Responsive Web Design for Enhanced User Experience (UX) and User Interface (UI)

Victor Aienobe
School of Computing, Engineering and Digital Technologies
Department of Computing and Games
Teesside University
Middlesbrough TS1 3BA

Muhammad Zahid Iqbal, PhD
Supervisor

ABSTRACT
This research focuses on enhancing user experience and interface within an e-commerce platform through responsive web design and the integration of an AI-powered chatbot. The research is to investigate methods for content and layout optimisation across a range of screen sizes, measure the influence of AI-driven solutions in improving user interactions, and analyse the impact of responsive design implementation on metrics related to user engagement.

React Vite was used in the project's frontend development, allowing for the creation of a flexible and visually appealing user experience. The backend was built with Node.js/Express, which provides stability for handling server-side operations. The database used was MongoDB, which facilitates effective data storing and retrieval.

The research methodology incorporated participant feedback collected through surveys and user interactions with the e-commerce platform. Results from the survey showed that responsive design and increased user engagement are strongly associated. Respondents lauded the aesthetic appeal and visual coherence that were preserved across a range of devices, highlighting the significance of responsive design concepts. Limitations in technical integration and resource constraints were encountered, which shaped recommendations for future improvements, such as expanding the chatbot's functionality to include external links and voice prompts.

This research proves the potential of AI chatbots to improve navigation in e-commerce applications and emphasises the importance of flexible web design in improving user experiences. To further enhance user interactions, future development will involve expanding chatbot capabilities and improving content layout tactics.

Keywords
Responsive Web Design, AI Chatbots, User Experience(UX), Frontend and Backend Development

1. INTRODUCTION

1.1 Background study
Consumer engagement with online platforms has undergone a paradigm shift because of the constantly changing digital world. Particularly, e-commerce has witnessed significant growth in recent years as more and more customers depend on the ease of online buying. However, the rise in online shopping has brought about new difficulties for application designers and developers, making the delivery of a seamless and user-friendly experience across various devices one of the most pressing challenges they face (Herman & Umi, 2022).

In today's digital age, users can access e-commerce websites from a diverse array of gadgets, including desktop computers, laptops, smartphones, and tablets. The task of ensuring a consistent and optimized user experience is significantly challenged by the wide range of screen sizes, resolutions, and capabilities among these devices. One key strategy that has emerged to address this issue is responsive web design (Ezwan et al., 2015), a transformative approach that allows web applications to adapt dynamically in response to different device characteristics. Thanks to responsive web design, a single web page can be displayed differently depending on the screen size of the device accessing it. This adaptability may lead to adjustments in the layout of web content, potentially affecting the amount of information displayed on the screen, ultimately influencing the efficiency and effectiveness of information delivery on a web page. Although there are drawbacks, responsive web design is regarded as the industry standard for contemporary web design, guaranteeing improved user experiences and device adaptability (Mohorovicic, 2013).

The importance of this project lies in its endeavour to harness the concepts of responsive web design and integration with artificial intelligence (AI) in order to enhance the user experience of an e-commerce web application. Beyond just tackling technological challenges, this project also delves into critical aspects of user-centred design, accessibility, and performance optimization, ensuring that the resulting e-commerce platform not only adapts to various devices but also prioritizes the needs and experiences of its users.

1.2 Research Questions
The following research questions would be addressed in this study:

- Does implementing responsive web design influence user engagement?
- How effective is the use of AI in enhancing user experience?
- What strategies/design patterns can be employed to optimize content and layout for different screens?

1.3 Aim and Objectives
This project's main goal is to apply responsive web design concepts to improve the user experience and interface of an e-commerce web application. The goal of the project is to build a highly functional, adaptive, and accessible e-commerce platform that can serve consumers on a variety of devices and provide a seamless shopping experience.

To achieve this, this project’s objectives are as follows:

- Develop a responsive e-commerce web application that can adapt seamlessly to a variety of devices such as tablets, laptops, smartphones, and desktop computers.
- Use performance optimisation techniques to make the e-commerce platform faster and more effective, providing users with seamless interactions and quick loading times.
- To evaluate the success of responsive design and user experience improvements through usability testing.

2. LITERATURE REVIEW

This chapter will provide insight into existing theories and concepts in the area of responsive web design, highlighting their methodologies and results gathered.

2.1 Definitions and Related Concepts

- **User Experience (UX):** User Experience, or UX for short, is the term used to describe the complete emotional and functional aspect of the user's engagement with a system, service, or product; it is usually used in reference to digital technology. It includes many different aspects such as performance, accessibility, aesthetics, usability, and the user's emotional reaction. An interface that is easy to use, effective, and fulfilling is what defines a positive user experience, since it increases user satisfaction and engagement. A product's usability value is bound to rise when its user experience is improved. One component of the UX development concept is usability (Sukamto et al., 2020).
- **User Interface (UI):** The visible and interactive portion of a software programme, website, or device that people engage with is called the user interface, or UI. It has every graphical and interactive component needed for user navigation and interaction, including menus, buttons, icons, and screens. A well-designed user interface improves usability and improves the satisfaction of the user.
- **User centred design:** User centred design, also referred to as UCD is an iterative design method that is focused on the preferences and needs of the users in the entire development process of a system or product. UCD seeks to provide solutions that are more efficient, effective, and rewarding for the intended users (Aryo et al., 2017). To make sure that the finished design meets user expectations, it entails ongoing user research, feedback gathering, and usability testing.

2.2 Evolution of web design: Responsive Design

Through static web pages to the dynamic, multi-device environment we see today, web design has undergone a revolutionary journey. Websites were traditionally created with preset layouts designed for desktop monitors. The increasing number of different devices that challenged the consistency of user experiences proved to be a limitation to this approach. Web applications used to have their own mobile user interfaces, which may or may not have the same features as the desktop version (Voutilainen et al., 2015). Pre-responsive web design era comprised of static and adaptive designs often leading to inconsistent and poor user experiences across various devices.

2.3 Responsive web design as a solution

Responsive web design is a collection of methods and techniques in a project's layout level. It is a design and development technique in developing web applications that respond to various screen sizes and devices to offer the best possible viewing and engaging experience (Ezwan et al., 2015). To make sure that web content and design elements automatically shift to fit the screen, whether it's a desktop monitor, tablet, or mobile device. The core principles of responsive web design involve the use of flexible layouts, media queries flexible images and grids (Jiang et al., 2014). This technique improves user experience and accessibility on a wide range of platforms.

Ezwan et al., (2015) examined the best usability guidelines for responsive web design. The content analysis method was used in this paper to examine the data. Their findings showed that flexibility, familiarity, consistency, effective feedback, and aesthetic appeal are the five usability characteristics that are most effective in responsive web design.

Jeonghyun et al., (2015) suggested a responsive web design strategy that uses media queries to adjust the layout according to each device's resolution, maximising the delivery of content across all platforms. They also touched upon related studies, including the use of frameworks like Bootstrap, media queries, and techniques for controlling font sizes in responsive web design. The capabilities of responsive design have been strengthened by ongoing developments in CSS, HTML, and JavaScript, opening the door to more intricate and engaging online experiences.

The shift in web design from desktop-centric, static layouts to responsive, multi-device adaptable designs highlights the significance of responsive design in today's web ecosystem and represents the continuous effort to provide comprehensive and user-friendly digital experiences.

2.4 Impact on E-Commerce Growth

The previous ten years have seen an exponential growth in electronic commerce(e-commerce), which has resulted in a significant change in customer behaviour. Data indicate a consistent and noteworthy rise in e-commerce across multiple sectors, fuelled by the ease of use, availability, and variety of choices provided by digital channels (Verma et al., 2003). Customer preferences have significantly changed because of elements like seamless shopping, safe payment gateways, a wide selection of products, and the ease of doorstep delivery. Elida et al., (2021) talked about the growing dependence of consumers on e-commerce platforms and their demands for a smooth and fulfilling user experience. The increasing dependence on shopping online highlights how important user experience (UX) design is to capture and holding customers' attention. For e-commerce platforms to succeed in this cutthroat market, they must design interfaces that are user-centric, visually appealing, responsive across devices, and intuitive.

Due to owners' varying levels of digital literacy, small and medium-sized businesses (SMEs) frequently have difficulties implementing e-commerce (Herman & Umi 2022). This research modelled user experiences in SME e-commerce using a succinct User-Centred Design (UCD) methodology. Jeonghyun et al., (2015) provided examples of responsive web designs for a restaurant homepage and concluded that responsive web design can enhance the user experience and potentially improve advertising effectiveness in the internet market.

Responsive web design and e-commerce platforms work together to create a symbiotic partnership that has transformed the digital buying scene by allowing businesses to reach a larger audience and offer intuitive and seamless experiences across a variety of devices (Sudiana et al., 2021). With the incorporation of modern technologies such as augmented reality for engaging shopping experiences and AI-driven personalization for customised user journeys, this expansion presents chances for innovation in User experience design.

2.5 Related Works

Sugosha et al., (2021) examined the increasing significance of online sales applications, with a focus on the health sector and the pharmaceutical industry. The use of black box testing to verify that the functionalities functioned as intended for the e-commerce website's front end was highlighted in the study. And emphasised on creating the front-end for merchants' e-commerce websites and overlooked the broad analysis of the entire e-commerce system, including back-end functionalities, which could also significantly influence user experience.

Cheng et al., (2016) examined the functional architecture of college e-commerce systems, highlighting essential elements including online payment platforms, malls, online grocery stores, travel service platforms, and platforms for trading used commodities that are customised to the unique requirements of the student population. It however focuses only on the theoretical aspects, and overlooked the challenges in implementation, security concerns and scalability limitations.

Herman and Umi (2022) gave a user-focused approach by simplifying information gathering through interviews that cover habits, personal information, and business insights. The result is low-fidelity wireframes that provide SMEs with a basic framework for improving their e-commerce user experience. Without a development plan, the study lacked detailed guidelines or follow-up on how to implement these wireframes, potentially reducing the practical usability for SMEs.

Aryo et al., (2017) evaluated the efficiency and effectiveness of web content distribution. It investigates how applying Material Design Guidelines and User-Centred Design affects the efficiency of content delivery. Both strategies improved content delivery on smartphones and tablets. This study concentrated on only smartphones and tablets neglecting other screen devices and made use of only material design, ignoring alternative design frameworks or approaches that could have an impact on the way content is delivered.

Jiang et al., (2014) discusses the importance of responsive web design in displaying information of a web page in different devices. To achieve responsive design, it introduces methods like Media Queries and Bootstrap responsive navigation. The results of browser compatibility tests shows that responsivity of the design when using bootstrap offered good cross browser support. It failed to give insights on practical implementation for performance metrics, accessibility, or user engagement as it only focuses on browser compatibility.

Mohorovicic (2013) highlighted the usefulness of responsive web design in making websites responsive to a variety of devices, given the growing popularity of smartphones and tablets as internet-accessible devices. The advantages of responsive design are emphasised throughout the report, including better user experience, better SEO management, and a single URL for all devices. It also recognises possible difficulties like incompatibilities and performance problems.

