripe payment processing fees, this platform is regarded as the most affordable way to manage eCommerce.

Stripe works by using a payment gateway and a processor which are necessary for every online transaction. The first one safely records and sends the processor the buyer's credit card information. After subtracting card-related fees, the payment processor then transfers funds to a different account. The money is then forwarded to the merchant's bank account.

Furthermore, Stripe works with a variety of payment options. The following global payment methods are supported by the Stripe API: Alipay, Apple Pay, Google Pay, Microsoft Pay, American Express Express Checkout, Masterpass, Visa Checkout, and WeChat Pay. Stripe also accepts regional payment methods. ACH, Bancontact, EPS, Giropay, iDEAL, Klarna, Multibanco, P24, SEPA Direct Debit, and SOFORT are a few of them.

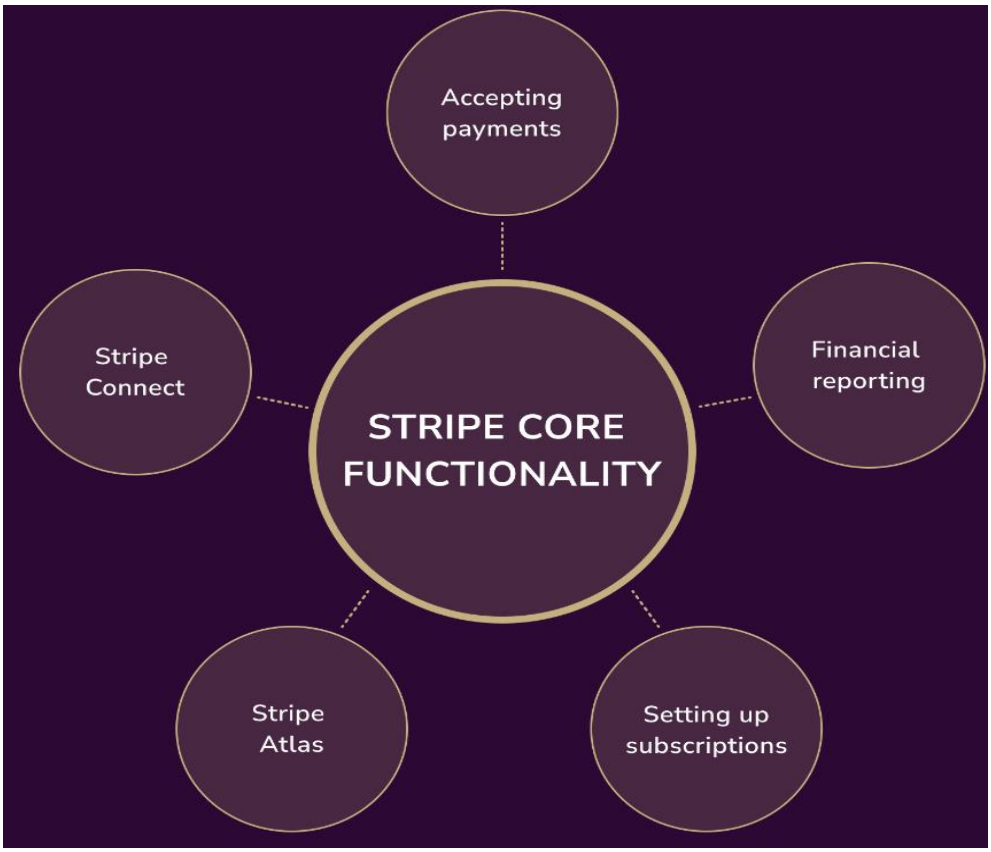


Figure 17 Picture showing stipe functionality - Merchant Maverick. (2020).

2.3.6 Cloud Computing

Businesses may launch their websites and web apps at a reasonable cost thanks to Amazon Web Services' cloud web hosting offerings. A single web server that hosts either a Content Management System (CMS), such as an eCommerce program, or a simple website is typical of such websites. The software makes it simple to create, modify, maintain, and deliver the website's content.

For low to medium traffic sites with numerous authors or a single author and more regular content updates, such as marketing websites, content websites, or blogs, simple websites are excellent. They provide websites a straightforward foundation from which to expand. These sites are often inexpensive, but they also need IT management of the web server because they are designed to be highly available or scalable.

Figure 18 Picture showing the benefits of AWS - Kcsitglobal (2022).

2.4. Other Relevant Research

2.4.1 Questionnaire and Survey

As part of the need's elicitation step, a Muslim customers-specific questionnaire was also developed. The survey was sent using Google Forms, and all respondents will remain anonymous. This questionnaire has a number of objectives. In order to better understand how customers, perceive websites that provide Islamic and traditional Muslim apparel and accessories, a questionnaire was created.

Using this questionnaire, some important information was gathered. 78% of respondents felt that there was a serious lack of websites that offered Islamic and traditional Muslim apparel and accessories. Furthermore, 73.7% think that existing Islamic online stores are inferior to other contemporary online retailers. Furthermore, 63.2% said that the prices offered at actual Islamic apparel and accessory stores at the time were not fair. Finally, 52.6% of participants said they preferred to purchase online.

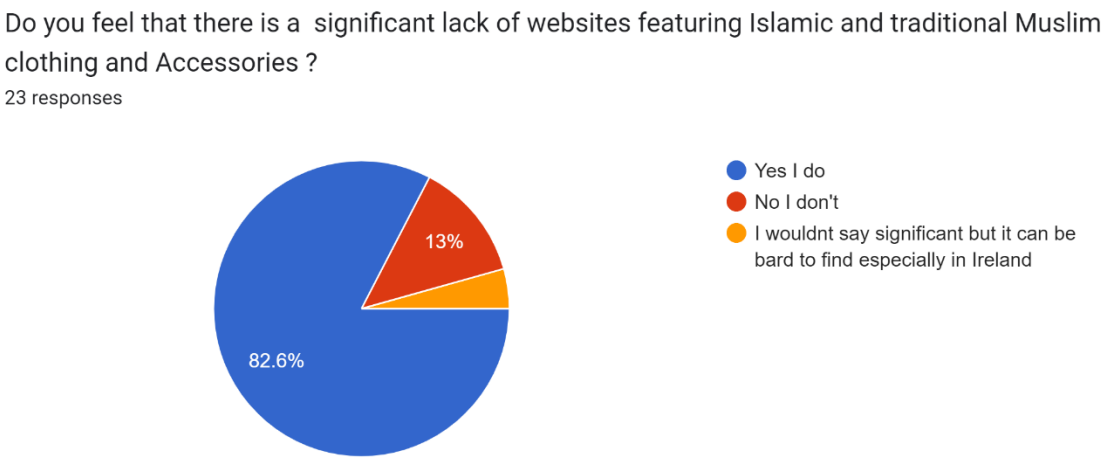


Figure 19 - Pie chart 1

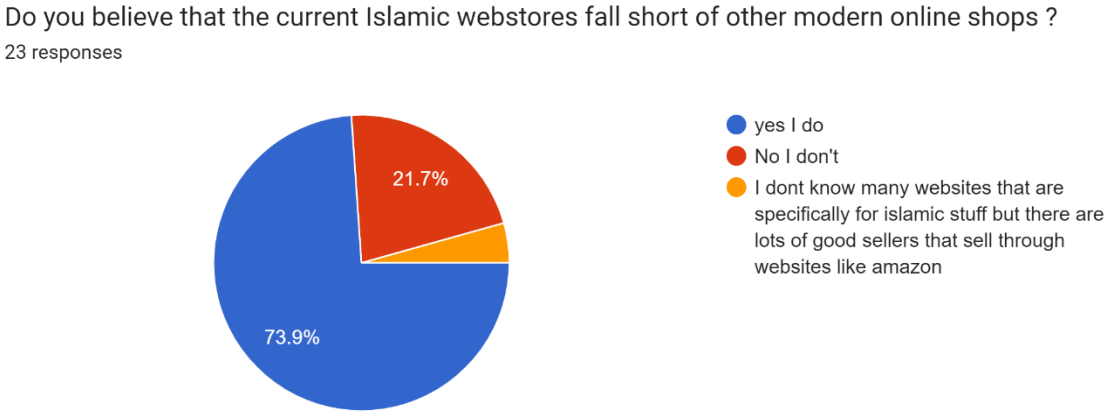


Figure 20 - Pie chart 2

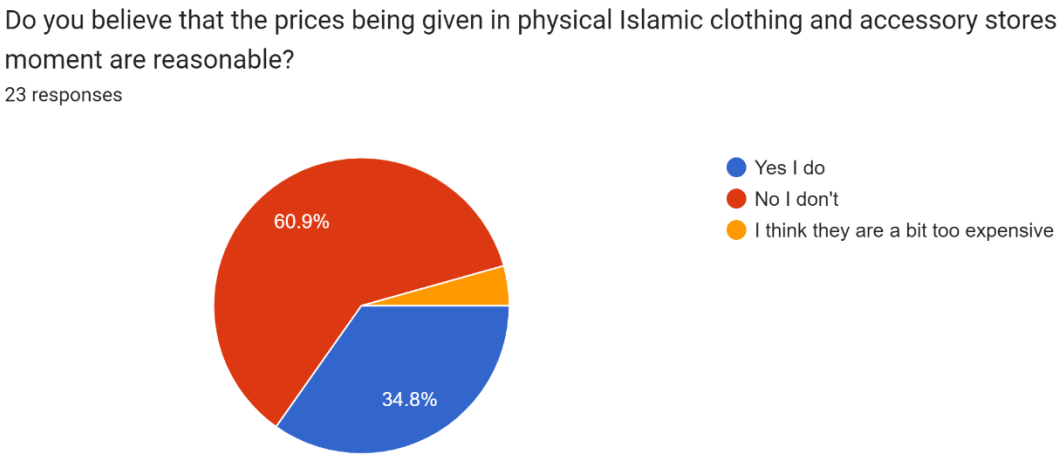


Figure 21 - Pie chart 3

Figure 22 - Pie chart 4

2.5. Existing Final Year Projects

2.5.1 Existing project 1

Title: Third-Level Education Course Suggestion Web Application

Student: Glen Holmes

Description

The resulting intent is to provide those considering third-level education, a framework that not only finds their most suited course, but in turn reduces the high percentage of first year dropout rates.

What is complex in this project?

This project is complex as it goes through validated scientific methods like Holland Codes, and a complex algorithm aims to simplify this decision process. Potential students will be matched to a course on individual traits such as capability and personality, and to a college on items such as hobbies, interests, amenities, and location.

What technical architecture was used

Programming using JavaScript, Database using Mongo dB, Programming using Node.JS

Key strengths and Weaknesses of this project

This project tackled a very important issue which even everyone had to deal with when transitioning from second level to third level Education. Some of the strengths would be that developer did a lot of good research and had a good understanding of the applications used. Using Node.JS with JavaScript and Mongo dB as a database is a good combination of software to use as they all complement and work well with each other. Changing the front-end CSS to make the website a bit more aesthetically appealing. Other than that, it's a pretty solid Project.

2.5.2 Existing project 2

Title: An eLearning system for anonymous feedback sessions with built in polling, using node.js.

Student: Daniel Hogan

Description

This application focuses on creating an eLearning system that tries to encompass the better features of the compared systems. Comparisons are done by features and Nielsen's heuristics. Furthermore, the application also allows students to use a web browser on their laptop or mobile device that they bring into the classroom, to ask an anonymous question through a website.

What is complex in this project

This project showed that creating a mobile interface that is identical to the desktop version without sacrificing any functionality or rendering the mobile interface useless is a challenging task. Additionally, this is meant to be minimized by prototyping both interfaces.

What technical architecture was used

His project uses technologies like MongoDB6 and Node.js7 with expressJS8 on the server and AngularJS on the client.

Key strengths and weaknesses

The developer's choice of technology to build the website was very good as well as the Implementation of how it all came together. However, his idea lacked novelty and originality. Furthermore, the developer could have worked more on the visuals of the website as looked kind of bland. Utilising a bootstrap to make it look nicer. Overall, it was an acceptable project.

2.6. Conclusions

After conducting further research and weighing the advantages and disadvantages of JavaScript, Dart, and typescript. The decision to use JavaScript for the project's front end because it provides the project with the flexibility, compatibility, and wide range of tutorials that are needed. After researching and weighing the benefits and drawbacks of ReactJS, VueJs, and Node.JS, this project will be using Node.JS. Additionally, Mongo dB will be employed for the project's database following investigation and assessment of the advantages and disadvantages of Mongo dB, MySQL, and PostgreSQL.

3. Experiment Design

3.1 Introduction

The design procedure utilized for this project is covered in this chapter. The comparison of several software techniques will come first. The one that is best appropriate for this project will be selected following an assessment of its benefits and drawbacks. After that, a system overview will be given, during which low-fidelity and high-fidelity wireframes will be covered. Additionally, the three-tier system architecture will be examined. moreover, a look at a use case diagram.

3.2. Software Methodology

Well-managed projects are successful projects. The developer must select the software development methodology that will be most effective for the project at hand if they are to manage the project effectively. Every methodology differs in its strengths and weaknesses as well as the motivations behind its creation. The most popular software development methodologies are described here, along with an explanation of why they exist.

3.2.1 Agile

In all agile development methodologies, software is created in iterations that include small increments of new functionality. The agile development methodology comes in a variety of forms, such as scrum, crystal, extreme programming, and feature-driven development.

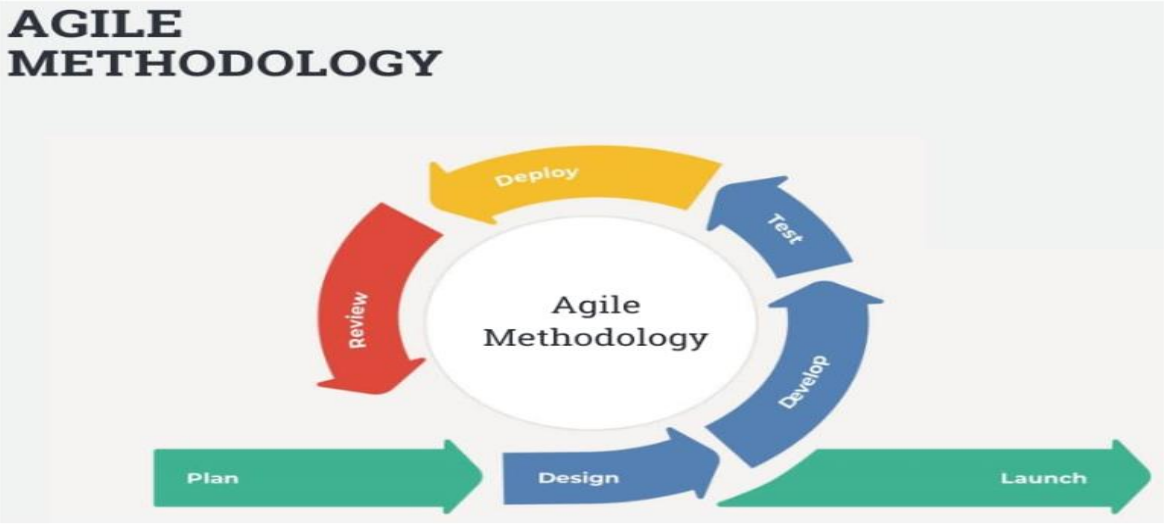


Figure 23 - Agile methodology example - Interqualitybg.com. (2019).

- **Advantages:**
  Agile software development's main advantage is that it enables iterative software releases. Iterative releases increase productivity by enabling teams to identify and correct flaws and set expectations early on. With regular incremental improvements, they also enable users to enjoy the benefits of software earlier.

- **Disadvantages:**
  Agile development methodologies rely on in-the-moment communication, so new users frequently lack the necessary background information to get started. They are time- and labour-intensive because developers must finish each feature within each iteration before asking users for approval.

3.2.2 DevOps

The focus of DevOps deployment is organizational change, which improves communication between the divisions in charge of various stages of the development life cycle, including development, quality assurance, and operations.

Figure 24 - DevOps methodology example - Raycad (2019).

- **Advantages:**

Along with being a development methodology, DevOps also consists of a set of procedures that help to maintain an organizational culture. The focus of DevOps deployment is organizational change, which improves communication between the divisions in charge of various stages of the development life cycle, including development, quality assurance, and operations.

- **Disadvantages:**
  Some customers do not desire frequent system updates. Thereafter, before a project can enter the operations phase, strict testing requirements may be imposed by regulations in some industries. Moreover, Unrecognized problems may creep into production if various departments use various environments. Additionally, human interaction is required for some quality attributes, which slows down the delivery pipeline.

3.2.3 Waterfall

The waterfall method is frequently regarded as the oldest approach to software development. The requirements, design, implementation, verification, and maintenance phases of the waterfall method each focus on a different objective. Before moving on to the next phase, each phase must be finished completely. In most cases, there is no procedure for going back and changing the project or direction.