Tiago et al., (2017) examined the impact that responsive web design has on blind users' emotional experiences. Even while responsive websites adhered to accessibility guidelines, actual testing showed that they presented serious usability issues for blind people, which caused them to feel extremely bad. It's interesting to note that compared to nonresponsive site designs, responsive ones caused blind individuals to feel more negatively. To guarantee that blind people have a positive online experience, this study emphasises the necessity for a more thorough approach to site design that has an emphasis on both accessibility and usability.

Nian and Zhang (2019) examined HTML5 and CSS3 based responsive web design. They created flexible online layouts and content by utilising the cross-platform capabilities of HTML5 and fluid grids, the media queries, and flexible images of CSS3. It however lacked quantitative metrics such as performance and loading times and no comprehensive user-centred evaluations or qualitative input to support statements regarding real-world perceived user experience gains brought about by the application of responsive design.

Sukamto et al., (2020) proposes ways to improve the user experience (UX) of a portal website through User-Centred Design (UCD). The study starts with a preliminary assessment using the User Experience Questionnaire (UEQ) and progressed on to the UCD process, which includes user modelling, system requirement definition, prototype evaluation and design. The final evaluation is carried out and compared to the first assessment. It however overlooked extensive real-world user testing scenarios, possibly leading to gaps in identifying practical usability issues or UX challenges encountered by actual users.

Lombardoni et al., (2005) proposed a systematic approach for designing complex e-commerce web apps offering highly customized products. The methodology aimed to reduce errors and support web architects in guiding users through product customization seamlessly. However, they focused on generating PHP pages as proof of concept, limiting insights into other advanced frameworks for implementing responsive web designs.

2.6 Research Gap

A review of all the papers presented shows a significant research gap emerges in the investigation of interfacing an AI-driven store assistant chatbot with responsive e-commerce platforms. Although conversations centre on the use of AI to improve user experience and responsive design in e-commerce, there has not been much research done on how these two technologies may work together. The studies mostly concentrate on two different aspects: the adaptation of responsive design across devices and the function of AI in enhancing systems or improving user engagement. However, none of these really focus on the smooth integration of both, particularly in the context of e-commerce. Investigation into how these technologies could work in harmony to produce a user-friendly shopping experience that balances adaptability of devices and personalized user interactions with AI-driven chatbots remains an unexplored territory. Closing this gap could provide new insights into how best to engage and satisfy users in modern e-commerce platforms.

The artefact's design will be developed from the following three perspectives in this study to fill in and enhance the gaps in the literature:

- Responsive framework integration like tailwind CSS, offering pre-built grids and responsive components which would expedite the development.

- Intelligent interaction using artificial intelligence (AI) in the e-commerce platform to boost user engagement.
- Designing a responsive application accessible by any screen device or operating system.

3. METHODOLOGY

The e-commerce platform's operation and customer satisfaction are largely dependent on the techniques used in its development and assessment. This section describes the thorough methodology that was used to plan, execute, and evaluate the platform's performance in satisfying user and market demands.

3.1 Research Approach

This project's research strategy entails a thorough literature search on current responsive web design applications with a focus on e-commerce apps and systems to investigate the techniques used in the development of some of these systems. This search also assisted in locating some problems and restrictions in the current user experience of e-commerce platforms, which was considered in this project to suggest a better and more effective approach. Robust research databases, including ACM Digital Library, IEEE, and Research Gate, were carefully used to conduct thorough searches. The results from these databases were used to gather significant information for earlier systems' development. Furthermore, RefWorks was helpful in creating and maintaining personal reference databases that offered a more organised method of seeing and citing references.

The 'design and creation strategy' is adopted as the research strategy for this project. Through this methodology, it is possible to integrate qualitative research with practical implementation. Knowledge gathered from the literature review and technical background will be used in implementing an e-commerce platform aimed at improving users' experience. Since this process is iterative, we may use the evaluation stage's input to enhance and improve the artefact that will be produced. This would ensure that the artefact meet the aims and objectives of this research. The iterative steps adopted from this strategy are awareness, suggestion, development, evaluation, and conclusion.

Awareness: This is the process of identifying an issue through a review of several literatures in which experts or researchers have determined that a software product needs more research or development.

Suggestion: When these research problems have been identified after reviewing existing literature, suggestions are given to fix them through critical thinking.

Development: this involves putting the suggested solutions to the research problems into practice. The e-commerce platform will be developed using JavaScript programming language, and MERN frameworks (MongoDB, Express, React and Node).

Evaluation: The software that has been produced will be examined during the evaluation phase to see if it fulfils the intended use. The artefact should perform all functionalities expected of an e-commerce application and maintain consistent user experience across various devices.

Conclusion: This gives an overview of the design process, knowledge gained from the overall design, unforeseen problems encountered, and certain design elements that might benefit from additional study and development.

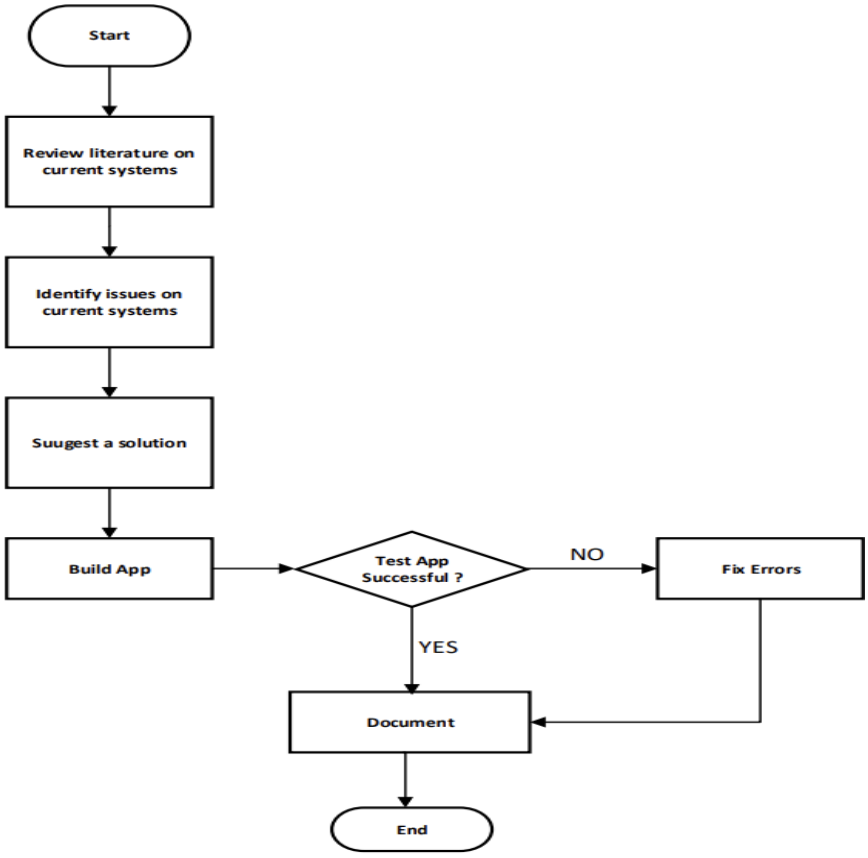

Figure 1 Flow chart of project methodology

3.2 Requirements Gathering

This is the first stage of the project where all relevant data and user expectations are gathered and recorded. This procedure involves compiling, evaluating, and specifying the requirements, goals, and limitations of the proposed system to ensure it meets users' expectations.

Some of the functional requirements/features of the proposed system are highlighted below:

- The proposed system should function efficiently as an e-commerce store allowing users to shop items.
- For the admin user to access the admin dashboard, the proposed system must have an authentication mechanism to validate the credentials.
- When an invalid login credential is used, the proposed system should give an error message.
- Admin user should be able to login into the system and perform admin duties (check inventory, add items, see orders/sales etc).
- Users should be able to perform operational tasks such as viewing items, adding items to cart, and checkout.
- Users should be able to interact with the shop assistant artificial intelligence (AI) chatbot to aid their shopping experience on the system.
- The proposed system should be responsive and remain consistent across different screen devices.
- The user experience of the proposed system must be straightforward in terms of navigations to different modules of the app.
- The proposed system should be easy to use and do not require prior training or manual.
- The proposed system must be able to run on different operating system versions (windows, android, ubuntu etc).

3.3 Agile Methodology

Agile methodology is a flexible, iterative approach to software development in which strong emphasis is placed on client interaction, flexibility, and teamwork throughout the development process (Chan et al., 2009). High priority is placed on producing usable product increments in brief cycles, referred to as sprints or iterations, that normally run one to four weeks. Some agile methodology frameworks include scrum, lean, kanban, extreme programming etc. For this project, the kanban framework would be used.

3.3.1 Kanban Framework

The Kanban system focuses on job visualisation, usually with a Kanban board that has columns for corresponding task stages. As work is completed, work items are moved across the board to assist teams in visualising workflow and limiting work-in-progress (C. Matthies, 2018). This gives transparency and an overview of all the stages of the development process.

The project management tool used for this project was Trello as it provides an effective visual approach in implementing the kanban methodology. Also, Because of its great flexibility and ability to fit into different workflows, it's widely used for both individual and group task management.

The figure below shows the Trello board which highlights the different tasks and stages giving a clear description of the project features and keeping track of the progress made.

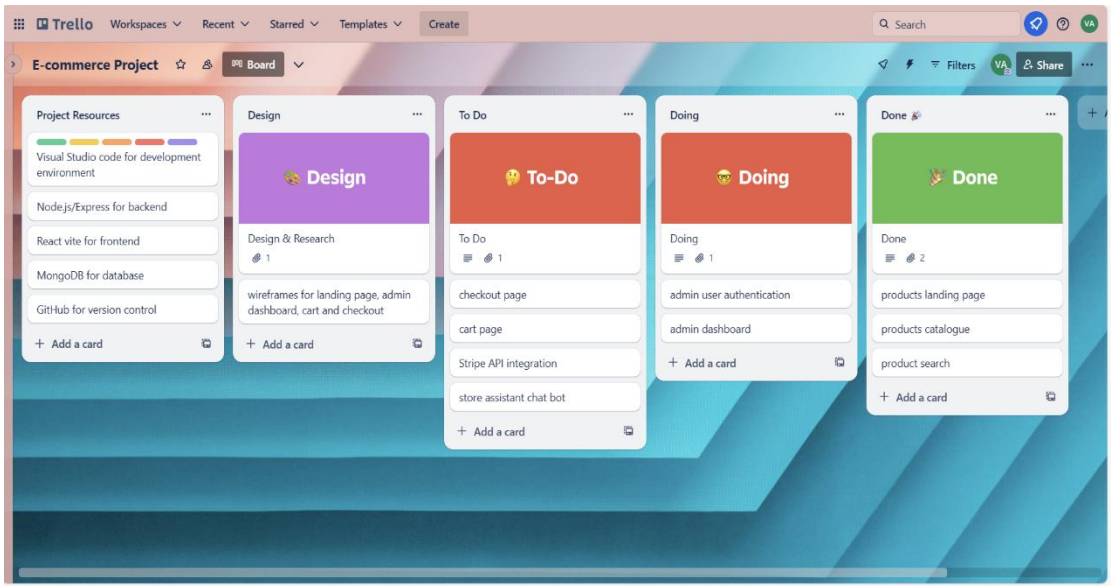

Figure 2 Trello kanban board

The kanban board has four cards, namely design, to do, doing and done. The design card contains all the tasks relating to designing the user interface. The to do card contains tasks that are yet to be done. The doing card contains tasks that are currently been worked on, while after any of the tasks are completed, they are moved to the done card.