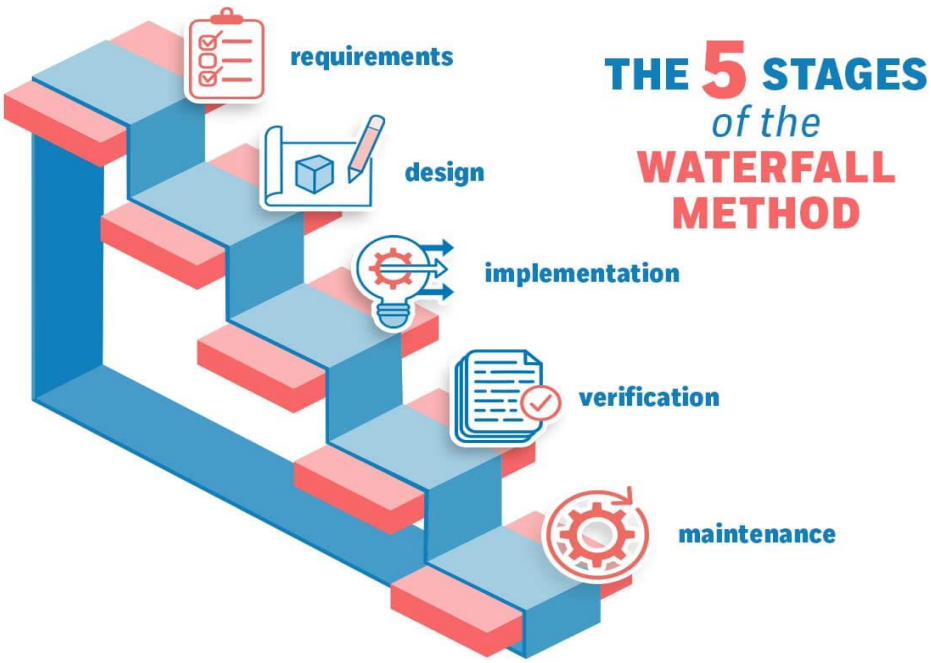


Figure 25 - Waterfall methodology example - Indeed Career Guide. (2021).

- **Advantages:**
  The waterfall development method is simple to understand and manage because it is linear. The waterfall method is most effective when used on projects with precise goals and consistent requirements. The waterfall development methodology may be most advantageous for project teams and project managers with less experience, as well as for teams whose membership changes frequently.

- **Disadvantages:**
  The rigid structure and stringent controls of the waterfall development method make it frequently expensive and slow. Users of the waterfall method may investigate alternative software development methodologies as a result of these drawbacks.

3.2.4 Rapid Application

The requirements planning, user design, construction, and cutover phases make up the four stages of the rapid application development methodology. Up until the user certifies that the product satisfies all requirements, the user design and construction phases are repeated.

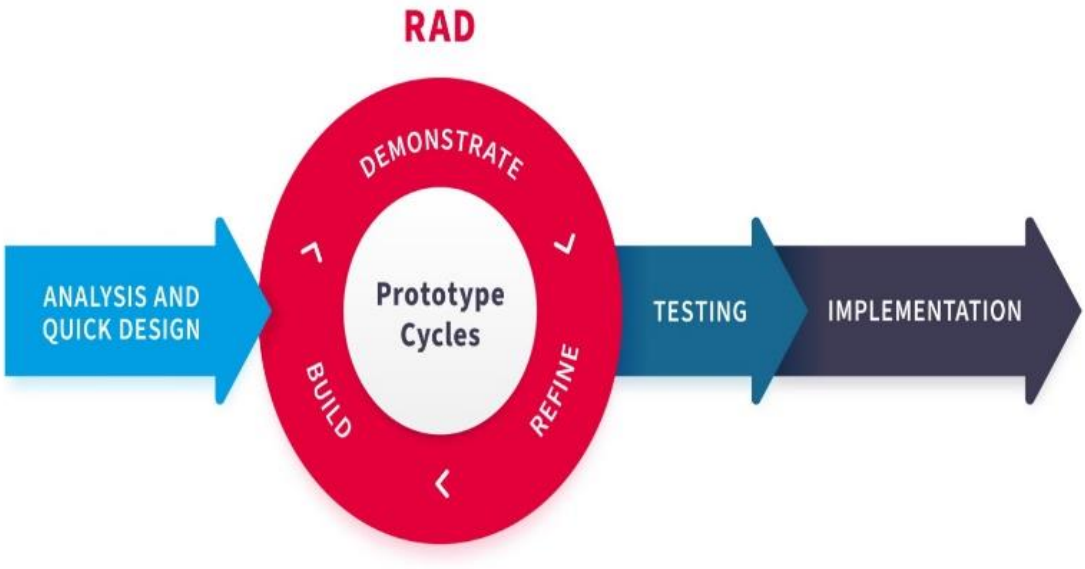


Figure 26 - Rapid Application methodology example - getbreakout.com. (2018).

- **Advantages:**
  The most successful projects for rapid application development are those that have a clearly defined business objective, a defined user group, and are not computationally complex. Time-sensitive small to medium-sized projects benefit the most from RAD.

- **Disadvantages:**
  A stable team composition with highly skilled developers and users with in-depth application knowledge is necessary for rapid application development. A compressed development timeline that calls for approval following each stage of construction necessitates in-depth knowledge. People that don't fit these criteria won't likely gain anything from RAD.

The methodology chosen for this project:

Agile methodology will be used for this project because it takes an iterative approach to software development projects and guarantees that input can be promptly implemented and that responsive modifications can be made at each step of a sprint or product cycle. There will be frequent meetings with the client. The project will be broken into smaller features that will be implemented. These features will then be created into tasks that need to be completed. Thereafter, sprints (meetings) will be held with the supervisor and the client to make sure all priorities are met.

3.3. Overview of System

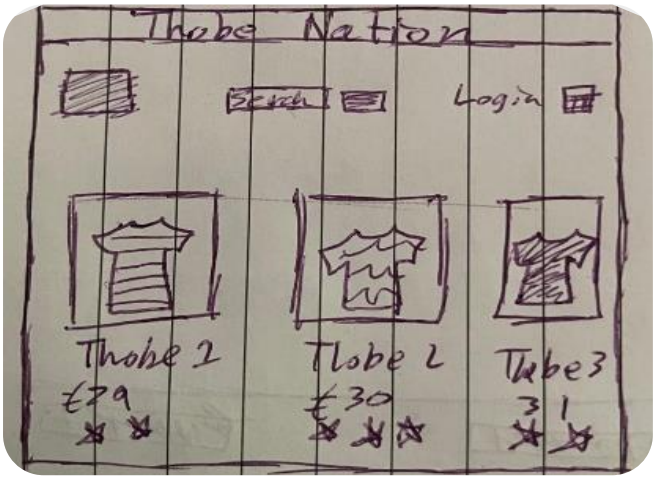
3.3.1 Wireframes

A wireframe is a straightforward visual diagram that shows the fundamental layout of a website or digital product. Think of it as the model for the project's final design. Though wireframes are often generated by designers, they must be simple enough for other designers, stakeholders, developers, and users to comprehend the concepts.

In a normal design process, wireframes are developed as either hand-drawn drawings or high-fidelity mock-ups or prototypes. There are two categories of wireframing: low fidelity and high fidelity.

3.3.2 low fidelity prototype

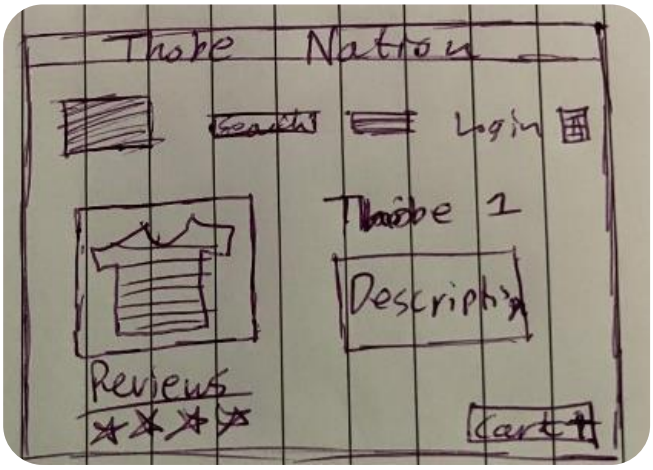
The most basic kind of wireframing is low quality. The simplest approach to depict ideas is still using paper and a pen but developing wireframes in Balsamic which will make it simple to share them and make sure everyone has access to the most recent ideas as that has been iterated. Grayscale low-fidelity wireframes that concentrate on layout and high-level interactions are used below.



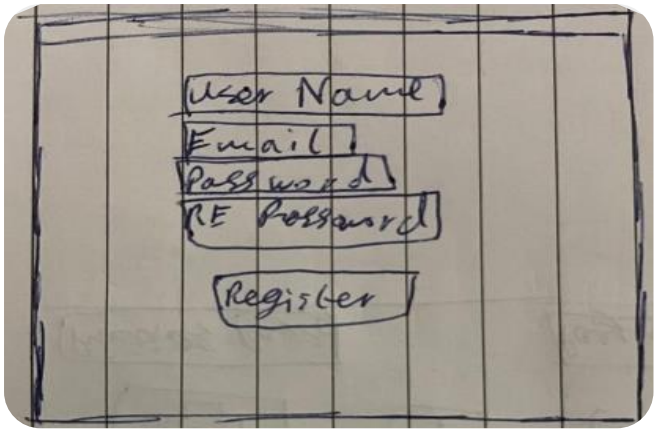
This is the product home page, where buyers may browse a selection of goods, see their prices, and decide whether to click for further details. Additionally, users have the option to click to see reviews. They have the option to log in and access their basket through this page.

Drawing figure 1

When a consumer clicks to examine product details, they may view a bigger image and read a description of the product's features. They have the option to continue shopping for more items or add the item to their basket.



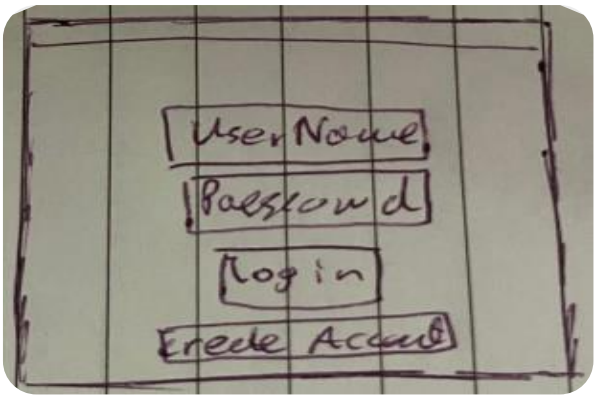
Drawing figure 2



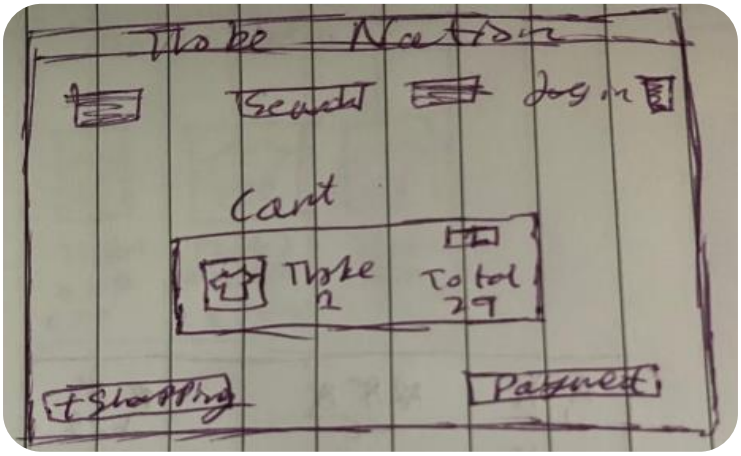
Drawing figure 3.

When a consumer clicks "Sign Up," it indicates that they wish to make purchases. They will be sent to a screen where they must provide their username, email address, and password before being asked to register.

A consumer can submit their information to log in after signing up. The password will be validated, and if it doesn't match, the user cannot sign in.



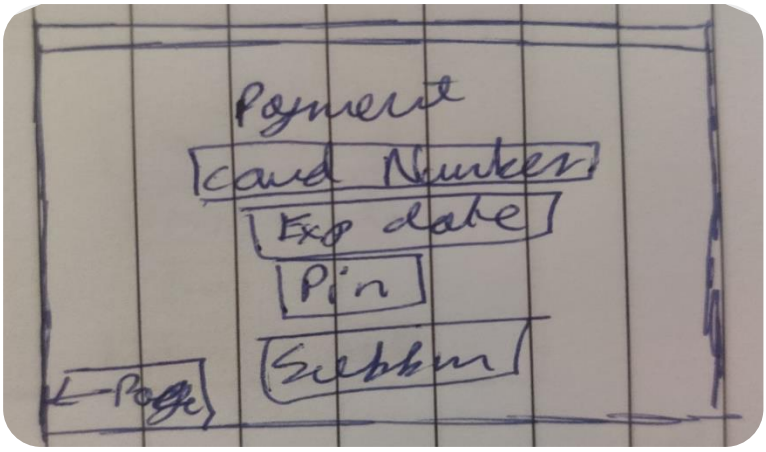
Drawing figure 5.



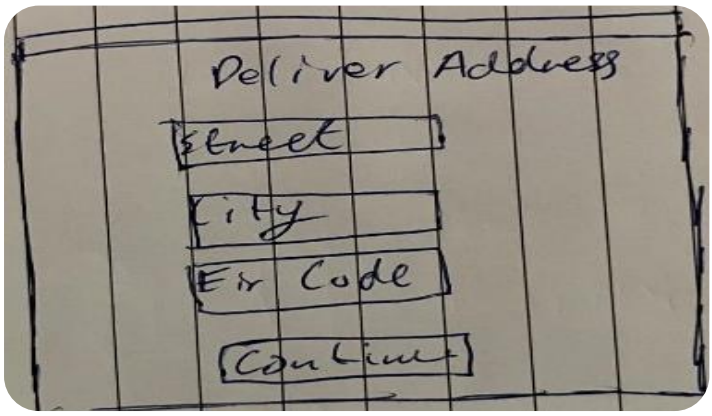
Drawing figure 6.

This is the cart page, where the user can view all the items they've chosen, as well as pick the quantity and enter the total cost.

This is the payment page which will allow the user to pay by either credit/debit cards or they can use PayPal.



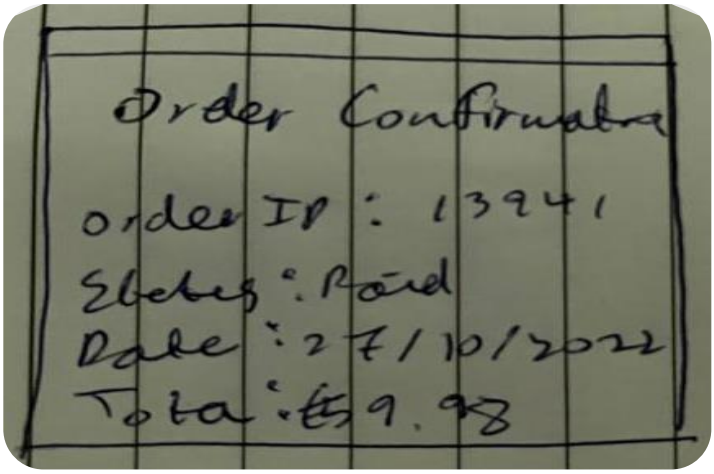
Drawing figure 7



Drawing figure 8.

This is the delivery page, where the customer will enter their address and it will be saved for the admin to send their ordered items.

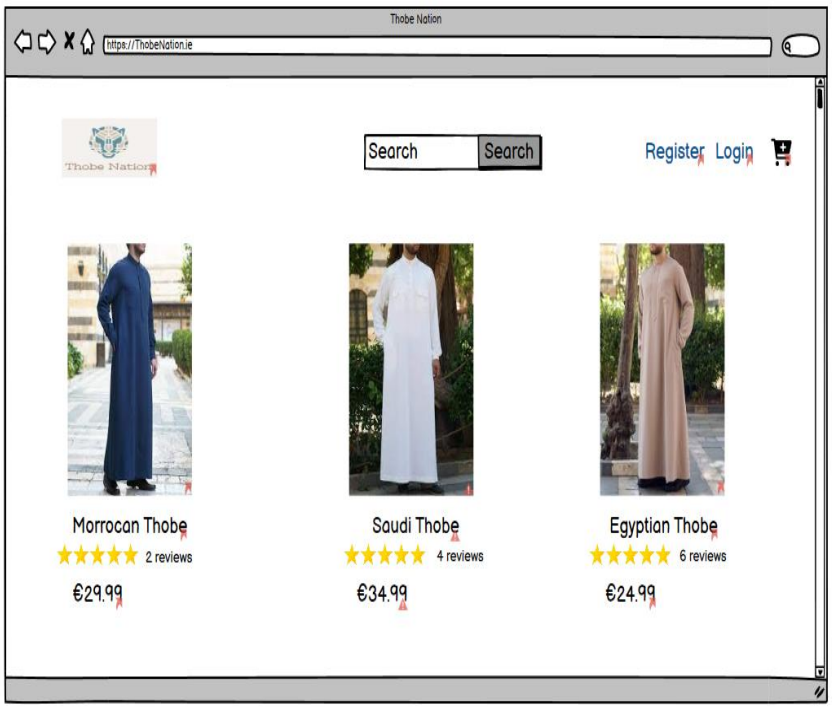
This page will show the confirmation details and order information it will display the order Identification, payment status, date and total.



Drawing figure 9.

3.3.3 High fidelity prototype

In high fidelity wireframing, one would add more of the intricate details to explain the low quality wireframes a person created. Colours, visuals, and font styles are examples of branding signifiers that are included in high fidelity wireframes. UI components have a realistic appearance and may even include textures and shadows. A designer could also decide to include visuals at this point.

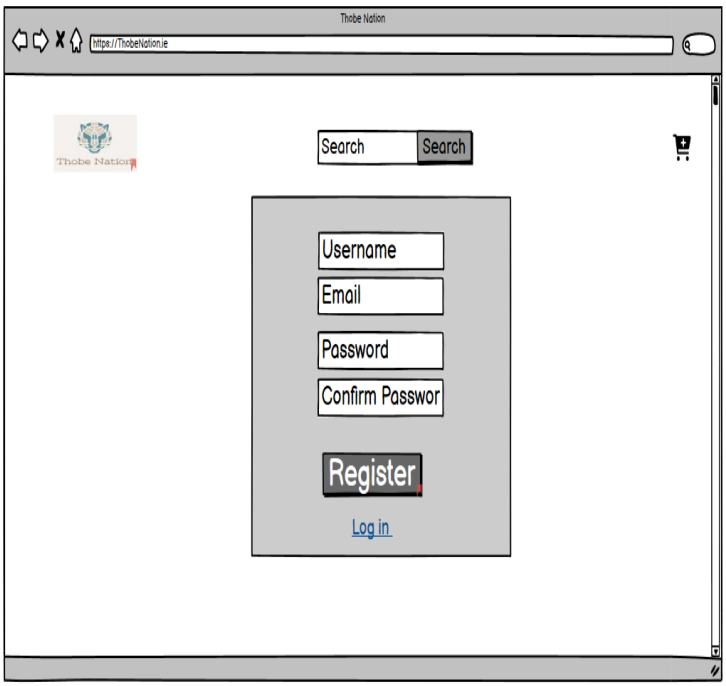


Customers may explore various products on this product home page, view their pricing, and choose whether to click for further information. Users also have the choice to click to see reviews. They can log in and access this page to access

Wireframe figure - 1

A customer may view a larger image and read a detailed explanation of the product's characteristics by clicking to view the product information. They can choose to either add the item to their cart or keep shopping for additional things.

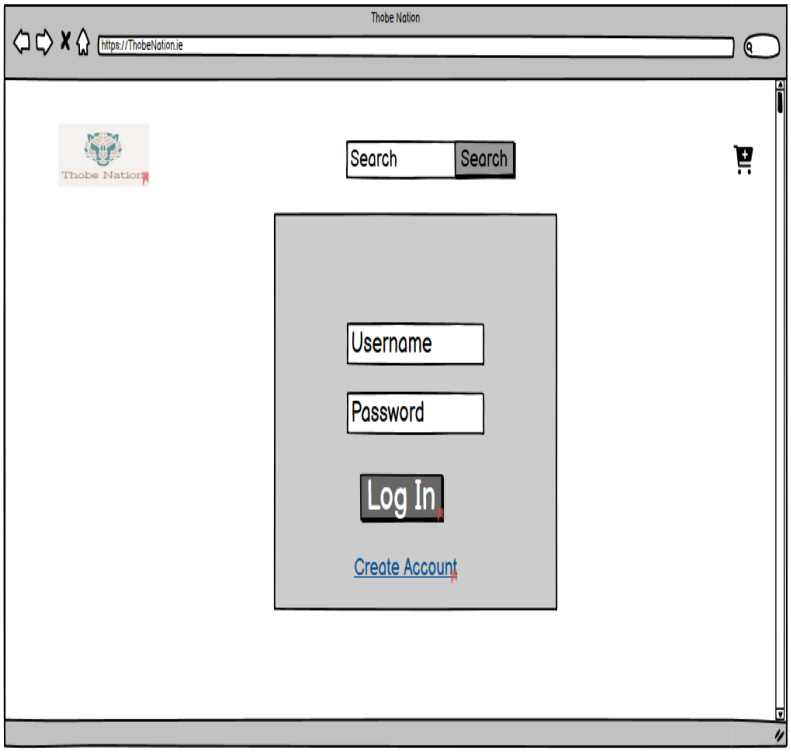
Wireframe figure - 2



Wireframe figure - 3

Consumers who click "Sign Up" do so because they want to make purchases. Before being prompted to register, they will be directed to a screen where they must enter their username, email address, and password.

After registering, a user can enter their data to log in. If the password is not verified and matches, the user cannot sign in.



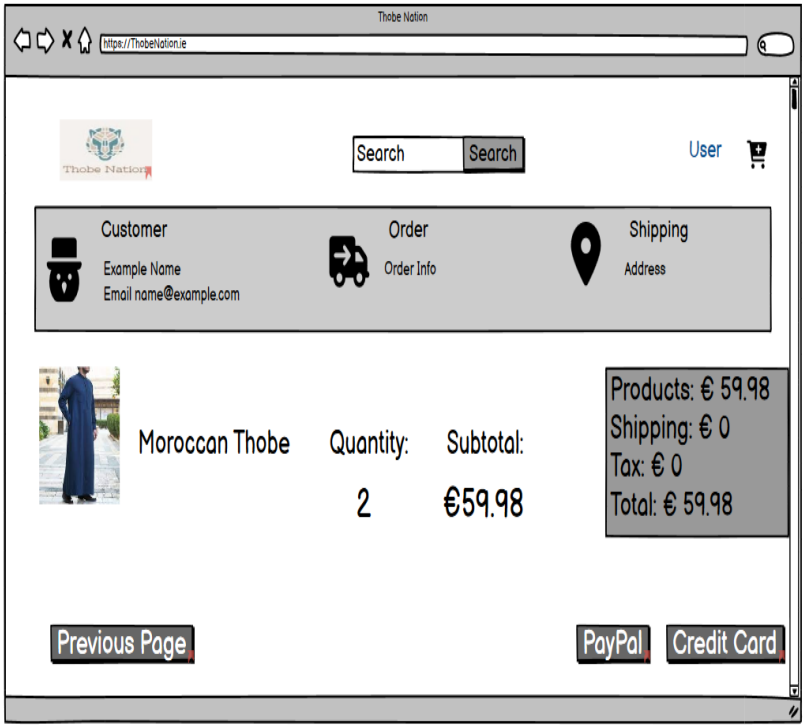
Wireframe figure - 4

The user may examine all the things they've selected on this page, as well as choose the amount and enter.

Wireframe figure - 5

The customer enters their address on this delivery page, where it is saved in the database for the admin to use when sending the items, they have ordered.

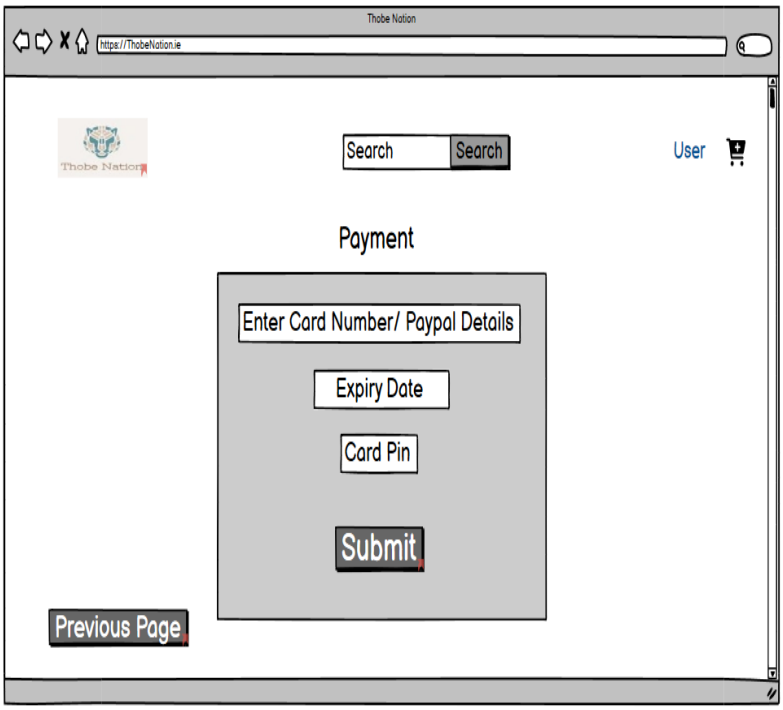
Wireframe figure – 6



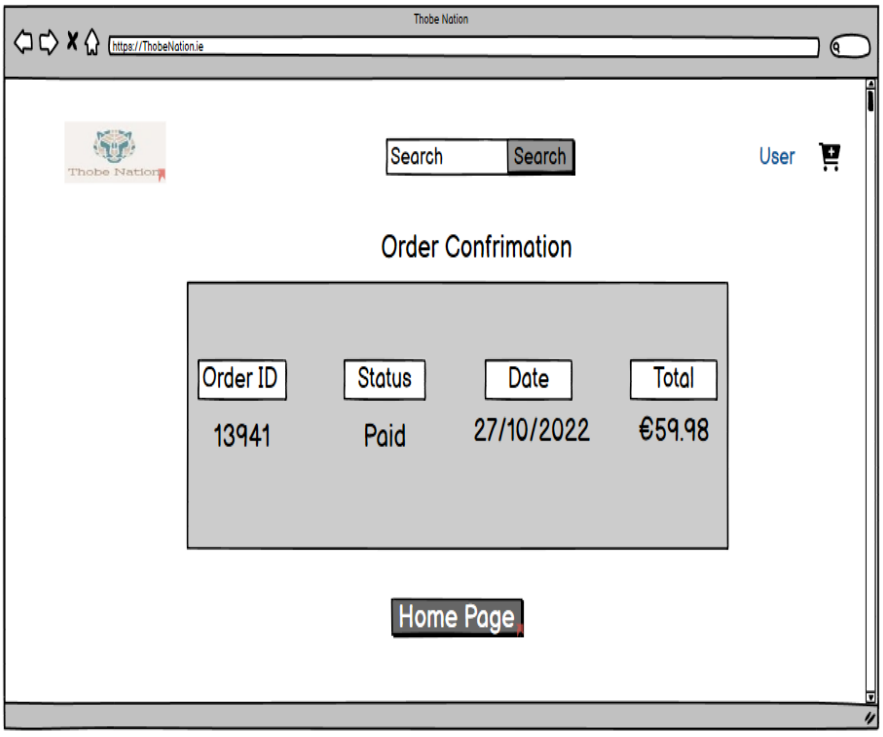
Wireframe figure - 7

This is review page where the customer can have a final look to see if all the order details and quantity is correct.

This is the payment page where users can use PayPal or credit/debit cards to make payments.



Wireframe figure - 8



The order IT, payment status, date, and total will be displayed on this page along with the confirmation information.

Wireframe figure – 9

3.4 System Architecture

The three-tier design that will be used for this project is a client-server architecture in which the user interface, data access, functional process logic, and computer data storage are all built and maintained as distinct modules on different platforms. Furthermore, a three-tier architecture enables the independent upgrade or replacement of any one of the three tiers.

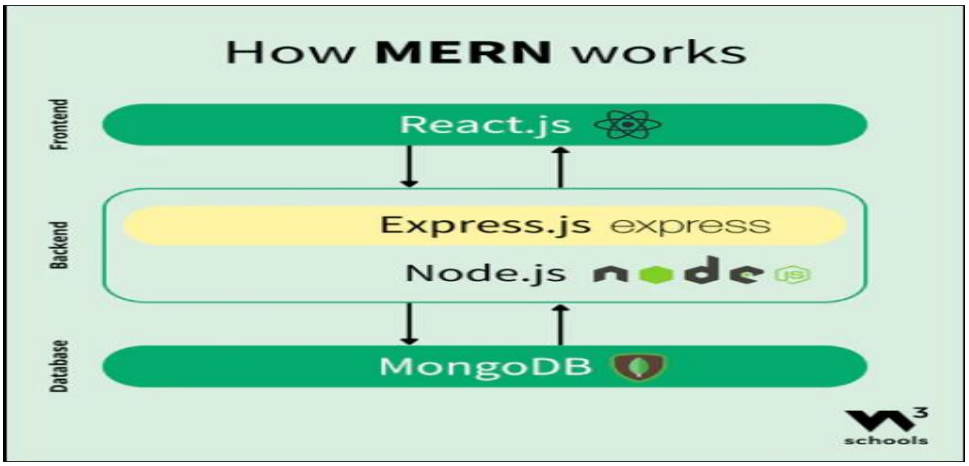


Figure 27 - Picture displaying the structure of a MERN project - Thorsell-Arntsen, T. (2022).

- **Presentation Tier:**
  React.js, a declarative JavaScript framework for building dynamic client-side apps in HTML, makes up the top layer of the MERN stack. React enables one to link basic components to data on the back-end server, connect complicated interfaces to those connections, and render those interfaces as HTML. React excels at managing stateful, data-driven interfaces with little effort and code, and it includes all the features one would expect from a contemporary web framework, including excellent support for forms, error handling, events, lists, and more.

- **Application Tier:**
  The server-side framework Express.js, which functions inside a Node.js server, is the next step below. It is true that Express.js describes itself as a fast and simple web framework for NodeJs. ExpressJs offers robust models for managing HTTP requests and replies as well as URL routing (correlating an incoming URL with a server function).This allows for connection to the Express.js functions that power the application by sending XML HTTP Requests (XHRs), GET requests, or POST requests from the React.js front end. These functions then access and change data in the MongoDB database using the Node.js drivers for MongoDB, either through callbacks or promises.

- **Data tier:**
  The first two application tiers can easily be integrated by using a MongoDB data base. MongoDB can help with this since it allows JSON documents written in React.js front end to be forwarded to the Express.js server for processing and, if they're valid, direct storage in MongoDB for later retrieval. Once more, if one is creating on the cloud, it is possible to use Atlas and for local host one could use Compass.

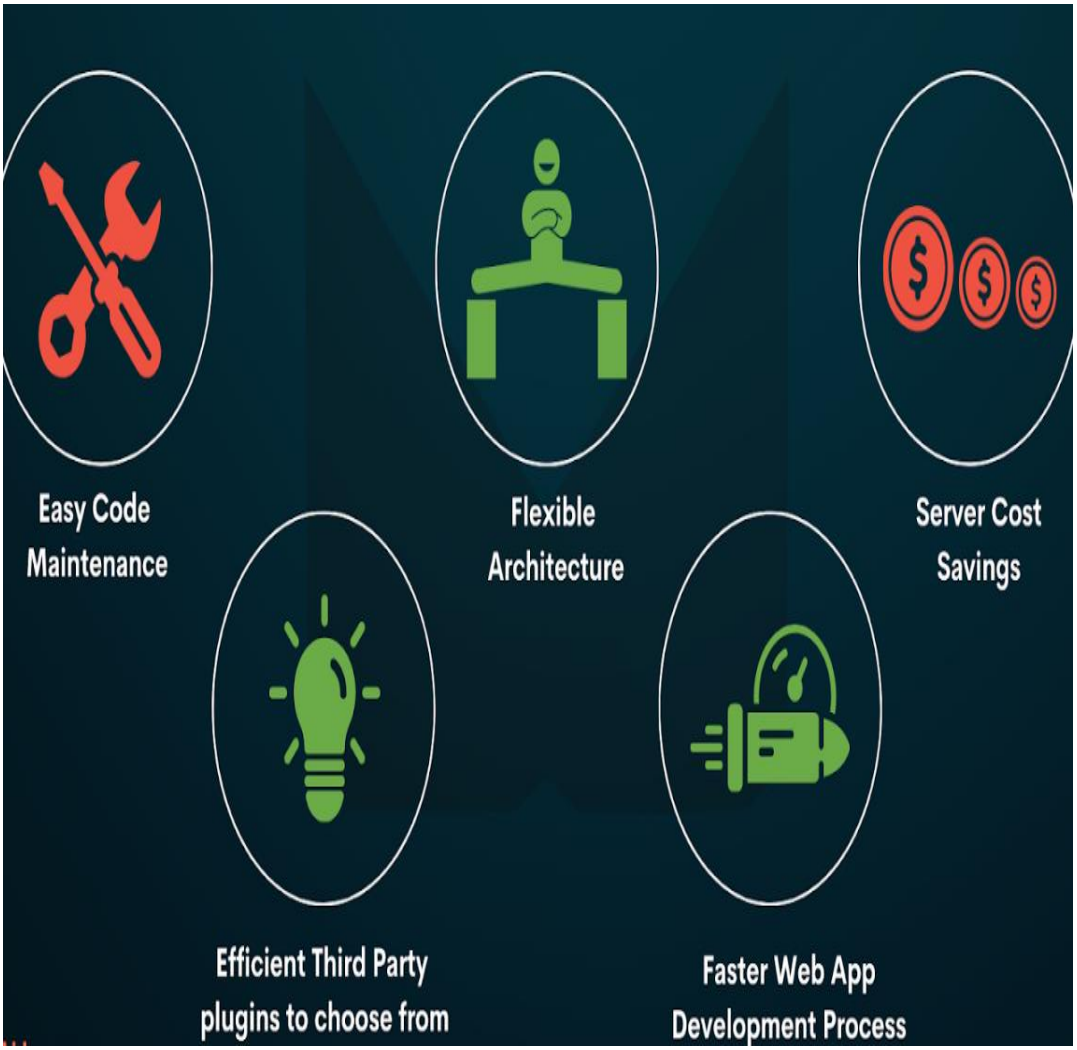


Figure 28 - Picture explaining the benfits of MERN stack - Thorsell-Arntsen, T. (2022).