3.4 Project Lifecycle

In the development of every software, there is a process known as the software development lifecycle (SDLC). This is a method used by software development teams to plan, create, test, and deploy software, meeting the user requirements, and upholding a high standard throughout the process (S. Gupta and N. Gayathri, 2022).

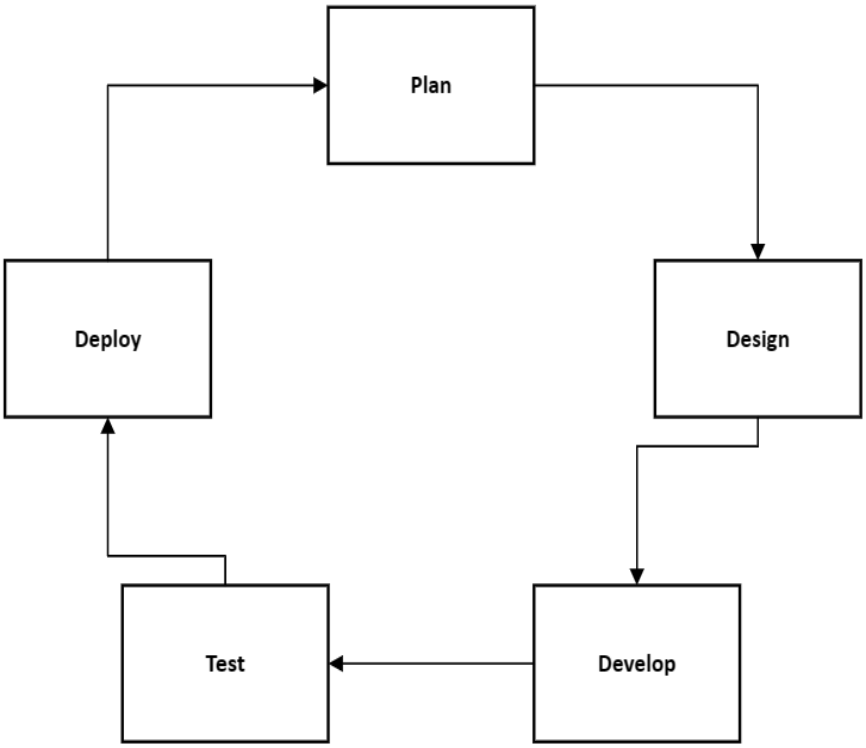

Figure 3 SDLC

Plan: this is the first stage in the software development life cycle. The project's objectives, requirements, scope, and resources are all defined. It lays the foundation for the whole undertaking. In order to develop the e-commerce application, sufficient information about the programming language and frameworks to be used, the database needed, and any additional materials that will aid in its successful completion was gathered through comprehensive web research.

Design: In the design phase, the architecture, structure, and functionality of the system are designed and documented, which comes before real coding starts.

Develop: the coding implementation is done at this stage using JavaScript programming language and libraries such as React, Node, Tailwind CSS etc.

Test: the testing stage comes after the code implementation. Before software or systems are deployed to users or customers, the main goal of testing is to make sure they function as intended, meet all criteria, and are error-free.

Deploy: after the successful completion of all the previous phases and ensuring all the requirements laid out have been met satisfactorily, the application is then deployed. It involves putting the system or software into use in a production setting so that end users can access and use it.

4. SYSTEM ARCHITECTURE AND IMPLEMENTATION

This section gives details about the development steps, all the tools and resources used in the implementation of the e-commerce application. The application is broken down into three (3) parts, the frontend (client-side) which was developed using React JavaScript library and tailwind CSS, the backend (server-side) which was developed using Node/Express framework and MongoDB which served as the database tool.

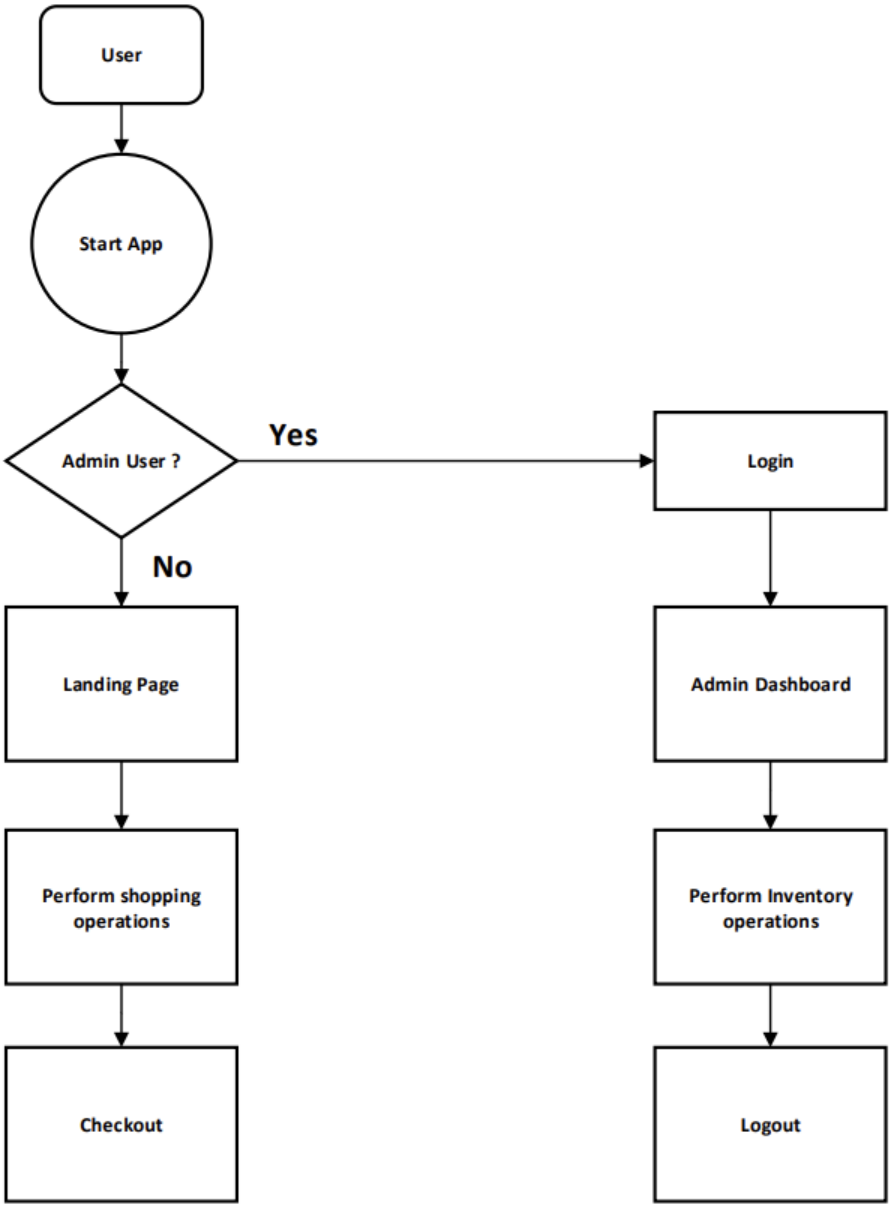

Figure 4 Application system Flow

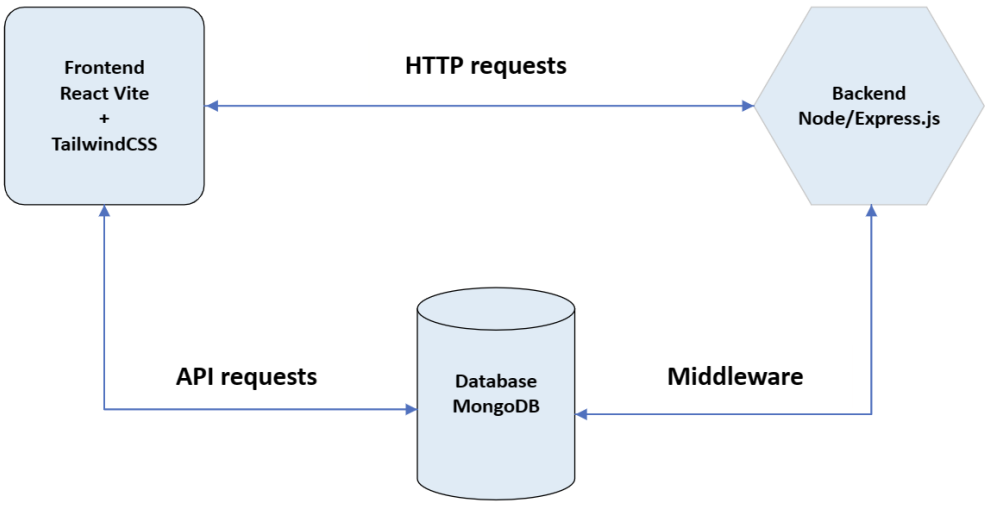

Figure 5 System Architecture

4.1 Frontend (client side) development

4.1.1 React Vite

React vite library was used in developing the frontend part of the application. React is an open-source JavaScript library that is used in developing user interfaces that are reusable and manage their own state efficiently when changes occur in the application. React vite was chosen for this project because of its remarkable development server which offers quick code builds resulting in quick cycles for testing and development. Also, it is supported by a vibrant community that is always expanding, offering resources, documentation, and continuous enhancements to guarantee improved support and updates.

4.1.2 Tailwind CSS

Tailwind CSS is a cascading style sheet (CSS) framework that uses a utility-first approach in styling web applications. With Tailwind, stylistic features are applied to HTML elements directly through utility classes, as opposed to standard CSS frameworks that offer pre-built components, hence offering more flexibility in usage. Some of the unique features of tailwind CSS which gave rise to its selection for this project include:

- Responsive design: rather than using custom CSS in creating responsive designs which is leads to writing more codes and longer time for development, tailwind CSS offers easy to use utility classes that seamlessly creates layouts which can adapt to various screen sizes and defining different style classes that become visible at certain breakpoints.
- UI consistency: it has a predefined set of utility classes that ensures the consistency of the UI in the entire application. Similar elements can make use of the same class hence ensuring design uniformity in the project.
- Quick development: eliminating the need for writing custom CSS leads to quicker development time.
- Optimization: in comparison to custom CSS, tailwind CSS reduces the redundant unused code as it only generates CSS files that is going to be used. This makes the size of stylesheet file light weight and leads to quicker load times when the application is accessed, ensuring optimization.

Tailwind CSS is an effective choice for creating cutting-edge, scalable responsive web applications because it provides a distinctive visual methodology that prioritises speed, consistency, and ease of customisation.