3.5. Oher Sections

3.5.1 Use case diagram

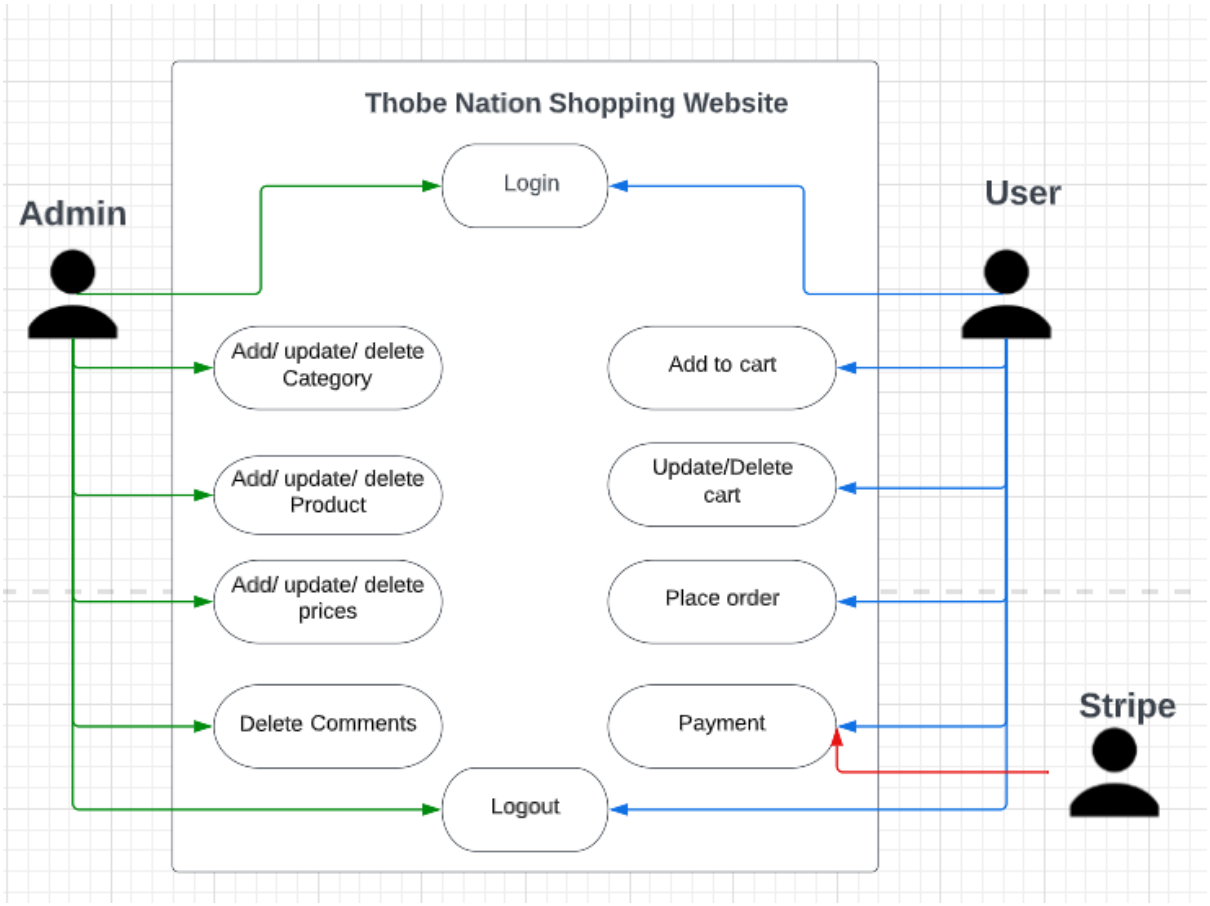


Figure 29 - Use case diagram

The use case diagram above shows in full the functionality interactions that both the admin and the consumer may have with the software. The log-in process is the single instance of a user and administrator interacting together. They then split out to follow their own paths. The user may explore the website and find goods they want using a crud-style user interface (UI), after which they can put the items to their shopping basket. They then have the option of updating or deleting their cart. They may then place the order and pay for their purchases.

Stripe then collects the payments after the purchase has been made.

Only the administrator has access to the capability; they may create, remove, edit, and delete categories and goods in a CRUD manner.

3.7 Stripe Integration Diagram

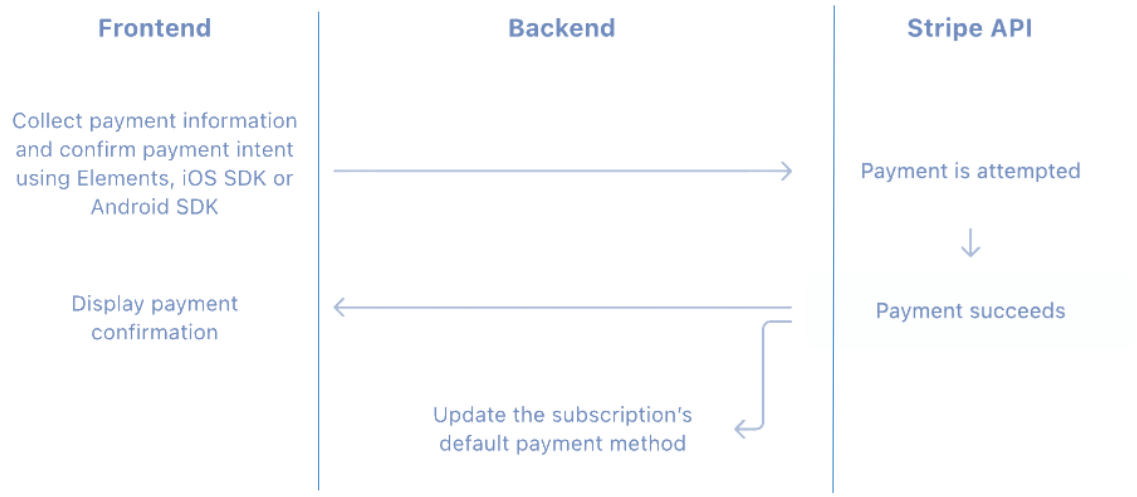


Figure 29 - Stripe Integration Diagram

The consumer enters their card details either in person or online. When those card numbers are entered, Stripe's payment gateway encrypts the information. This information is sent by Stripe to the acquirer, which is the bank handling the transaction on the merchant's behalf. If the payment is successful, a notification is sent back to the front-end.

3.8. Conclusions

The method used for this project has been explored and is presented in this chapter. compared several software methodologies. Then, after weighing the advantages and disadvantages of each, the best option for this project was chosen. The system overview that followed covered both low-fidelity and high-fidelity wireframes. The three-tier system architecture was also looked at. discussed a use case diagram as well.

4. Experiment Development

4.1. Introduction

The creation of the main system components will be explained in this chapter. First a brief description of the Database, server and client setup will be described. Thereafter, individual features will be explained.

4.2. Software Development

4.2.1 Server

Figure 30 - Code for express webserver

This code sets up an Express web server with a few middleware functions and routes. Once everything is satisfactory it then starts the server and listens for incoming requests on the specified PORT.

```
index.js
Ecom-be > src > routes > api > index.js > ...
1   const express = require("express");
2   //requiring the authentication routes
3   const authRoutes = require("./authRoutes");
4   //requiring the adminRoutes routes
5   const adminRoutes = require("./adminRoutes");
6   //requiring the reviewRoutes routes
7   const reviewRoutes = require("./reviewRoutes");
8   //requiring the productRoutes routes
9   const productRoutes = require("./productRoutes");
10
11  let router = express.Router();
12  // all the routes in one place
13  router.use("/auth", authRoutes);
14  router.use("/admin", adminRoutes);
15  router.use("/review", reviewRoutes);
16  router.use("/product", productRoutes);
17
18  module.exports = router;
19
```

Figure 31 - Server routes

This code is used to create a router object that handles requests to different endpoints in a Node.js Express application by delegating those requests to sub-routers defined in separate route files.

Imports four different route files (authRoutes, adminRoutes, reviewRoutes, and productRoutes) that are responsible for handling requests to specific endpoints related to user authentication, administration, reviews, and products.

The router.use() function is then used to define the routes for each of the imported route files. The use() function specifies the endpoint path to handle, and passes in the router object that was defined in the corresponding route file. This allows the main router object to delegate requests to the appropriate sub-router based on the endpoint path.

Inspired from freeCodeCamp.org (2021).

4.2.2 Client

```
1  // importin axios to sending API requests to the backend
2  import axios from "axios";
3
4  //
5  //API version 1 is hosted on the local server at port 5000.
6  const api = axios.create({
7      baseURL: "http://localhost:5000/api/v1",
8  });
9
10 export default api;
11
```

Figure 32 - API

JavaScript library Axios is used to send HTTP requests from Node.js servers or web browsers. In order to communicate with a server-side API

```
import React from "react";
// imporitng the packages from the react-router dom for mangae rou
import { BrowserRouter, Routes, Route } from "react-router-dom";
// importing all our app routes here
import WebRoutes from "./webRoutes";

export default function Index() {
  return (
    // wraping all the routes in BrowserRouter tag so that when we
    <BrowserRouter>
      <Routes>
        {/* all the request containg '/' will goues to webRoutes
        <Route exact path="/*" element={<WebRoutes />} />
      </Routes>
    </BrowserRouter>
  );
}
```

Figure 33 - Client routes

Setting up the main routing for the website. The BrowserRouter, Routes, and Route components from react-router-dom. Then, it imports the WebRoutes component from another file, which contains all the website's routes.

The Routes component is used to define the different routes of the website. The only route defined in this code has a path of "/*" which means that all requests with a path containing a forward slash will go to the WebRoutes component.

Inspired from DEV Community. (n.d.).

4.2.3 Database

```
JS user.js M    JS connection.js ×
Ecom-be > src > db > JS connection.js > ...
1   // Requiring the mongoose package to develop t
2   const mongoose = require("mongoose");
3   // conneting to the mongoDB
4   mongoose
5     .connect(process.env.DB_URL, {
6       useNewUrlParser: true,
7       useUnifiedTopology: true,
8     })
9     .then(() => {
10      // after connecting showing this message i
11      console.log("DB successfully connected!");
12    })
13    .catch((err) => {
14      console.log(err);
15    });
```

Figure 34 - Mongoose connection

Connecting to a MongoDB database using Mongoose, using the connect() method provided by Mongoose. It takes the DB_URL environment variable as a parameter, which contains the URL for the MongoDB instance to connect to. The options object { useNewUrlParser: true, useUnifiedTopology: true } specifies options for the MongoDB driver used by Mongoose. The useNewUrlParser option is used to parse the connection string and the useUnifiedTopology option is used to opt-in to using the new Server Discovery and Monitoring engine. If the connection is successful, it logs the message "DB successfully connected!" to the console. If there is an error while connecting, it logs the error message to the console.

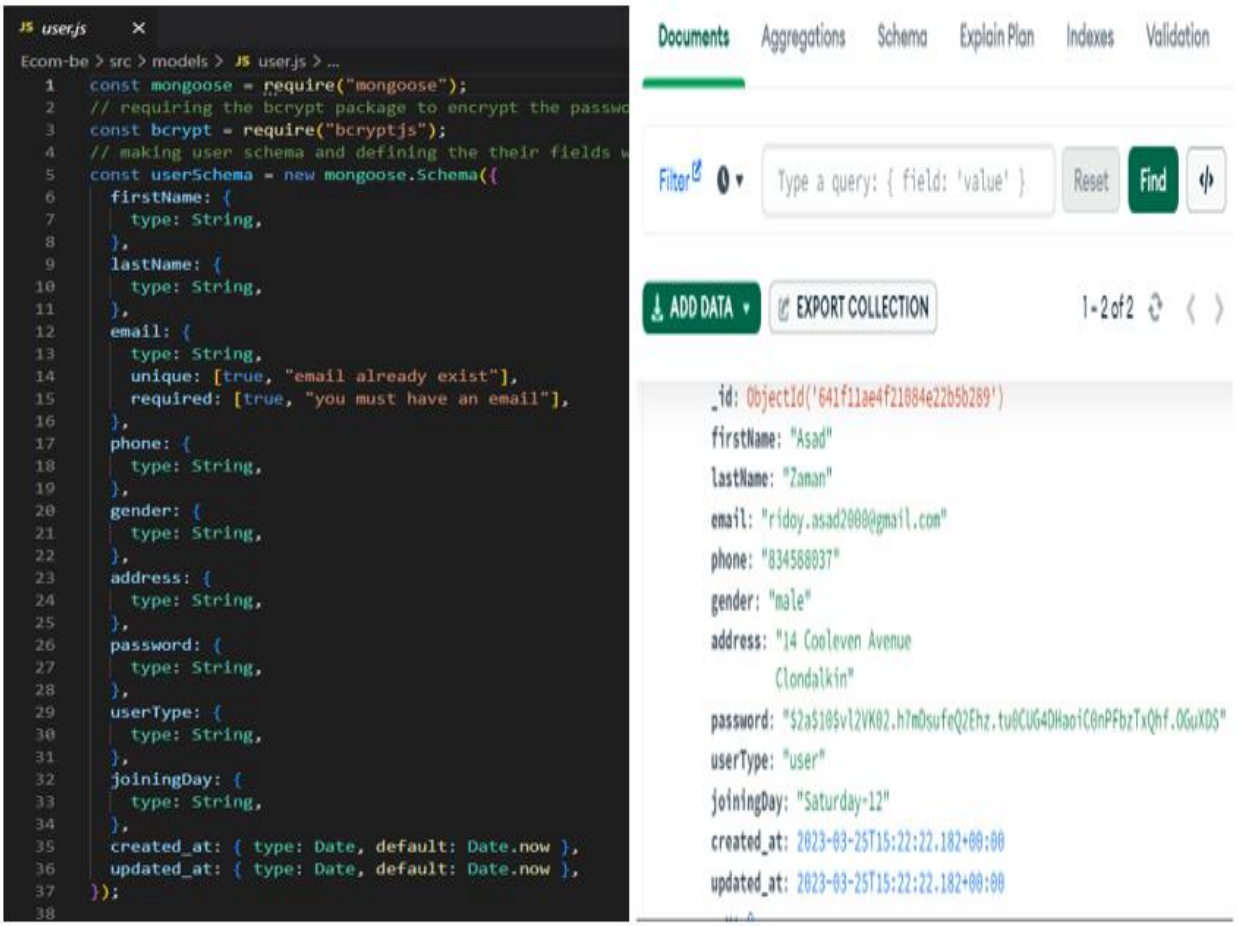


Figure 35 - MongoDb Schema

This is Mongoose schema for a user object, which includes various fields such as first name, last name, email, phone, password, etc. It also defines a pre-save middleware function that hashes the user's password using bcrypt before saving it to the database. Additionally, it defines a method called correctPassword which compares the user's entered password with the hashed password to check for a match.

Inspired form Section. (n.d.).

4.2.4 Sign up

```
// this function will run when user will click on
const onSubmit = async (values, resetForm) => {
  try {
    // it is hitting the api of our backend and g
    values.userType = "user";
    const res = await api.post("/auth/signup", {
      ...values,
      role: "user",
    });
    console.log("res", res);
    // then we are navifating the user to sign pa
    navigate("/signIn");
  } catch (err) {
    console.log("err", err);
  }
  resetForm();
};
```

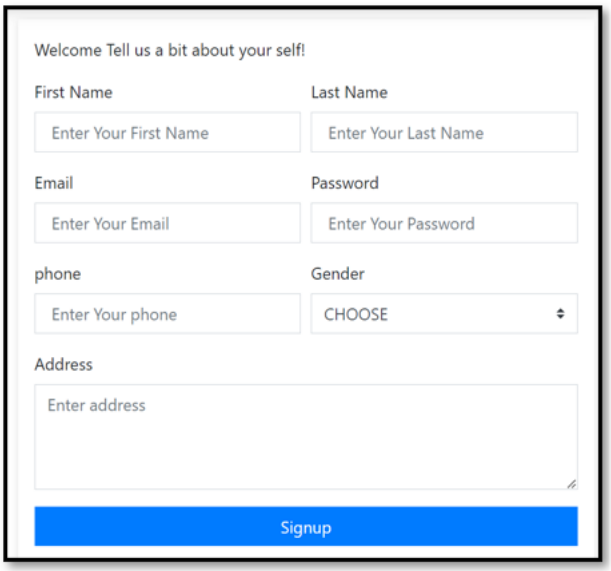


Figure 36 - Sign up

This is function called onSubmit that deals with handling signup form submission asynchronously. This function accepts two parameters when the form is submitted: values, which holds the user-inputted values, and resetForm, which can be used to clear the form fields after submission.

Inspired from bezkoder (2021).

4.2.5 Log in

```
const onSubmit = async (values) => {
  try {
    // it is hitting the api of our backend and
    let res = await api.post("/auth/login", {
      ...values,
    });
    // after geting the response this func is ru
    loginSuccess(res.data.token, res.data.user);
    if (res.data.user.userType == "user") {
      // navigate("/");
      // then we are navifating the user to home
      window.location = "/";
    } else {
      // navigate("/admin/addProduct");
      // if the user will be admin then we naviga
      window.location = "/admin/analytics";
    }
  } catch (err) {
    // when user will enter the wrong email or pa
    toast("Invalid Email or Password!");
    console.log(err);
  }
};
```

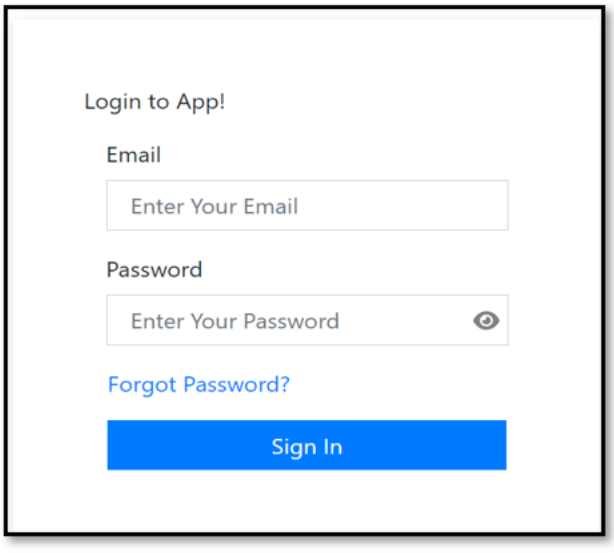


Figure 37 - Log in

When a user submits the sign-in form, a onSubmit function is invoked. To validate the user's email address and password, this method calls the backend using the Axios API. If the user's credentials are

legitimate, the loginSuccess function is called to save the user's information in the app's state, and depending on the user type, the user is either routed to the home page or the admin dashboard. The toast function is used to display an error message and the console reports the error if the user's credentials are incorrect.

4.2.6 Forgot Password

```
const onSubmitForgotPassword = async (values, resetForm) => {
  try {
    // it is hitting the api of our backend and getting the response
    let res = await api.post("/auth/forgotPassword", {
      ...values,
    });
    // sending notification
    toast("Please Check your email for the link!", { type: "success" });
    setForgotPasswordModal(false);
    // empty form after submit
    resetForm();
  } catch (err) {
    toast("Invalid Email", { type: "error" });
    console.log(err);
  }
};
```

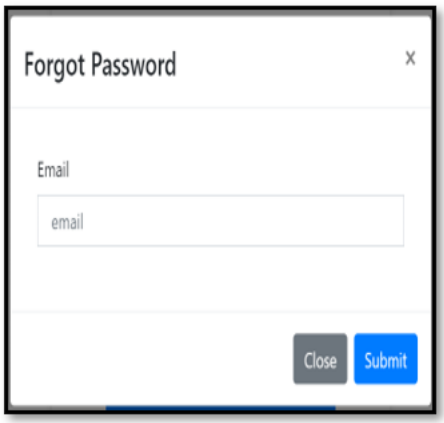


Figure 38 - Forgot password

When a user submits the forgotten password form, the onSubmitForgotPassword function is activated. For the purpose of sending a password reset link to the user's email address, this method calls the backend using the Axios API. The user receives a message telling them to check their email for the reset link after the api.post method, if the email address is genuine, sends the email. The toast function is then used to inform the user that the operation was successful.

4.2.7 Log out

```
const Logout = () => {
  const navigate = useNavigate();
  //   useEffect is removing the user
  useEffect(() => {
    localStorage.removeItem("user");
    localStorage.removeItem("token");
    navigate("/signin");
  }, []);
  return <div></div>;
};
```

Figure 39 - Log out

By utilizing the localStorage.removeItem() method, the "logout" function deletes the user and token information from the browser's localStorage. The user is then sent to the sign-in page using the useNavigate() hook from react-router-dom. When the component is first displayed, the useEffect()

hook is used only once to carry out this operation by giving an empty array as its second parameter. When a user selects the "logout" button, the localStorage is erased and the user is only ever returned to the sign-in page once.

4.2.8 Products

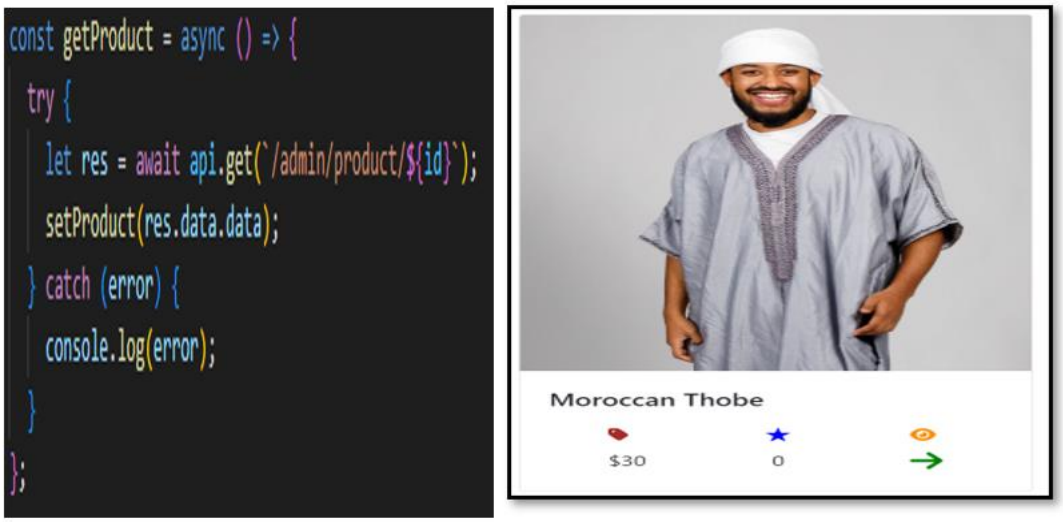


Figure 40 - Single Product

Makes an API call to the server to fetch the product details for the specified id and updates the state variable product with the received data.

Figure 41- Similar products

Brings up items that fall under the same category as the current product. Using a query argument defining the category of the current product, it sends an HTTP GET call to the backend API's /admin/product endpoint using the api object.

Inspired from React.js Examples. (2022).

4.2.9 Personal Information

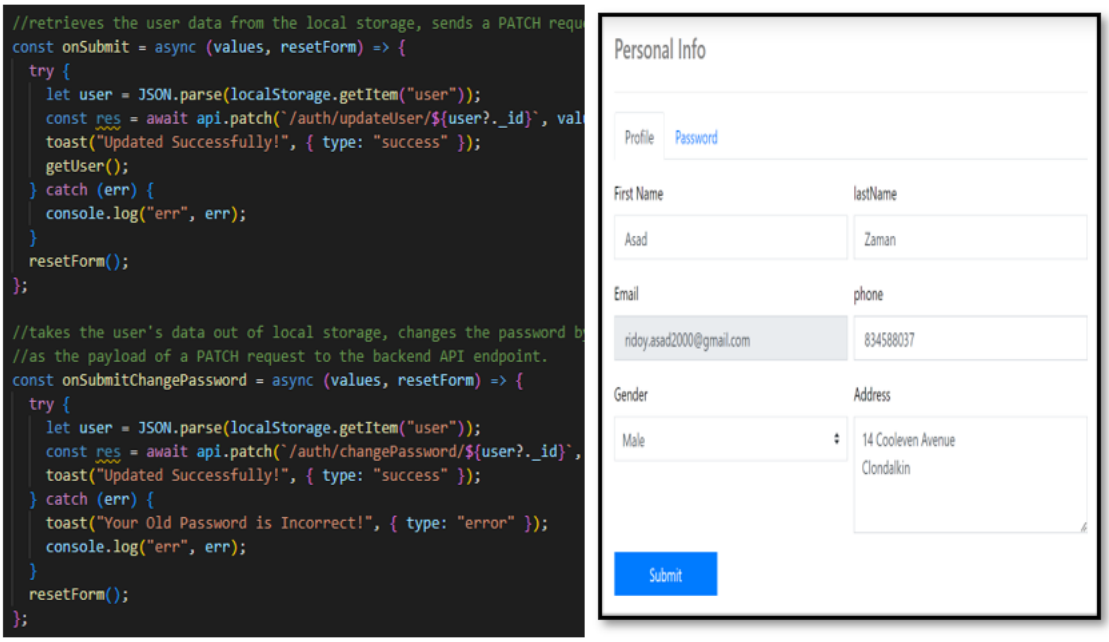


Figure 42 - Personal Information

Allows the user to update their personal information and change their password by retriving the user data from the local storage. Thereafter sending a PATCH request to the server with the updated user information.

4.2.10 Review

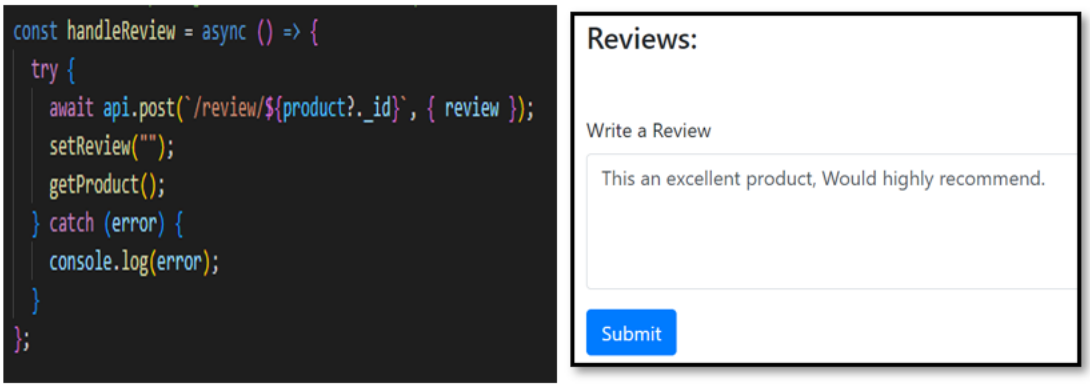


Figure 43 - Reviews

makes a POST request to the server in order to submit a product evaluation. The HTTP request is made using the api object, and the review text and product ID are sent along with it. If the request is

successful, it uses the getProduct method to update the product data with the new review and changes the review status to an empty string.

4.2.11 User products home page

```
// funtionction will run and take all the products from
const getProducts = async () => {
  try {
    let res = await api.get(`/admin/products`);
    setProducts(res.data.data);
    setProductsCopy(res.data.data);
  } catch (error) {
    console.log(error);
  }
};
// this will run first when component will render
useEffect(() => {
  getProducts();
}, []);

// funcion for handling search bar
const filterProducts = () => {
  if (search == "" || !search) {
    setProducts(productsCopy);
  } else {
    setProducts(
      productsCopy?.filter((el) => el?.name?.toLowerCase
    );
  }
};
```

Figure 44 - User products home page

This code provides the method getProducts, which makes an API call to the backend to receive a list of items. The state variables products and productsCopy are used to hold the obtained information. When the component is installed, the getProducts method is called once using the useEffect hook. Another stated function, filterProducts, allows users to filter items using the search term they have typed into the search field.

Inspired from Alhaddad, A. (2022).

4.2.12 Purchased bookings

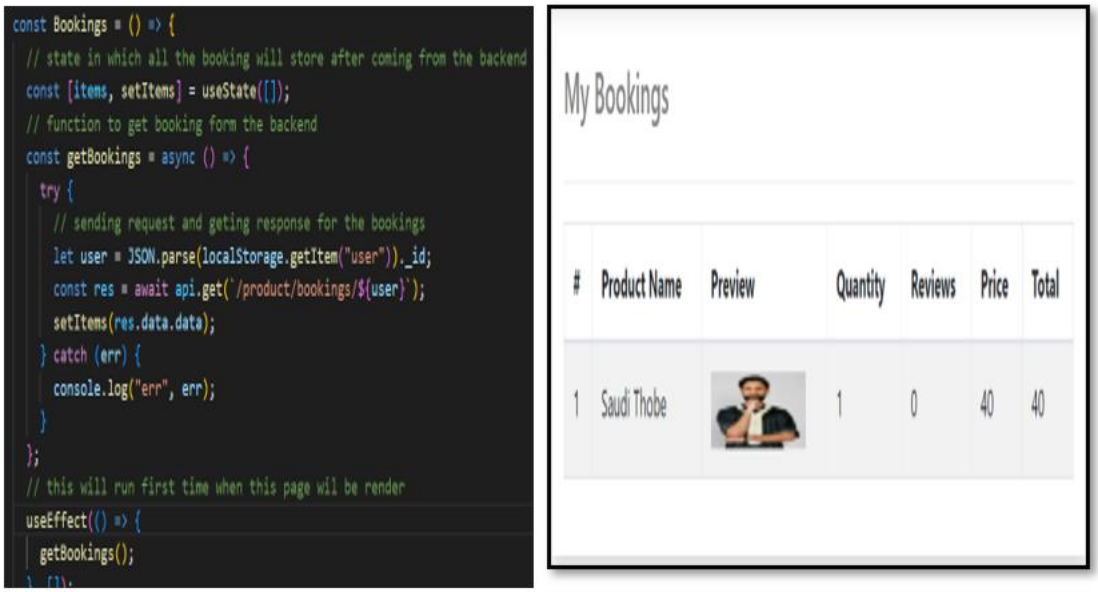


Figure 45 - Purchased booking

Uses an API call to the back end to fetch the booking data. The data is extracted from the response using res.data.data and sets it to the items state variable. This data is displayed on the site.

4.2.13 Admin functionality

Add Product

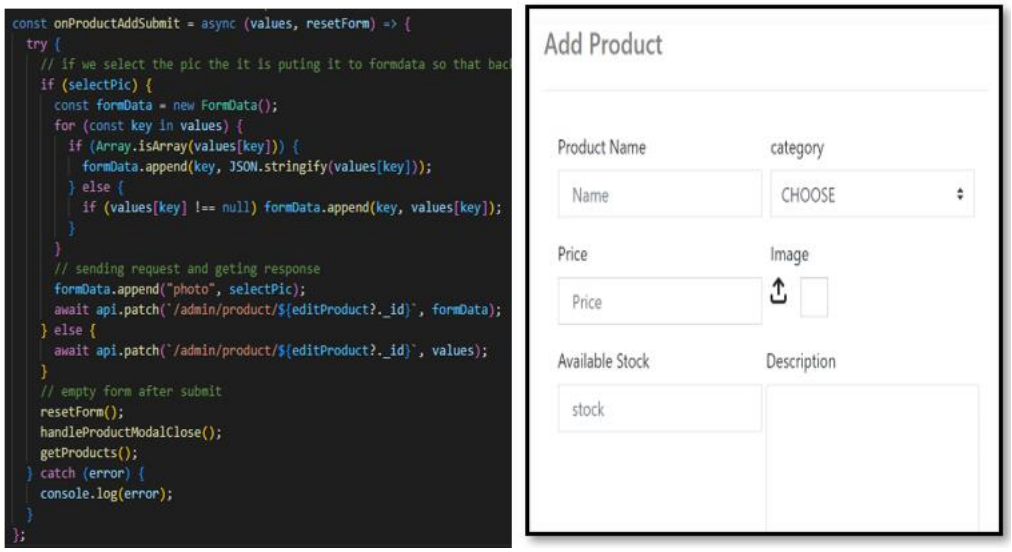


Figure 46 - Add product

Used for handling the submission of a form for adding a product. The method first determines if the user has picked an image file, and if so, generates a new FormData object and populates it with all of the form's variables as well as the selected picture.

After that, it uses api.patch to send a PATCH request with the product ID and the form data to the backend API. If no picture was chosen, the form values are sent in their place.

Delete and Edit Product

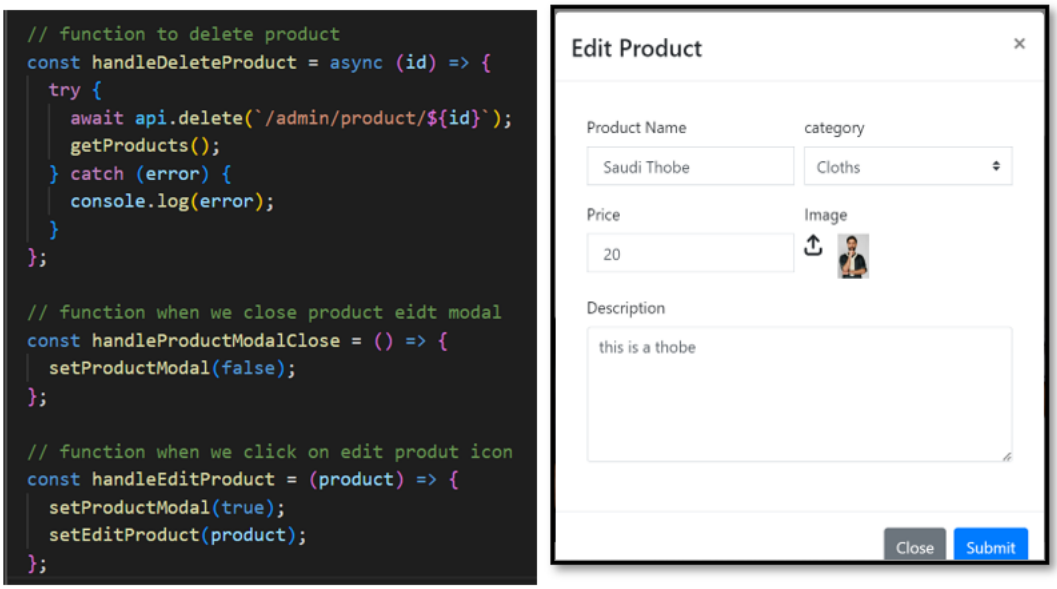


Figure 47 - Delete and edit product

Using an API call to the backend to delete product by making a delete request to the /admin/product endpoint using the api object. The function to edit product sets the value of productModal to true, which opens a modal or dialog box for editing the product.