4.2 Backend (server side) development

4.2.1 Node/Express framework

Node.js is a JavaScript runtime environment for servers that is renowned for its event-driven approach and non-blocking input/output operations, because Node.js allows for the effective processing of several requests at once. It is a good choice for microservices architectures and real-time applications (Liang et al., 2017). Express.js is a simple and adaptable web application framework for Node.js. It offers an extensive feature set for developing APIs and online applications. It allows developers to concentrate on creating the logic for the application while handling HTTP requests, setting routes, and managing middleware more easily.

One of the major reasons for using this tool for the backend development was because it allows for development in a single language (JavaScript) throughout the stack since the same language is already in use for the frontend, which encourages code reusability and results in consistent data structures. Also, it is highly efficient in managing concurrent requests and input/output-intensive processes, improving scalability and performance of web applications.

4.3 Database

4.3.1 MongoDB

MongoDB is a popular nonrelational (NOSQL) database management system which offers a high performance and flexible way of storing data. It offers an elastic data storage strategy that makes it simple for users to store and query many forms of multivariate data. This makes database management easier for developers and makes the environment for cross-platform services and apps very scalable (Liang et al., 2017).

4.4 Integrated Development Environment (IDE)

4.4.1 Visual studio code (VS code)

IDE is a software tool that gives developers all the tools and features they need to create software applications. For this project, VS code was used for the code implementation. It was chosen for this project because it offers an optimized text editor, with an efficient debugger which helps while finding and fixing bugs. It contains build automation tools and supports version control integration that is essential for tracking code changes and keeping track of version history during development.

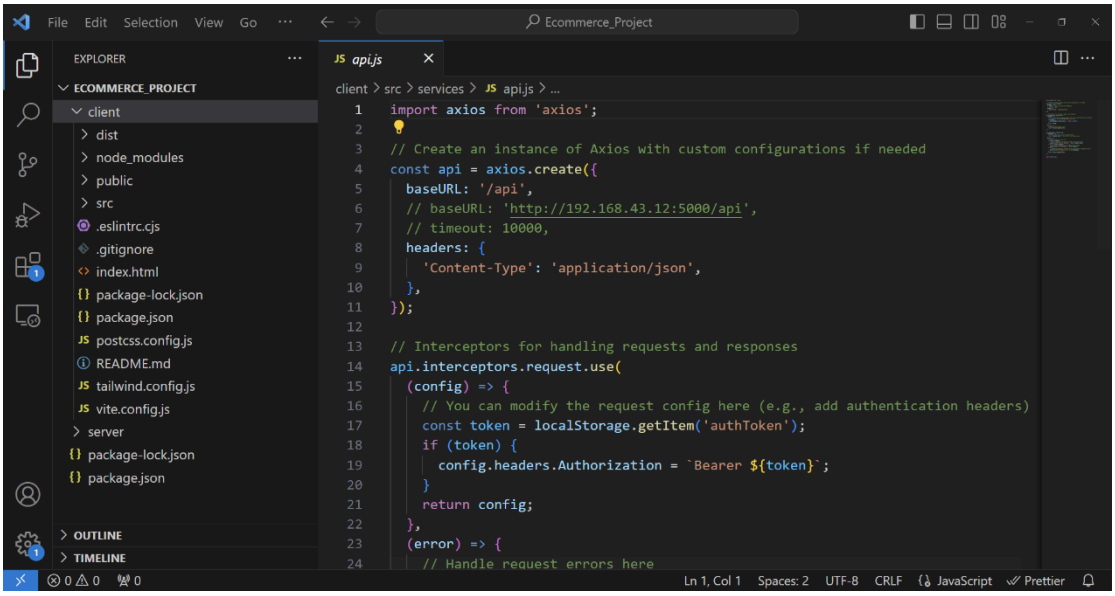

Figure 6 VS code editor

4.5 Version Control

4.5.1 Git

Version control is an integral part in the development of any software. Git allows programmers work together, keep track of changes made to their codebase, and oversee several project iterations. Through the usage of git, code implementation made on the local machine using VS code was committed and pushed to an online repository (GitHub) which is a web-based hosting solution for git repositories. Here is a link to the project's repository on GitHub: ainobs/Ecommerce_Project (github.com)

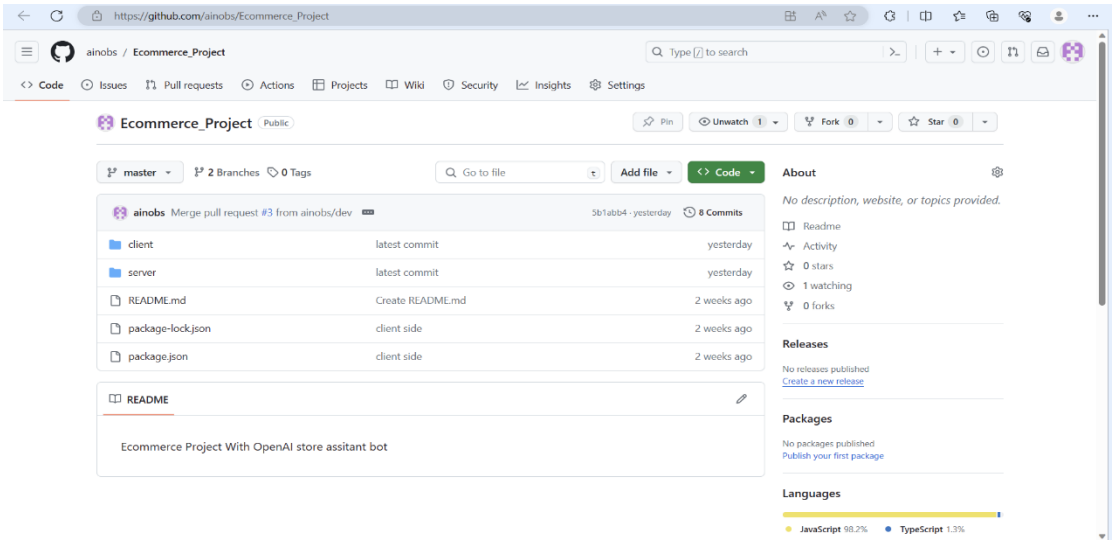

Figure 7 GitHub repository

In the development phase, the GitHub repository was initialised with a "master" branch which is the primary branch containing all the files and directories of the project. Following git's best practices concerning branching, a new branch was created "dev", and all modifications were stored in this separate "dev" branch. All the code changes that were made during development was committed and then pushed to the "dev" branch before a pull request is created. Afterwards, the updates from the "dev" branch are merged into the "master" branch. This continuous procedure continued throughout the development process, making it easier to roll back to previous iterations and keep an eye on updates in case problems occurred with the application. Making it easier to find and fix bugs/issues that may arise after new updates are made.

4.6 Stripe

Stripe is a popular payment processing platform which enables companies to manage online transactions effectively and safely. It offers a set of tools and APIs that let companies to manage and accept online payments. Stripe API is built around REST (representation state transfer) architectural style and provides unique API keys, which are used to authenticate requests. Test mode API keys were integrated into the e-commerce application. Stripe was chosen to handle the payment channel because it focuses on scalability, security and provides a user-friendly interface. It supports different payment methods including debit or credit cards and digital wallets like google pay.

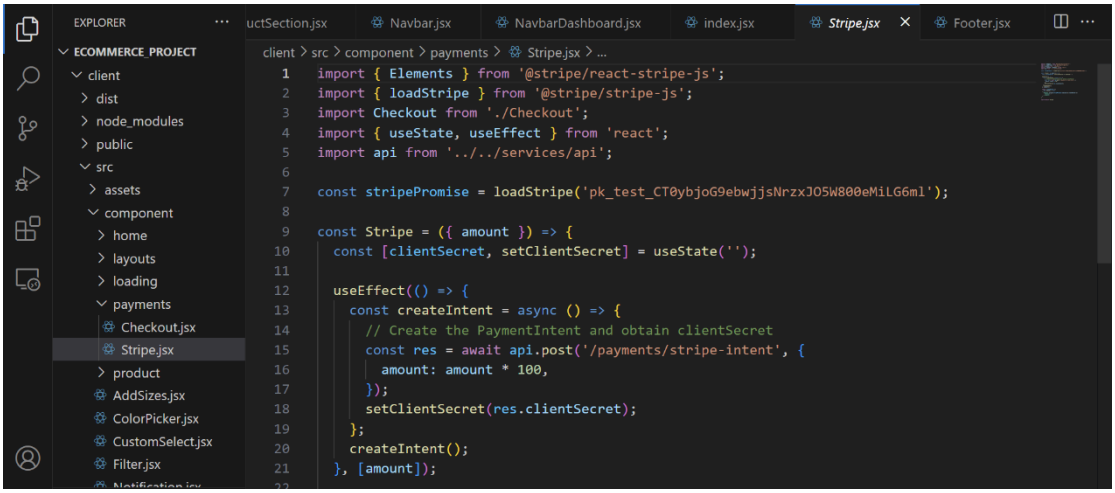

Figure 8 Stripe API integration

4.7 OpenAI

OpenAI provides a powerful language model which are designed to handle various processing tasks, text, and voice recognition in natural language. In the e-commerce application, OpenAI's 3.5 turbo model was used to handle chat completions. This model is designed to comprehend natural language and produce responses that are similar to those of a human. With the use of a large dataset and state-of-the-art language processing techniques, it provides enhanced capabilities for comprehending context, delivering precise information, and having meaningful discussions. The 3.5 turbo model is a good choice for chatbot applications, customer support, content creation, and several other natural language processing jobs since it incorporates cutting-edge technology to replicate human-like text responses. Since, this project is aimed at enhancing the user experience while shopping, structured interactions between different roles (system, user, assistant) in a conversation was used in developing and testing of the AI's responses to various user inputs, guaranteeing that the chatbot offers appropriate and useful answers to frequently asked questions.

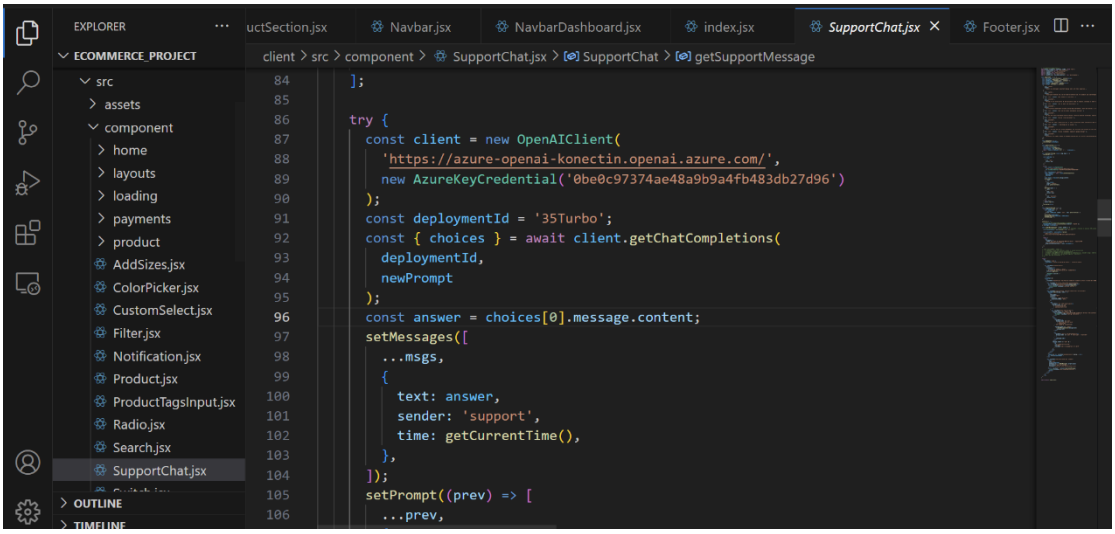

Figure 9 OpenAI integration

4.8 API Testing (Postman)

Postman is an effective tool for testing and documenting APIs. It makes managing APIs easier by offering a user-friendly platform for submitting requests, reviewing results, and automating workflows. Developers can create several HTTP requests with Postman, such as POST, GET, PUT, and DELETE, enabling thorough API testing and debugging (Dave Westerveld, 2021).

It was necessary to use postman in carrying out these tests to ensure that all the backend services are functioning as intended. Also, the responses from these tests are delivered in JSON format for ease of reading and includes a response code output (200 for a successful response or 401 for a failed request). The HTTP requests used during the API testing are described below:

- **POST:** this is used to send data to a designated resource to be processed.
- **GET:** this is used to retrieve data from a designated resource.
- **PUT:** this is used in updating a given resource with fresh information. It completely replaces the current resource and has the ability to create new ones if none already exist.
- **DELETE:** this eliminates a given resource. It removes the data from the server.

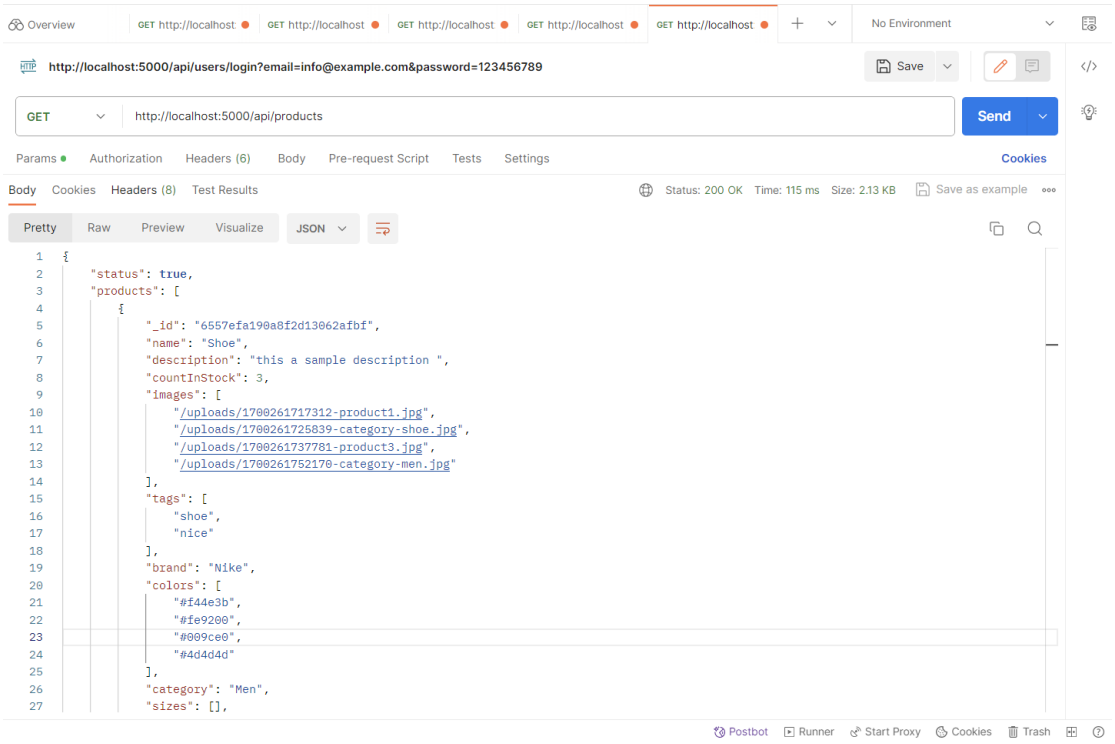

Figure 10 Products API endpoint test

The figure above shows the postman dashboard while conducting an API test on the products endpoint (http://localhost:5000/api/products). The GET request returns a status code 200, which signifies success and returns an output of all the products in the database in JSON (JavaScript object notation) format in key value pairs.

4.9 Implementation

The e-commerce application was created to improve overall user experience and engagement leading to customer retention for organisation's e-commerce platforms. The design and development of this system was accomplished, and the objectives of the implementation were met. This section highlights the features of the developed application along with descriptions.

4.9.1 Admin Login

Before an admin user can have access to the dashboard, login access is required. For the sake of the user testing, which is meant to be anonymous without collecting any personal information, the testers would not be required to register. Instead, a unique admin login key was created which would be used to login into the admin dashboard module.

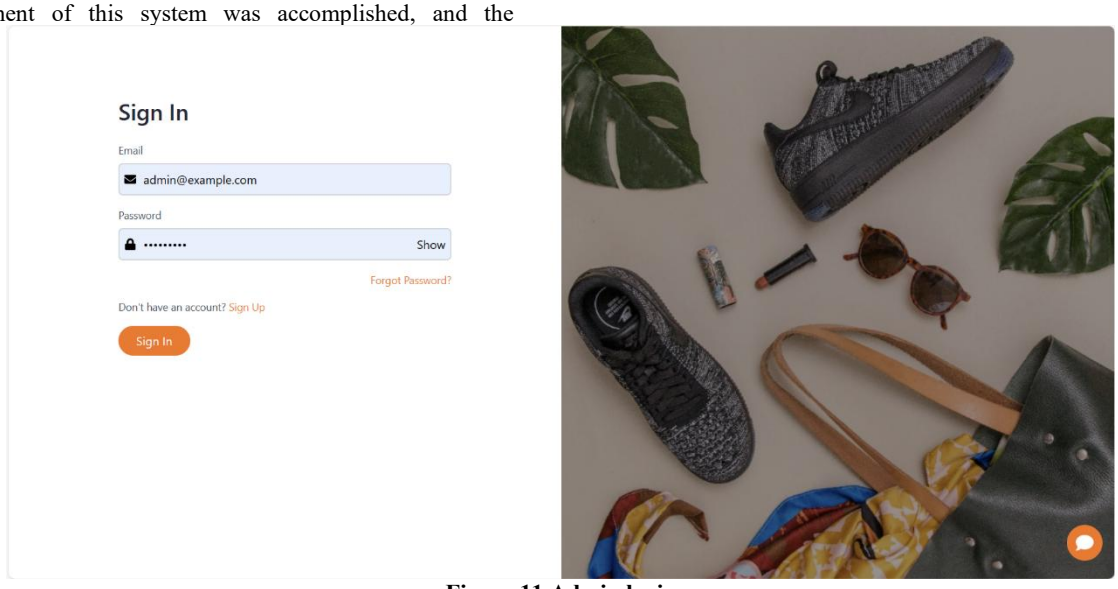

Figure 11 Admin login

4.9.2 Admin dashboard

The admin dashboard gives an overview of the orders, products, profile and analytics of the e-commerce platform.

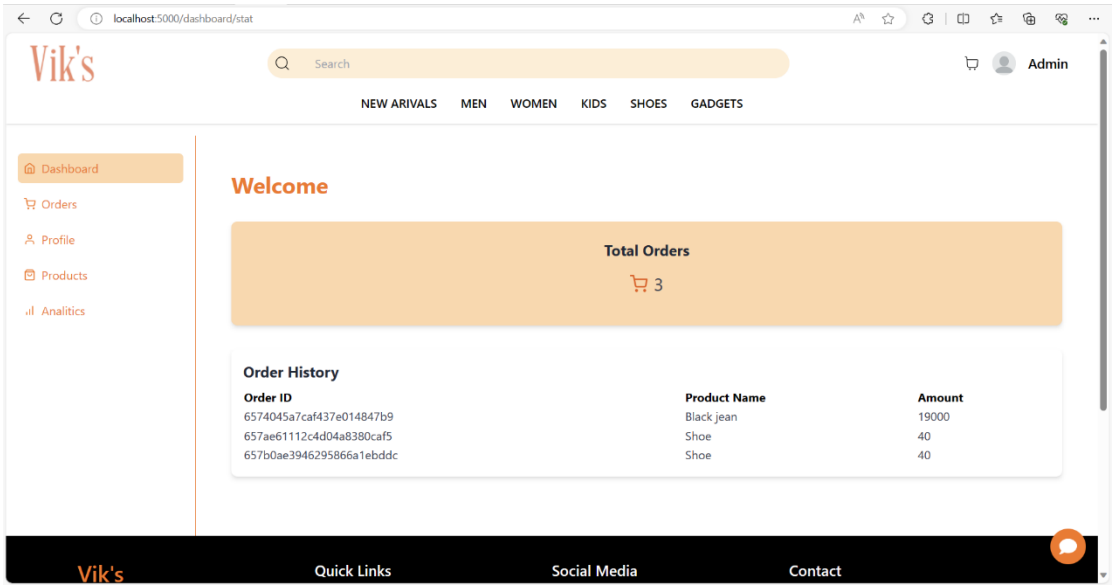

Figure 12 Admin dashboard

From the figure above, the dashboard gives details about the transaction history on the total orders made by the customers, the product name along with their respective amounts. The sidebar contains the orders, profile, products and analytics tab.

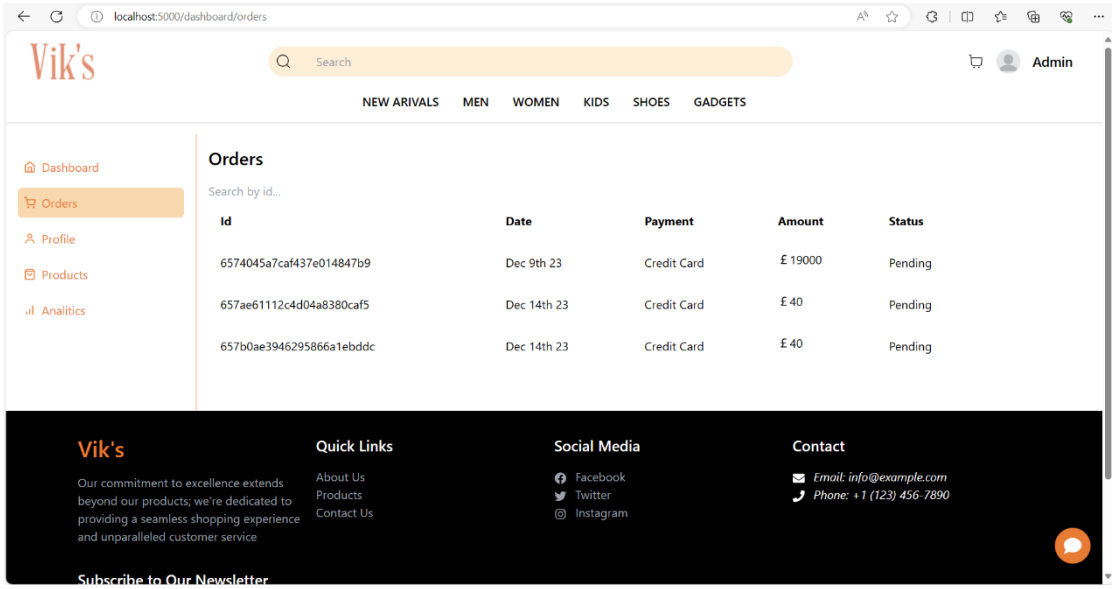

Figure 13 Customer Orders

The orders tab gives details of the unique transaction id of every order, the date the order was made, the payment method (credit or debit card), the amount of the product and the status of the transaction. A search bar is available to enable filtering through the list and returning a particular item based on its transaction id.