Product Category

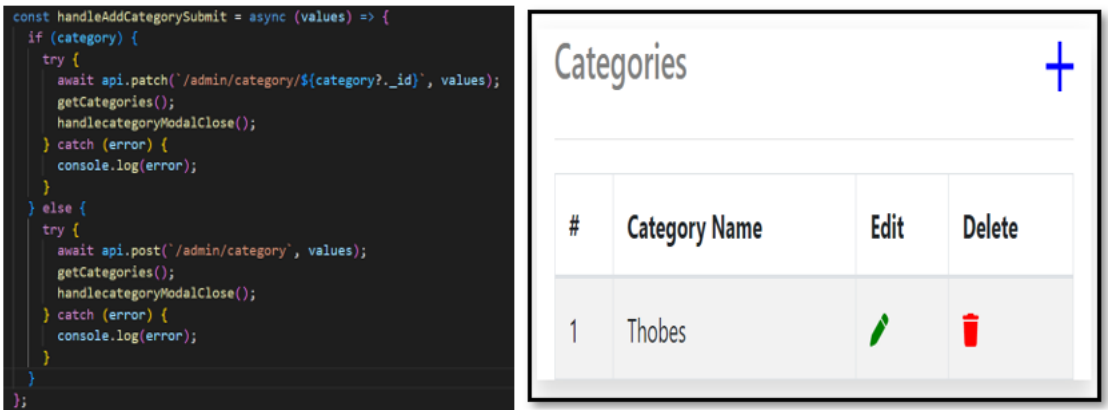


Figure 48 - Product category

The function is used to create and update new category section and sends a PATCH request to the API endpoint to update the category if a category variable is defined. If the PATCH request is successful, the handlecategoryModalClose method is called to shut the modal for adding or modifying categories, and the getCategories function is invoked to update the list of categories.

4.2.14 Analytics

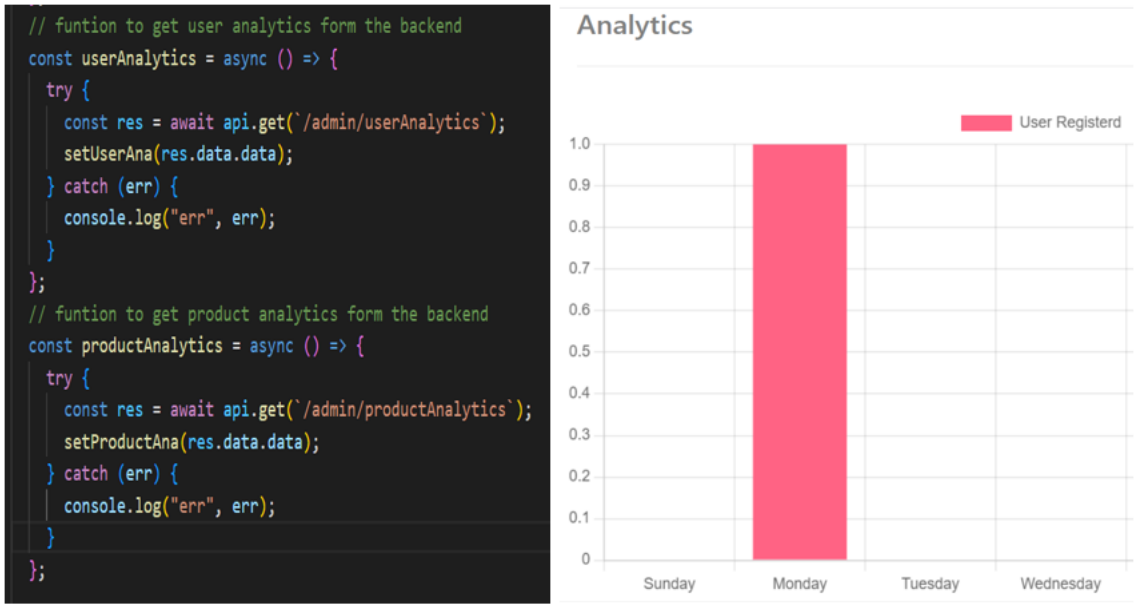


Figure 49 - Analytics

The functions userAnalytics and productAnalytics are in charge of obtaining user and product analytics information from the backend API.

Both methods utilize the api object to call the backend API using the async/await syntax. The requests make use of the API endpoints /admin/userAnalytics and /admin/productAnalytics.

Inspired from Maduabuchi, C. (2022).

4.2.15 Stripe Payment capture

```
let user = await userModel.findById(req.params.userId);
// stripe creating their checkout form
const session = await stripe.checkout.sessions.create({
  // declaring payment method
  payment_method_types: ["card"],
  // declaring currency and total products prices
  line_items: [
    {
      price_data: {
        currency: "eur",
        unit_amount: Math.round(price) * 100,
        product_data: {
          name: title,
          // description: 'Comfortable cotton t-shirt',
          images: [],
        },
      },
      quantity: 1,
    },
  ],
  mode: "payment",
```

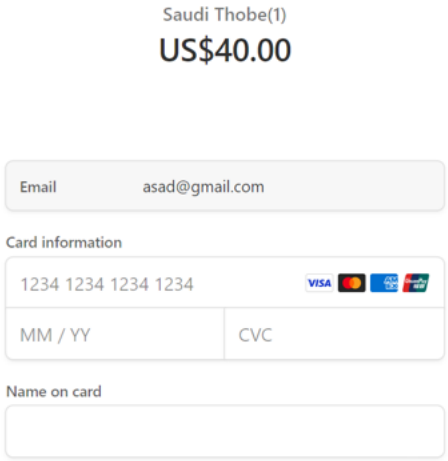


Figure 50 - Stripe form

This code creates a Stripe checkout form for processing payments. The functions help in the processing of items in the cart and calculates the total price. It then creates a checkout session using the Stripe API, providing information about the payment method, currency, product details, success and cancel URLs, customer email, and a client reference ID.

4.2.16 AWS web hosting

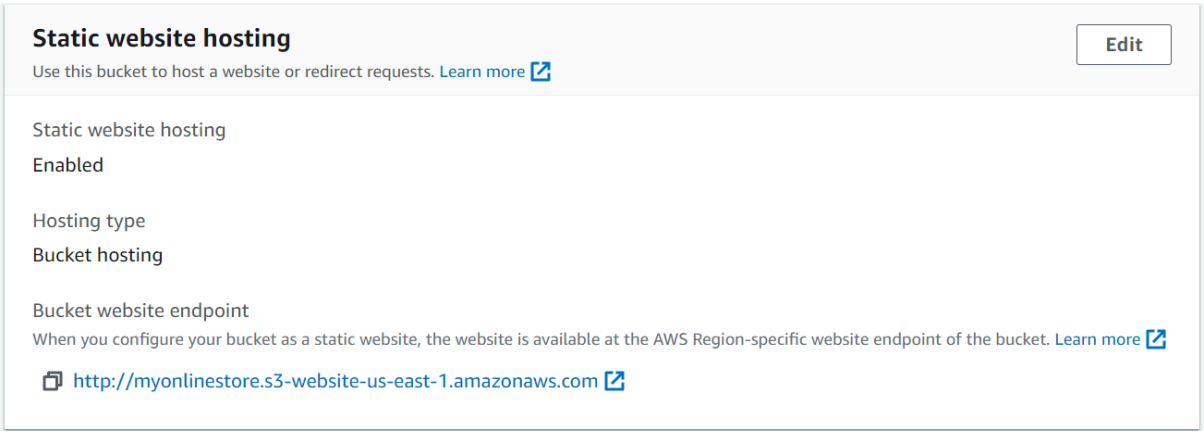


Figure 51 - AWS hosting

The site is being hosted on an S3 bucket as a static website. This can be a cost-effective way to host a site, especially for smaller sites with low traffic volumes.

4.3 Other Sections

4.3.1 Site responsiveness

```css
@media (max-width: 720px) {
  .getIntouchPhone {
    margin-top: 0rem;
    width: 100%;
    display: flex;
    flex-direction: column;
  }
  .getInTouchMain {
    padding: 5rem;
  }
}

@media (max-width: 770px) {
  .section_findSchools {
    display: flex;
    flex-direction: column;
  }
}
```

Figure 52 - Reponsive CSS

Media queries are used to define different styles for different devices based on their screen size. They allow for a responsive design that can adapt to various screen sizes, such as mobile devices and tablets.

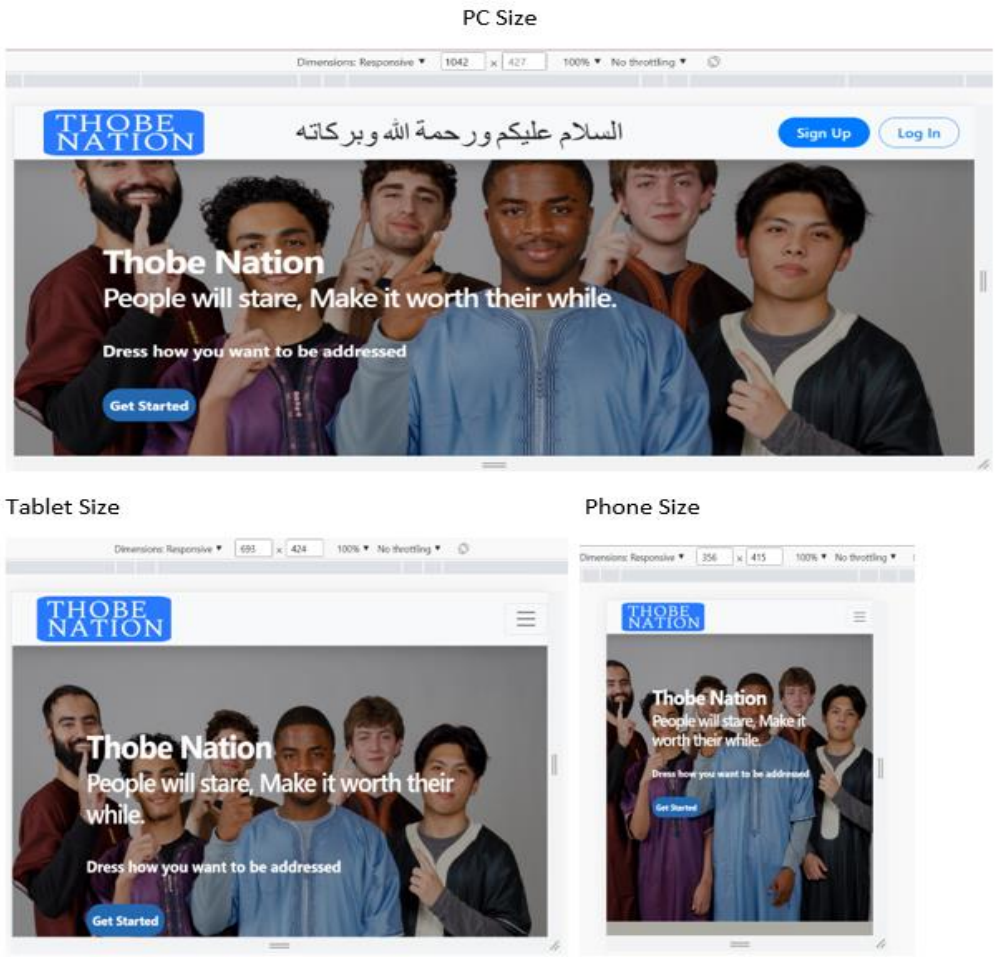


Figure 53 - Responsive sizes

4.3.2 Multer

```
exports.multerUploadS3 = multer({
  storage: multerS3({
    s3: getS3(),
    bucket: "tours-ecommerce",
    acl: "public-read",
    metadata: function (req, file, cb) {
      cb(null, { fieldName: file.fieldname });
    },
    key: function (req, file, cb) {
      cb(
        null,
        Date.now().toString() + "-" + file.originalname.split(" ").join("-")
      );
    },
  }),
});
```

Figure 54 - Multer S3 bucket

This code provides functionality for uploading and processing images. Multer-S3 libraries to handle file uploads to an Amazon Web Services (AWS) S3 bucket. The multerUploadS3 function returns a middleware function that can be used to handle image uploads.

4.3.3 Landing Page

Figure 55 - Landing page

Made using a bootstrap template which was then customized to suite the needs of my project.

4.4 Conclusions

In conclusion, this chapter has provided an overview of the main system components. By providing a detailed description of each component, one can have a better understanding of how the website functions. Additionally, the chapter has highlighted individual features that are essential to the website's operation.

5. Testing and Evaluation

5.1. Introduction

In this chapter there will be an examination of black box testing, usability testing, and mobile device testing to ensure that the website is optimized for functionality, usability. A website must be tested to ensure dependability, security, user experience, and revenue generation. Testing can improve user experience by making sure that the website is simple to use and gives users a satisfying experience. Customers can identify items, complete transactions, and receive the help they require with ease if user interfaces, and navigation are tested.

5.2. System Testing

5.2.1 Black box testing

Black box testing is a software testing approach that focuses on assessing a system's or application's functioning without taking into account the internal organization, design, or implementation specifics. In other words, the tester evaluates the software's input and output behaviours as a "black box" without being aware of how the code executes or what it does on the inside.

Customer side black box testing

Test Case	Expectation	Result
Click logo	Directs to home Page	Pass
Click “get started” button	Redirects to signup form	Pass
Click Register button	Redirects to sign in page	Pass
Click sign in without filling in parameters	Error message	Pass
Validation tests for some forms	User enters certain characters and page does not load.	Failed
Click sing in after filling in parameters	Directs to products page	Pass
Hover over product	Activates a zoom animation	Pass
Click product	Directs to product description page	Pass

Write review then click submit	Displayed a review	Pass
Select quantity then press add to cart	Adds product to cart	Pass
Click cart icon	Directs to cart-items page	Pass
Click check-out	Directs to payment page	Pass
Filled in card details then click pay	Directs to bookings page after payment successfully taken	Pass
Click logout button	User is logged out	Pass

Admin side black box testing

Test Case	Expectation	Result
Click add products button	Directs to add products page	pass
Pressing submit after completing add product form.	Opens device folder to choose from images	pass
Once images are chosen and press submit again.	Adds product to customer side.	pass
Click categories button in menu	Directs to categories page	pass
Click + button in categories	Adds new category	pass
Click edit button in categories	edit category	pass
Click delete button in categories	Deletes category	pass

Click products button	Directs to products page	pass
Click edit button	Edit product	pass
Click delete button	Deletes product	pass
Click analytics button	Directs to analytics page	pass
Customer sign in	User registered is displayed in graph	pass
Customer buys product	User sales is displayed in graph	pass

5.2.2 Website Speed and Performance Test

It is crucial that the website functions effectively, giving visitors a quick and dependable experience. Performance and speed tests should be carried out on a regular basis to make sure a website complies with these requirements. GTmetrixs will be to test the site.