On the products tab, admin users can view the entire stock collections of the store. The product name, number in stock, date the product was added and price. The products catalogue can also be filtered through using the product name.

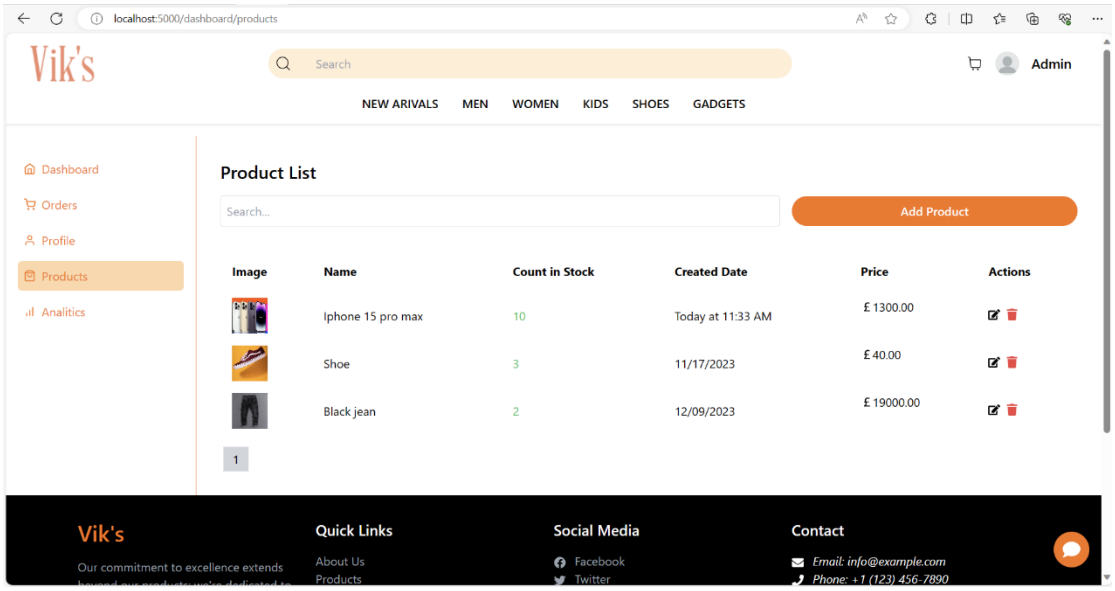

Figure 14 Products Catalogue

Admin users can also perform inventory operations such as adding items, edit the existing items (price, brand, description etc) and delete items. When adding a product, the product name, category, selling price and cost price, brand, tag, description, number in stock, colour and product image are all required.

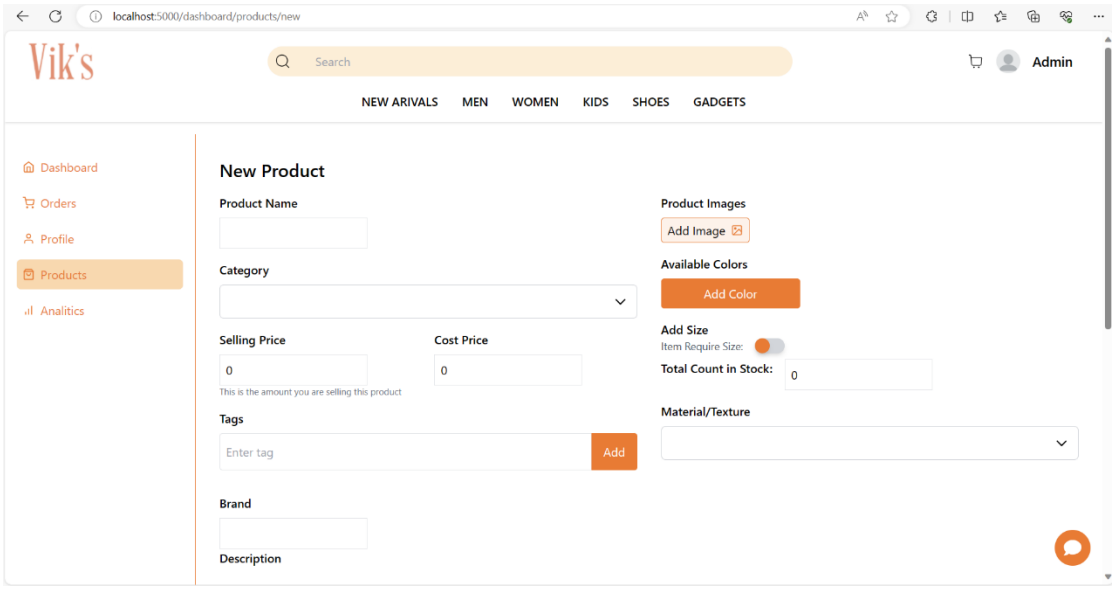

Figure 15 Add product

The analytics tab gives information about the number of users on the system and the total number of available products.

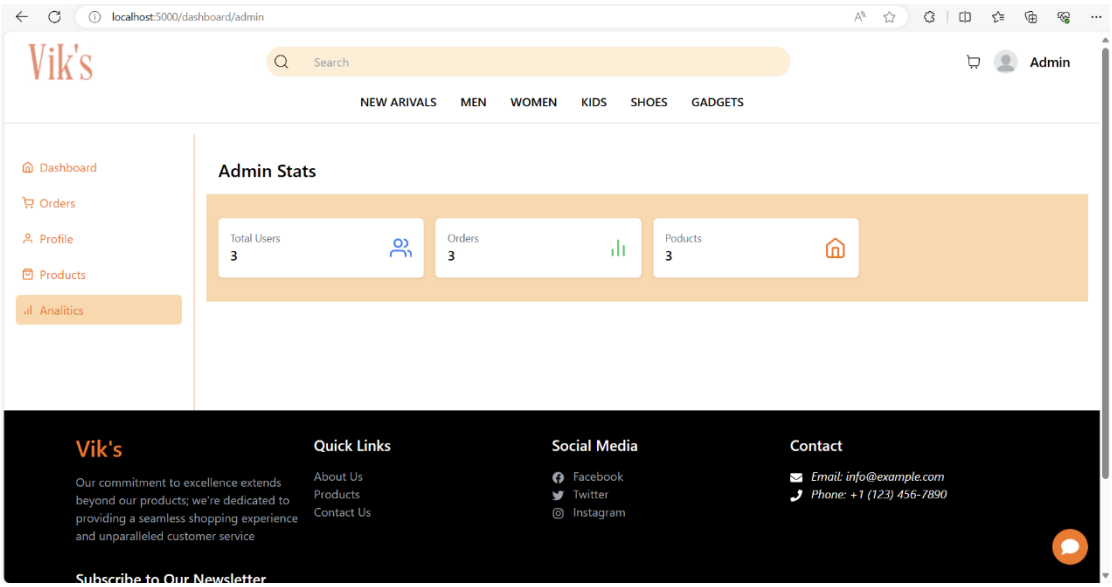

Figure 16 Store Analytics

4.9.3 User landing page

This is the starting point of the application when accessed by users. The page contains a slideshow hero image showcasing products. The floating chat icon at the bottom right of the page is the AI powered store assistant whose sole function is aid users in their shopping needs and enhance the user experience by providing quick responses and information about where to find their item of choice.

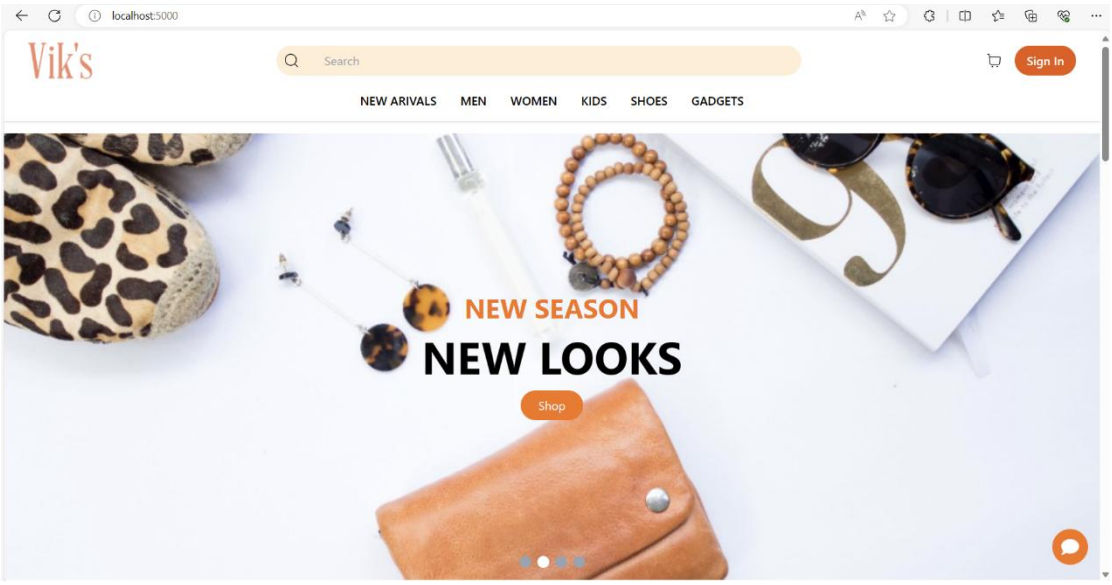

Figure 17 Landing page

4.9.4 AI powered store assistant

Leveraging on the service from Azure OpenAI 3.5-Turbo model, the store assistant was trained to perform as a shopping aid offering chat completions to the users by interacting with them to know their needs and providing responses based on these requests. The response is always a link to a section of the products section where the item they are looking can be found if available in the store.

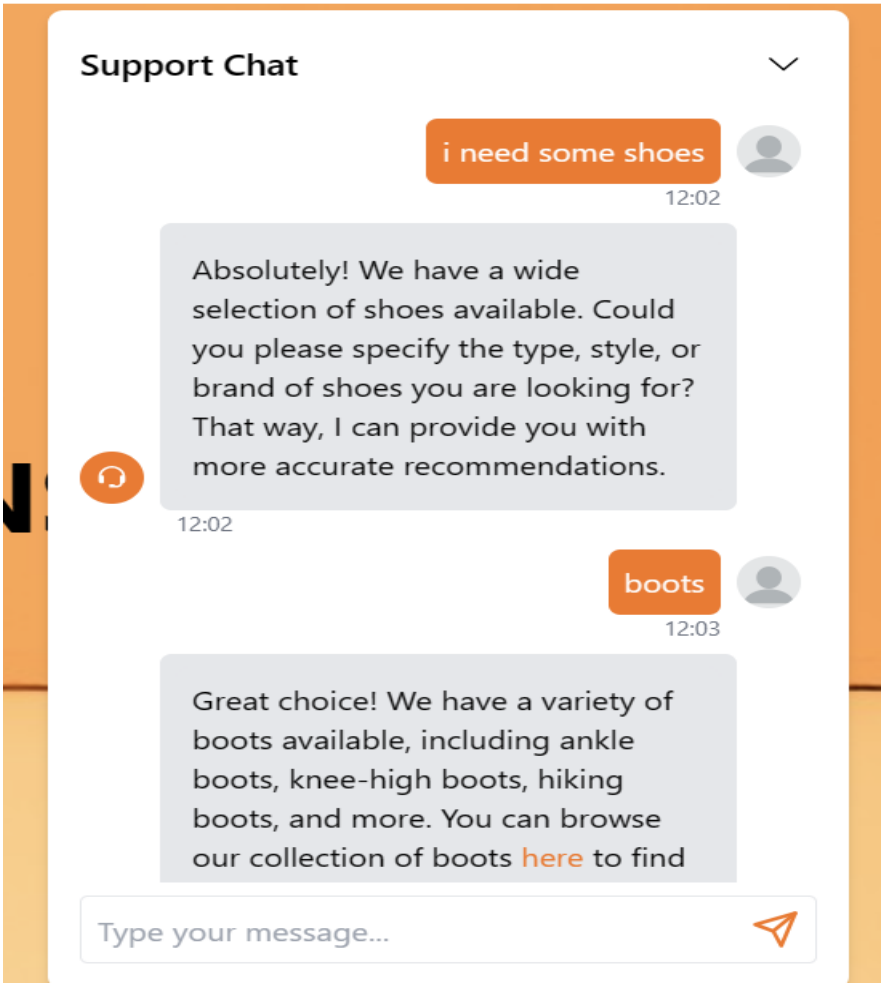

Figure 18 AI store assistant

From the response in the above figure, it is seen that the store assistant possesses chat completion compatibility using natural language. Clicking on the link provided redirects to the products section where the item could be found if available.

4.9.5 Checkout

When the user has found the product of interest, the items can be added to the cart. As many items as possible can be added to the cart, then the user can proceed to checkout.

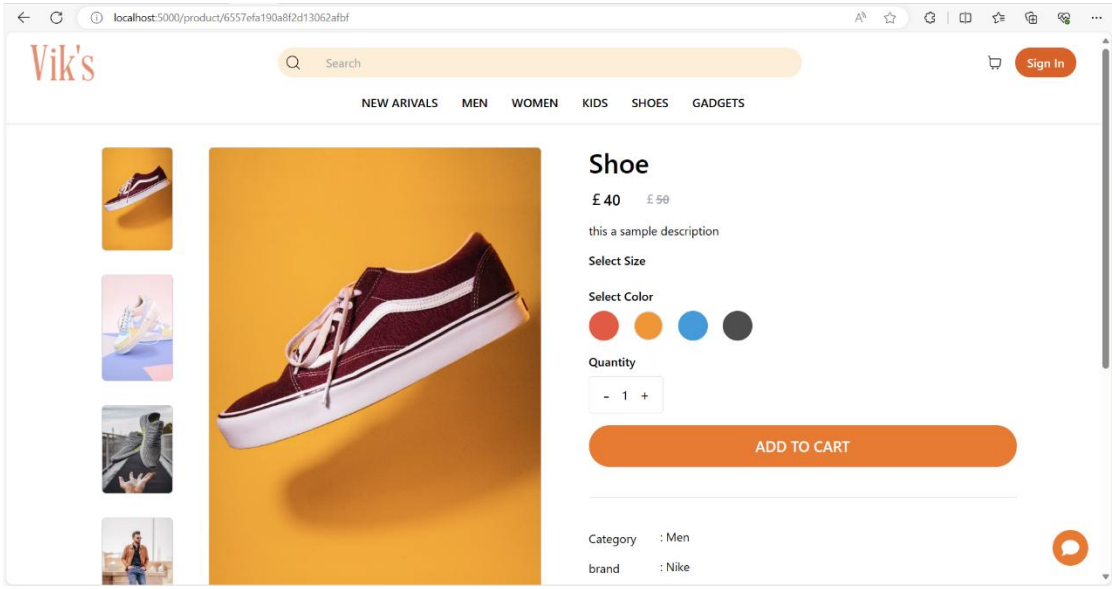

Figure 19 Add to cart

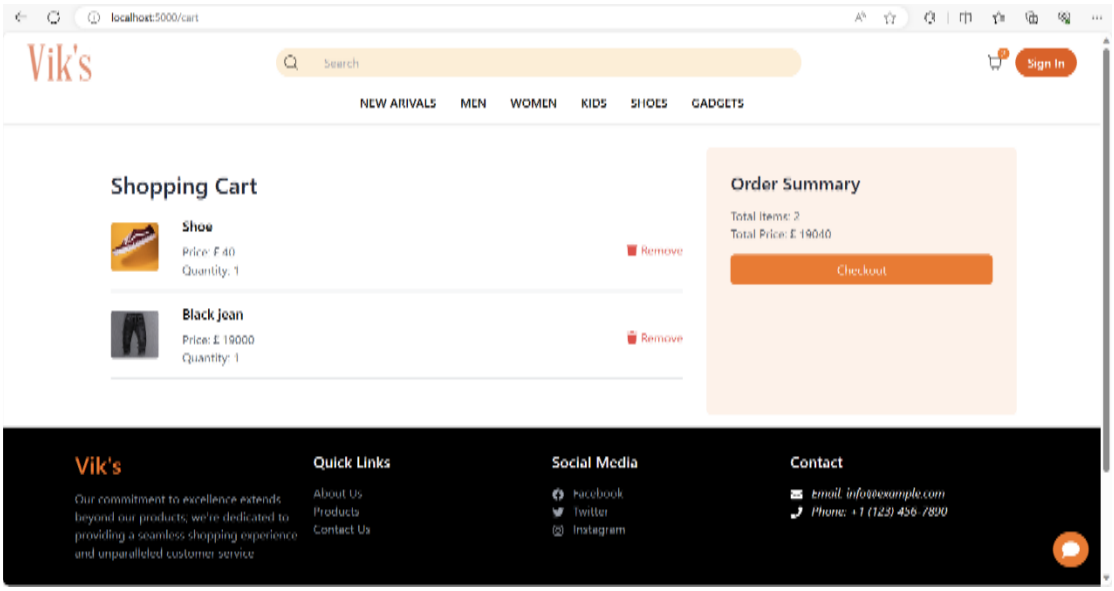

Figure 20 Shopping cart

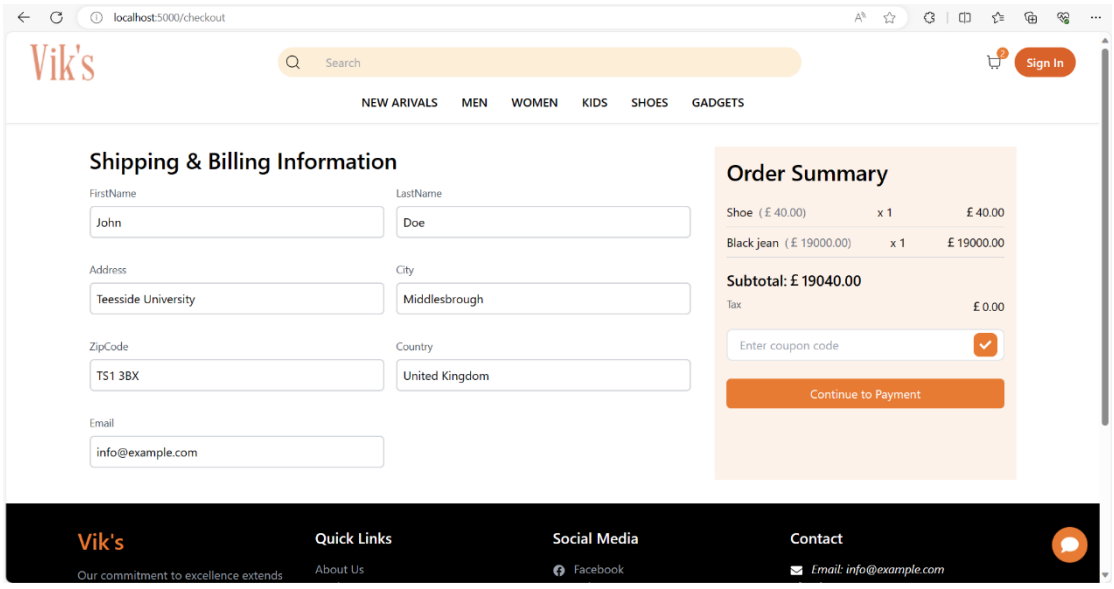

Figure 21 Shipping & billing information

For testing purposes, dummy data is used for the shipping and billing information since no personal information will be collected from the users. The order summary on the right gives details about all the selected items in the cart, their quantity and total cost. After all the required fields are completed, then the user can proceed to payment.

Stripe API payment channel was integrated into the application to handle the checkout process.

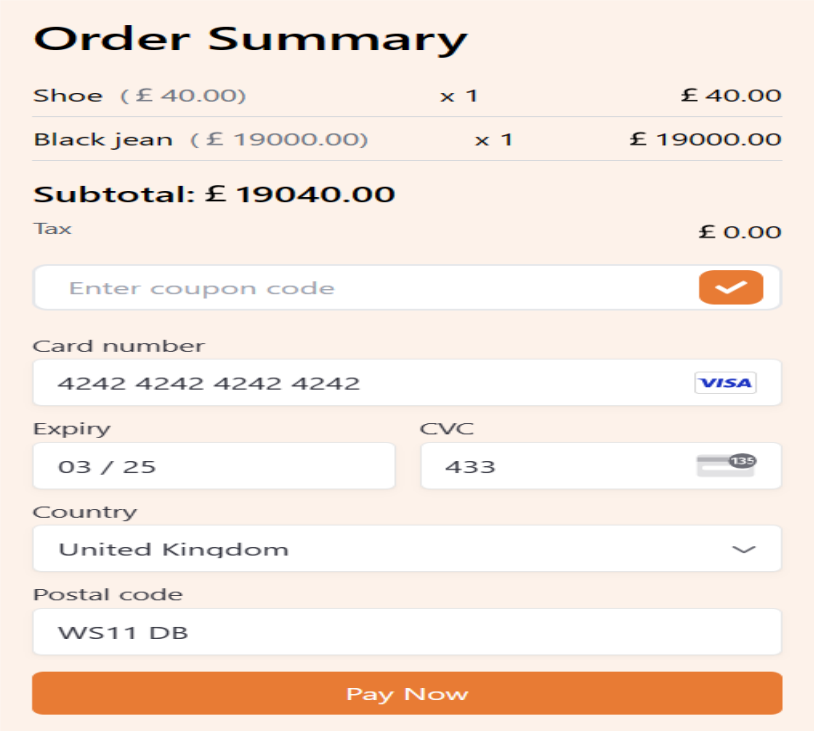

Figure 22 Stripe payment

Stripe API provides dummy card data from different card brands (Visa, MasterCard, American Express etc) to test the payment gateway to ensure the correct integration. After the transaction is confirmed, a success page is displayed to the user, and all the information of this current order gets updated in the admin dashboard under the orders tab.