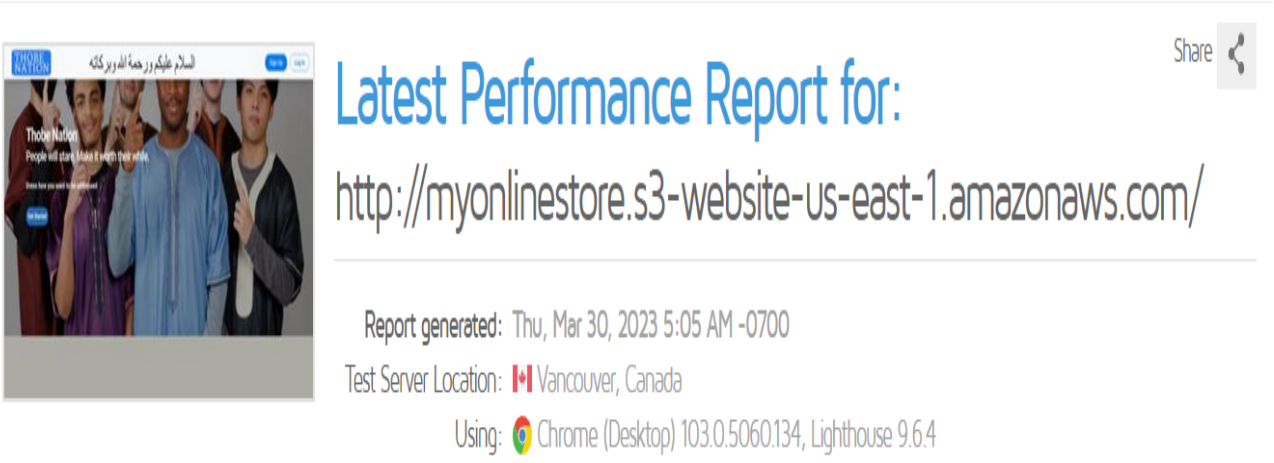


Figure 56 - Performance report

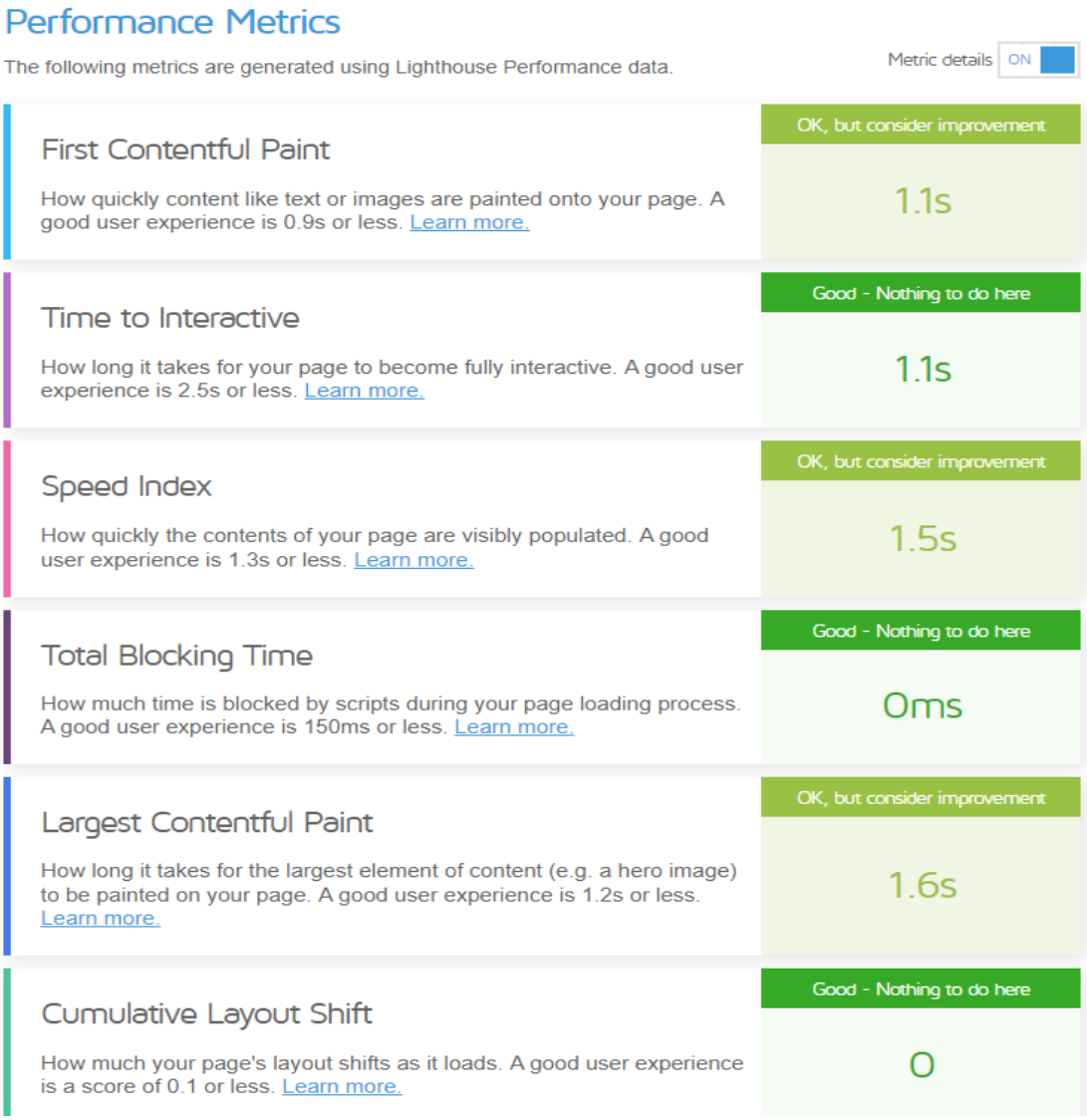


Figure 57 - Performance metrics

As shown by the test results there is room for minor improvements but over the website seems be running as a good speed and has sufficient performance.

5.2.3 Usability Test

This usability testing survey was created using google forms to get user input on how they found the website. Users were questioned about several parts of their experience, including how easy it was to browse items, utilize the website, and check out. This data is necessary to enhance user experience on the website and to spot any problems that may be confusing or frustrating visitors.

Figure 58 - Usability report 1

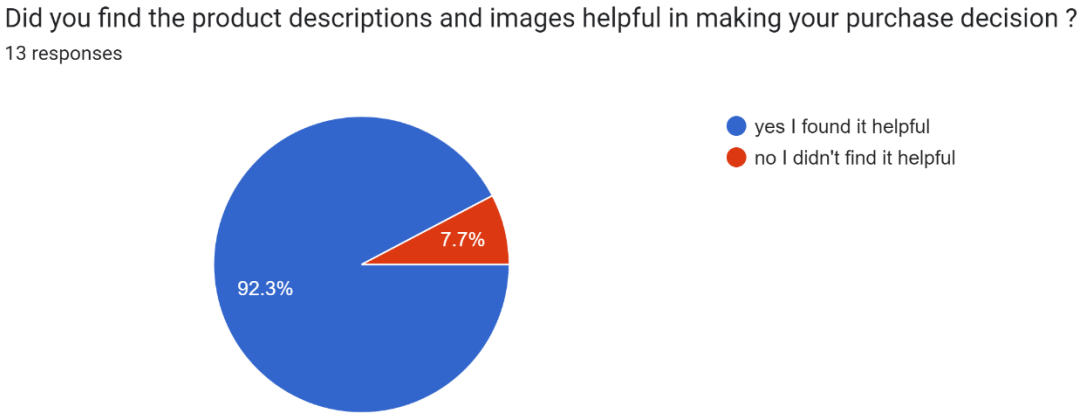


Figure 59 - Usability report 2

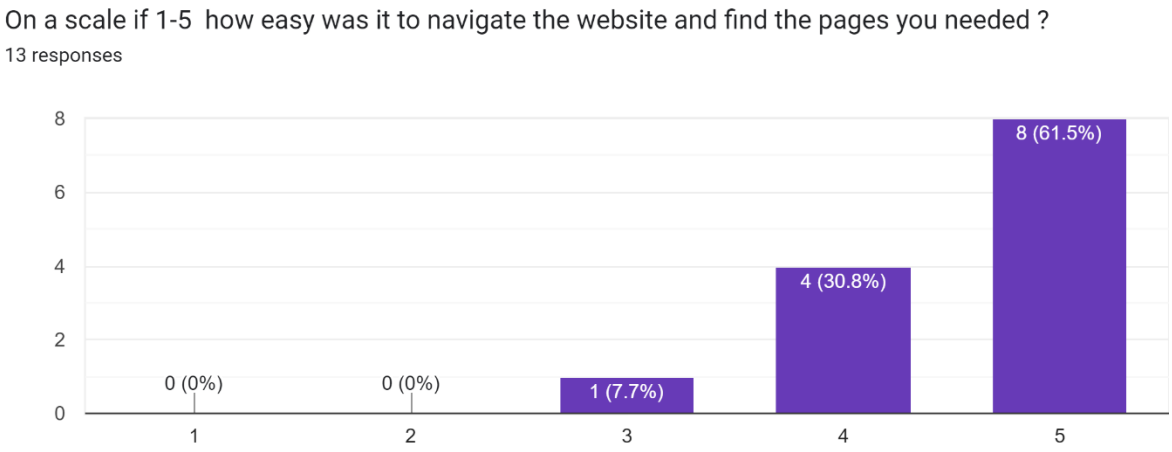


Figure 60 - Usability report 3

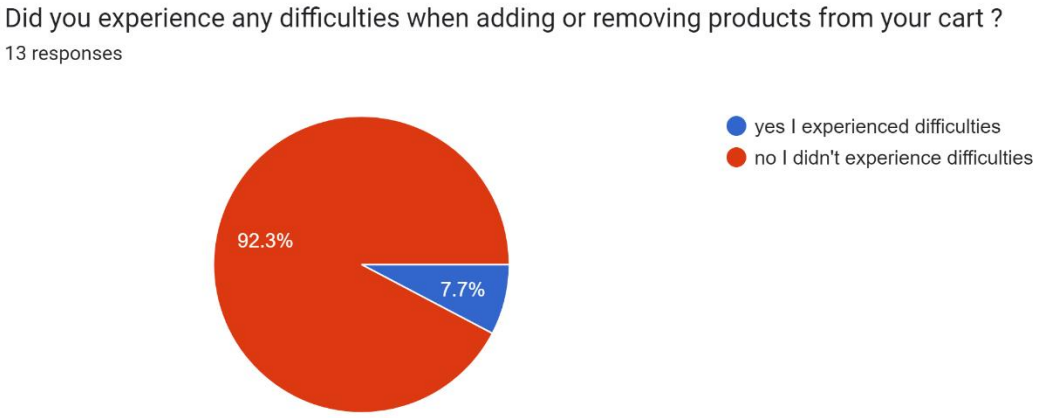


Figure 61 - Usability report 4

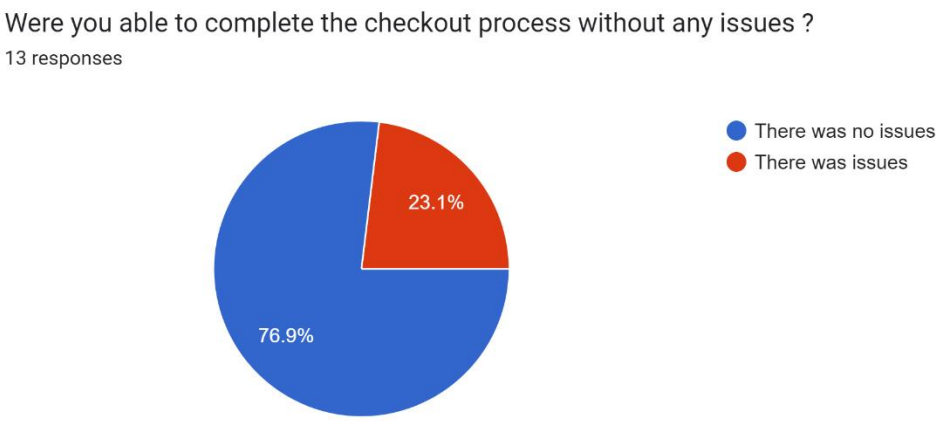


Figure 62 - Usability report 5

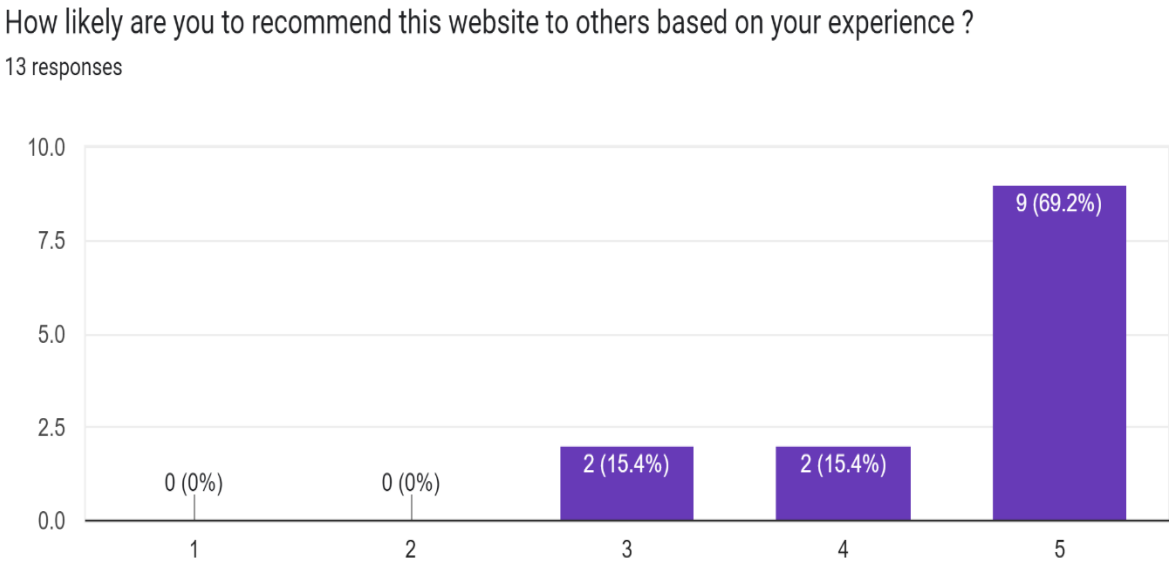


Figure 63 - Usability report 6

5.2.4 Mobile Testing

Most of the users of this site is going to use mobile devices therefore, mobile device testing is necessary to ensure that an application works well on different devices and provides a good user experience. Without mobile device testing, there is a risk that the application will not work as expected and that users may have a poor experience.

Mobile Tests

Test Case	Result
User can launch website using link	pass
User can log in and explore the screens	pass
When disrupted by notification website still works	pass
All GUI component is in correct position	Fail some button CSS need to be rearranged

Mobile device user interface

This test will make sure that all buttons and texts, pictures and other content in displayed properly in both the user and admin side. This test is conducted by opening a page on the mobile platform and changing any content in CSS until it is satisfactory. The satisfactory Graphical user interface of the mobile site will be displayed below.

5.2.4.1 Mobile user side

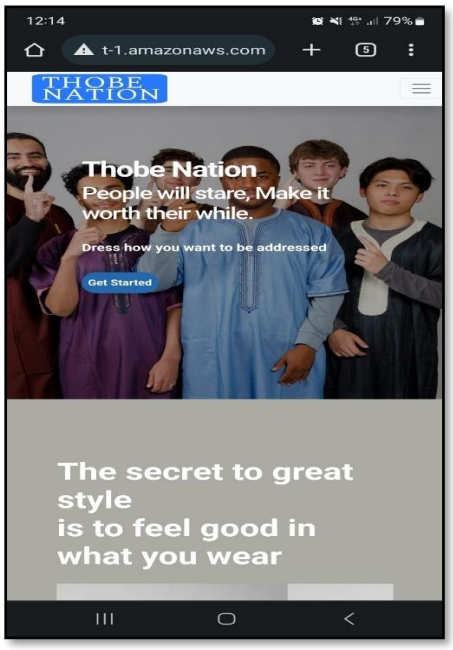


Figure 64 - Mobile landing page

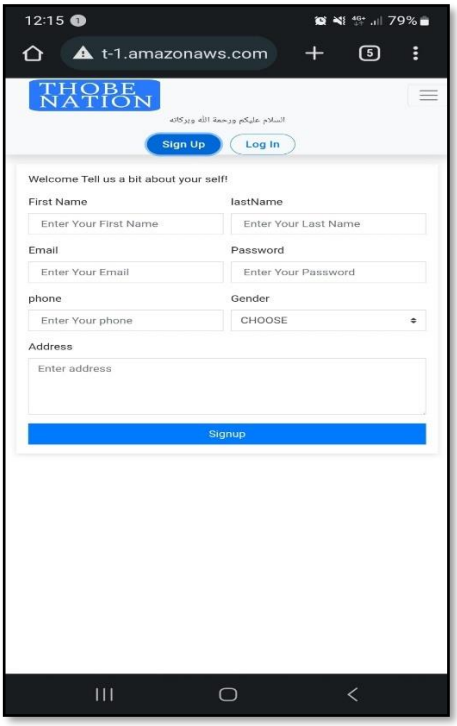


Figure 65 - Mobile sign up

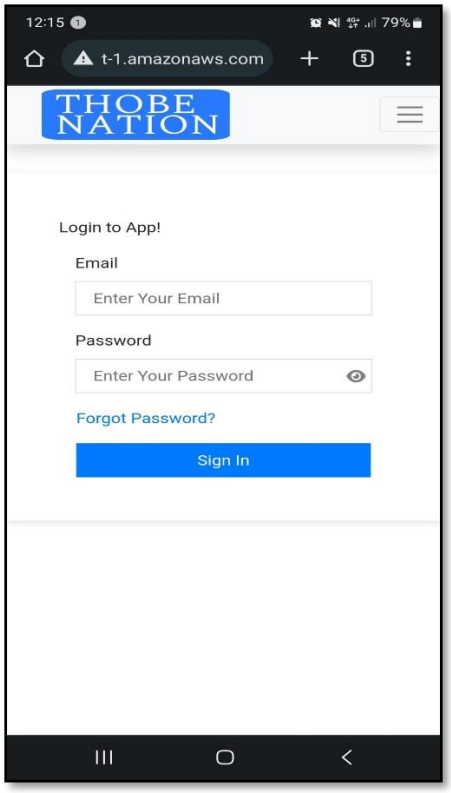


Figure 66 - Mobile Login

Figure 67 - Mobile product details page

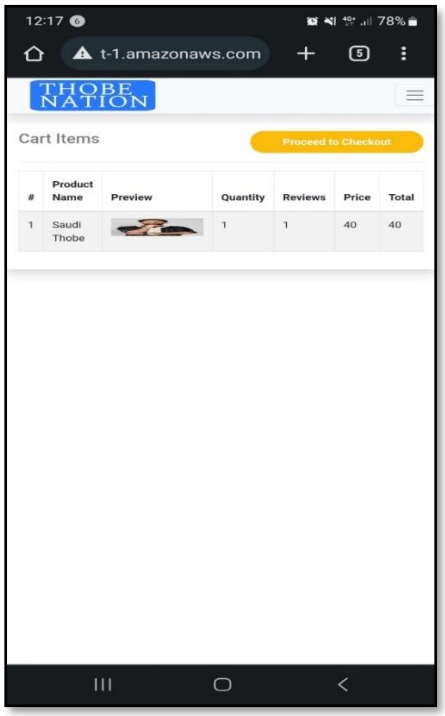


Figure 68 - Mobile cart items

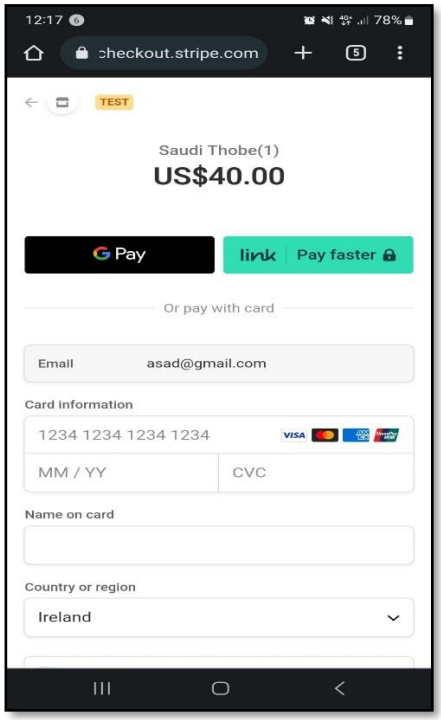


Figure 69 - Mobile payment

5.2.4.2 Mobile admin side

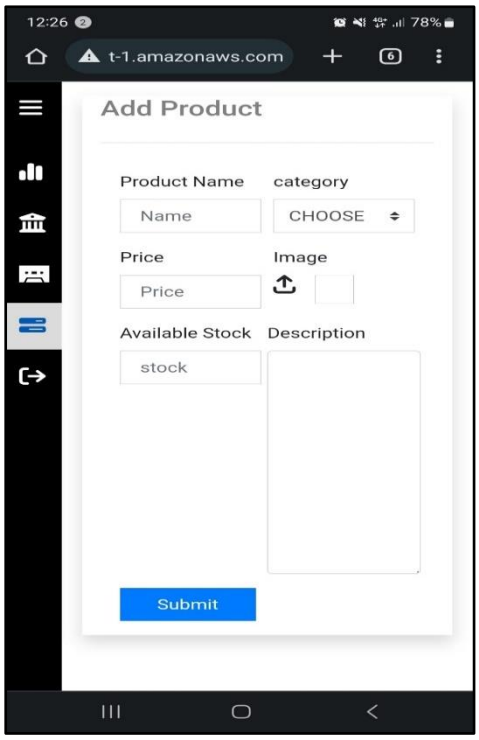


Figure 70 - Mobile add product

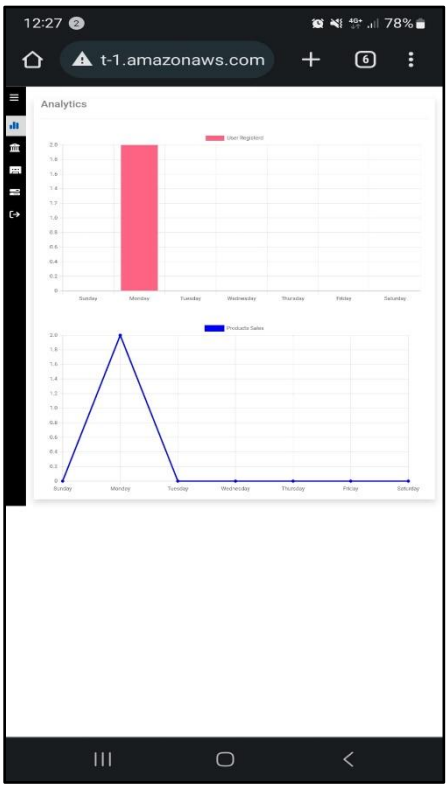


Figure 71 - Mobile Analytics

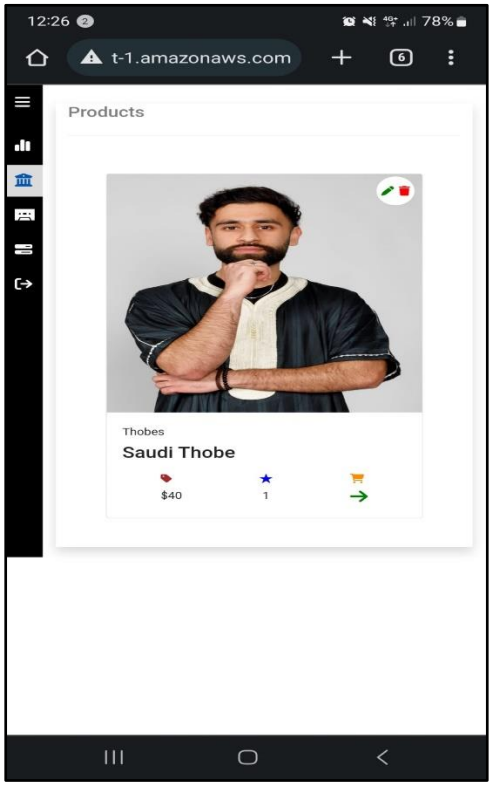


Figure 72 - Mobile product edit

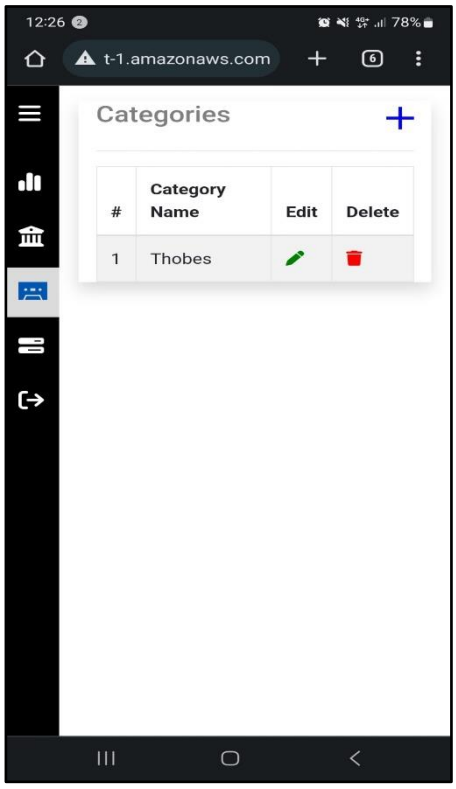


Figure 73 - Mobile categories

5.3. System Evaluation

The black box testing process identified and resolved many functionality issues, ensuring that the website is functioning as expected. The shopping cart and checkout process are working smoothly, and the website is loading quickly, providing users with a seamless and enjoyable experience. The black box testing also highlighted that the website is user-friendly and easy to navigate, making it easy for users to find the products they need. There are still a few bugs to fix but overall, it was a beneficial testing method.

Furthermore, the feedback gathered during usability testing will be used to further refine and improve the website, ensuring that it meets the needs of users and provides a high-quality platform.

Moreover, the mobile testing process has ensured that the website is optimized for different screen sizes and resolutions, and that it works well on different mobile devices and operating systems. The website's user interface and user experience have been tested, ensuring that users can easily navigate and use the website on their mobile devices.

5.3.1 Admin Evaluation

In this section there will be a feature of the admins evaluation and questions he asked when he used the site.

Admin Questions

Admin Questions	Developer Answers
How did you make sure the website is secure?	In the back-end, JWT tokens were utilized to create a Sign-in function, which provides a secure method for authorizing users and protecting confidential information. Additionally, protected routes were implemented in the front-end to ensure that only authenticated users are granted access to certain features or data.
What payment method are accepted on the website?	The site is utilizing Stripe for payment capture, which enables them to accept payments via debit cards, credit cards, Google Pay, and Apple Pay.
	There is contact information in the bottom of

How is customer service handled?	the landing page for customers to use. However, in the future there can be a dedicated form for customer queries.
Is the website optimised for mobile devices?	The website is mobile responsive. There are a few GUI changes that will be implemented before deployment.
Are there any analytics tools available?	In the admin panel once the user signs in or a customer makes a purchase it will be displayed in the analytics section.

Admin Feedback:

"The Website looks very pleasing to the eye, I like the animation when I scroll down. It has a very simplistic yet modern design. Adjust the picture sizes on the product description page other than that I have happy with your progress. I cant wait to see the finished product."

5.4. Conclusions

In conclusion, testing is a crucial component of making sure a website is dependable, secure, user-friendly, and profitable. The website may be optimized for functionality, usability, and mobile devices by doing black box testing, usability testing, and mobile device testing. This will guarantee that users like using the website, can quickly locate what they need, can conduct transactions without difficulty, and can get the help they need when they need it. In the end, testing helps to enhance the entire user experience, which boosts customer happiness and revenue generating.

6. Conclusions and Future Work

6.1. Introduction

In this chapter there will be a final reflection on how the whole project was developed, there will be an explanation about future work that needs to be completed and thereafter a personal reflection on the experience.

6.2. Future Work

There is still work to be done. The plan is to continue developing and refining the website, with a focus on improving the user experience and adding additional functionality which will be displayed below.

- Fix all the faults found in the tests which run such as minor validation and GUI alignment for buttons and pictures.
- Display purchase information on the Admin Panel which be created using back-end API development and frontend component development.
- Make admin able to delete comments from just in case they are inappropriate.
- Buy domain name and integrate it into the URL.

6.4. Conclusion

This project allowed me to pick up so many new skills and meet so many new people. This application was made possible thanks to the extensive knowledge gained throughout the research stage. It took many obstacles, setbacks, and hours that felt squandered throughout the project's development stage until it all just clicked. Giving someone the freedom, flexibility, and opportunity to create not just an application of their choosing but also to employ technology of their choosing is like a blessing and curse because of the vast freedom but also challenging learning curve. Yet there is always a solution if one has the courage and desire. As soon as this application is finished and of a level that can be released, it will have a bright future. The real test of its effectiveness will be if it fulfils its intended goal of providing individuals with access to traditional Islamic clothes.

Bibliography

1. Pires, R. (2020). *What is the Role of Data Visualization in E-commerce?* [online] Prisync. Available at: https://prisync.com/blog/what-is-the-role-of-data-visualization-in-e-commerce/.

2. Murray, C. (2019). *Why do small businesses in Ireland need a website 2019 | Craig Murray*. [online] Why do small businesses in Ireland need a website ? Available at: https://craigmurray.ie/why-do-small-businesses-in-ireland-need-a-website-2019/

3. Panchaud, K. (2021). *Nielsen's 10 usability heuristics & their importance*. [online] Medium. Available at: https://bootcamp.uxdesign.cc/jakobs-10-usability-heuristics-their-importance-6f0ddec8c938

4. Shopify (n.d.). *Luxury Emirati Thobes - Al Jawdah Attire │ Ireland Based*. [online] AlJawdah. Available at: https://aljawdahattire.com/

5. Wordpress (n.d.). *Kurta Pyjama – Libass Boutique*. [online] LibaasBoutique. Available at: https://libaasboutique.com/product-category/men/kurta-pyjama/

6. Annaqah (n.d.). *Annaqah*. [online] Annaqah. Available at: https://annaqah.com

7. Nuraly, A. (2020). *Major advantages and disadvantages of Dart language*. [online] Code Carbon. Available at: https://codecarbon.com/pros-cons-dart-language/

8. Team, D. (2019). *Pros and Cons of JavaScript - Weigh them and Choose wisely!* [online] DataFlair. Available at: https://data-flair.training/blogs/advantages-disadvantages-javascript/

9. AltexSoft. (2016). *The Good and the Bad of TypeScript*. [online] Available at: https://www.altexsoft.com/blog/typescript-pros-and-cons/

10. TechVidvan Team (2019). *Python Advantages and Disadvantages - Step in the right direction - TechVidvan*. [online] TechVidvan. Available at: https://techvidvan.com/tutorials/python-advantages-and-disadvantages/

11. Prasanna (2021). *PHP Advantages and Disadvantages | What is PHP Language? Merits and Demerits of PHP*. [online] A Plus Topper. Available at: https://www.aplustopper.com/php-advantages-and-disadvantages/

12. Editor (2022). *Top JavaScript Framework Comparison in 2023*. [online] wpwebinfotech.com. Available at: https://wpwebinfotech.com/blog/javascript-frameworks/

13. Bhatt, R. (2020). *React JS Examples must do for a Beginner*. [online] Weekly Webtips. Available at: https://medium.com/weekly-webtips/react-js-examples-must-do-for-a-beginner-bb26ca4fe160

14. Express (2020). *node.js - MODULE_NOT_FOUND trying to share code by directory between modules in Node JS / TypeScript projects*. [online] Stack Overflow. Available at: https://stackoverflow.com/questions/66446206/module-not-found-trying-to-share-code-by-directory-between-modules-in-node-js.

15. Arsenault, C. (2017). *The Pros and Cons of 8 Popular Databases - KeyCDN*. [online] KeyCDN. Available at: https://www.keycdn.com/blog/popular-databases

16. MySQL (2021). *MySQL :: MySQL Workbench: Visual Database Design*. [online] www.mysql.com. Available at: https://www.mysql.com/products/workbench/design/

17. PostgreSQL (2017). *PostgreSQL Create Database - javatpoint*. [online] Available at: https://www.javatpoint.com/postgresql-create-database.

18. MongoDB (2022). *Create A MongoDB Database*. [online] MongoDB. Available at: https://www.mongodb.com/basics/create-database.

19. Merchant Maverick. (2020). *How Does Stripe Work? The Beginner's Guide To Stripe*. [online] Available at: https://www.merchantmaverick.com/how-does-stripe-work/.

20. Kcsitglobal (2022). *Amazon Cloud Services | Amazon Web Services - KCS*. [online] Available at: https://www.kcsitglobal.com/solution/cloud/amazon-web-services.

21. Interqualitybg.com. (2019). *AGILE METHODOLOGY*. [online] Available at: https://interqualitybg.com/en/resources/scrum-and-agile-resources/agile-methodology

22. Raycad (2019). *DevOps methodology and process*. [online] Medium. Available at: https://medium.com/@raycad.seedotech/devops-methodology-and-process-dde388eb65bd

23. Indeed Career Guide. (2021). *A Complete Guide to the Waterfall Methodology | Indeed.com*. [online] Available at: https://www.indeed.com/career-advice/career-development/waterfall-methodology.

24. Getbreakout.com. (2018). *Rapid Application Development - 2021 Complete Guide | Breakout*. [online] Available at: https://getbreakout.com/no-code/rapid-application-development/

25. WireFrame (2019). *How to wireframe*. [online] Figma. Available at: https://www.figma.com/blog/how-to-wireframe/.

26. Thorsell-Arntsen, T. (2022). *What is MERN stack*. [online] W3Schools.com. Available at: https://campus.w3schools.com/blogs/blog/what-is-mern-stack.

27. RCDesing (2019). *Responsive Website Design made for PC, Tablet & Mobile devices*. [online] Funky Digital Ltd. Available at: https://funkydigitalagency.co.uk/responsive-web-design

28. GitHub. (2022). *Invalid success message when build task fails · Issue #8467 · microsoft/vscode-cpptools*. [online] Available at: https://github.com/microsoft/vscode-cpptools/issues/8467.

29. BBTesting (2011). *Black Box Testing: An In-depth Tutorial with Examples and Techniques*. [online] Softwaretestinghelp.com. Available at: https://www.softwaretestinghelp.com/black-box-testing

30. Anon, (2021). *How to do a survey on concept testing? -Voxco*. [online] Available at: https://www.voxco.com/blog/how-to-do-a-survey-on-concept-testing/

31. Gtmetrix.com. (n.d.).(2023) *GTmetrix Performance Report*. [online] Available at: https://gtmetrix.com/reports/myonlinestore.s3-website-us-east-1.amazonaws.com/6N3ysGY7/

32. freeCodeCamp.org. (2021). *How to Create a React App with a Node Backend: The Complete Guide*. [online] Available at: https://www.freecodecamp.org/news/how-to-create-a-react-app-with-a-node-backend-the-complete-guide/

33. DEV Community. (n.d.). *How to use Axios with React*. [online] Available at: https://dev.to/femi_dev/how-to-use-axios-with-react-5fom.

34. Section. (n.d.). *How to Connect MongoDB to Node.js Using Mongoose*. [online] Available at: https://www.section.io/engineering-education/nodejs-mongoosejs-mongodb/.

35. bezkoder (2021). *React.js Login & Registration example - JWT & HttpOnly Cookie*. [online] BezKoder. Available at: https://www.bezkoder.com/react-login-example-jwt-hooks/.

36. React.js Examples. (2022). *E-commerce website using the MERN Stack*. [online] Available at: https://reactjsexample.com/e-commerce-website-using-the-mern-stack/.

37. Alhaddad, A. (2022). *Build a E-Commerce site using MERN Stack! STEP 5A: Setup Basic FrontEnd, using React*. [online] Medium. Available at: https://alialhaddad.medium.com/build-a-e-commerce-site-using-mern-stack-step-5a-setup-basic-frontend-using-react-2a0b43c76150/

38. Maduabuchi, C. (2022). *Using Chart.js in React*. [online] LogRocket Blog. Available at: https://blog.logrocket.com/using-chart-js-react/.