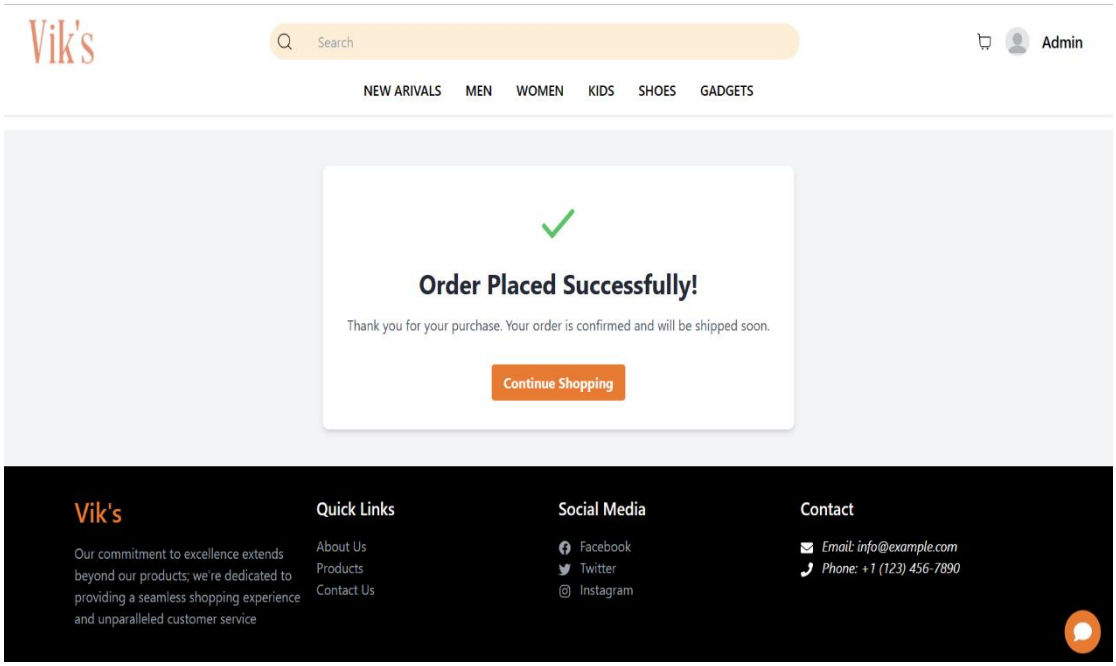

Figure 23 Order confirmation

5. EVALUATION AND DISCUSSION

This section will address the study's limits and ethical considerations in addition to examining and evaluating the results of the survey carried out from the user testing, considering the findings, and providing answers to the research questions that were presented at the beginning. Ten respondents participated in the survey.

5.1 Research Question One

Question: Does implementing responsive web design influence user engagement?

Answer: From the survey carried out, the results prove that responsive design does have a positive effect on the end users of an application. The results from this analysis are highlighted below:

- Consistency across devices: The results show a high degree of consistency across various devices suggesting that irrespective of the device they use, users had a flawless experience. This consistent behaviour promotes engagement as it grants users seamless access to the application on any device.

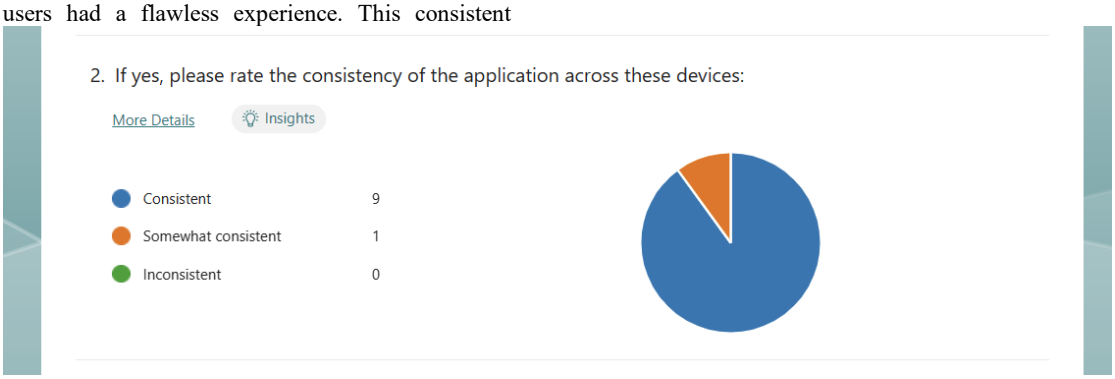

Figure 24 Survey result 1

- Visual appeal and aesthetics: The positive ratings for visual appeal and aesthetics suggest that the design elements were impressive. A visually appealing application is more likely to entice users to explore it further, which could result in more interaction and engagement.

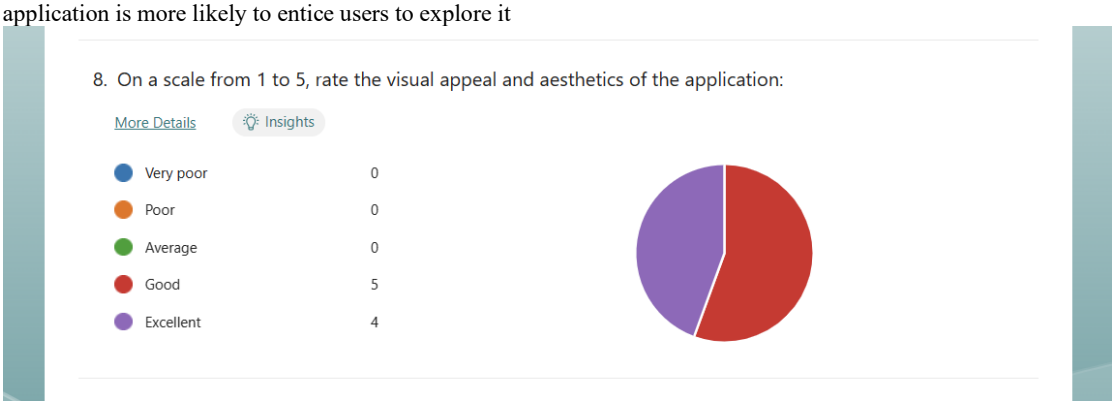

Figure 25 Survey result 2

- Accessibility: Accessibility is always guaranteed since responsive designs can adjust to different screen sizes. Users are more likely to interact with an application's features and content when they find it simple to access and utilise.
- User-centric approach: A user-centric approach is demonstrated by the emphasis on the design's responsiveness. By giving users a nice and aesthetically pleasing application, this approach helps to attract users and eventually enhance engagement.

In summary, the results of the survey confirm that the usage of responsive web design enhances user engagement through the provision of visually appealing aesthetics and consistency across devices, which in turn promotes more interaction and engagement within the application.

5.2 Research Question Two

Question: How effective is the use of AI in enhancing user experience?

Answer: The feedback from the survey on the impact of the chatbot AI on user engagement and experience is analysed below.

- Encouragement to explore: From the survey data, an overwhelming 80% of respondents to the survey agreed that the chatbot prompted them to browse more products or website sections. This implies that by focusing users' attention on different parts or items, the AI assistant encouraged exploration and actively engaged users.

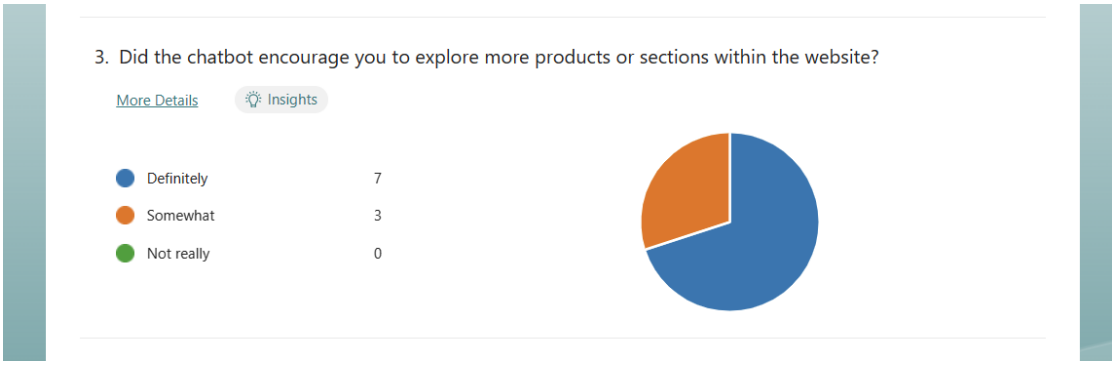

Figure 26 Survey result 3

- Ease of use and navigation: 80% of respondents thought the chatbot's UI was simple to use and navigate. This encouraging response suggests that the functionality and design of the interface helped to provide a smooth user experience by encouraging exploration and engagement.

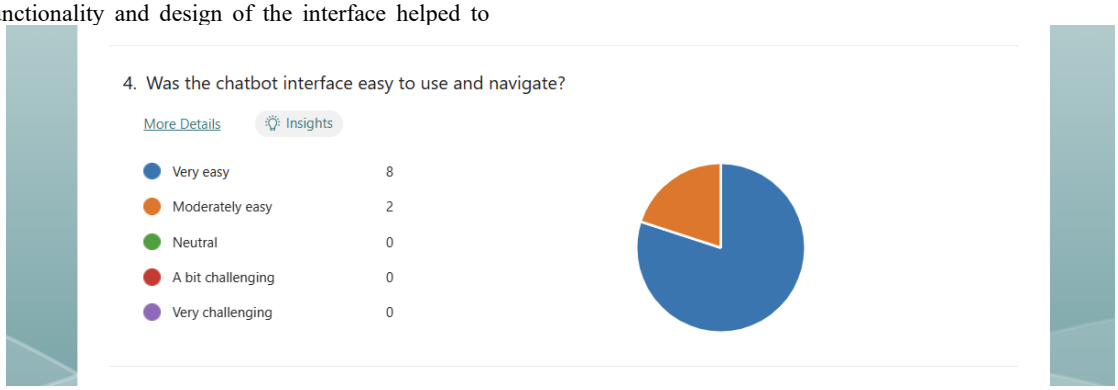

Figure 27 Survey result 4

- Preference over traditional navigation: A great percentage of the respondents said they found the chatbot to be more useful than standard navigation tools like search bars or menus. This indicates a movement in customer preference towards AI-powered support, demonstrating how well the chatbot functions to assist users on the website.

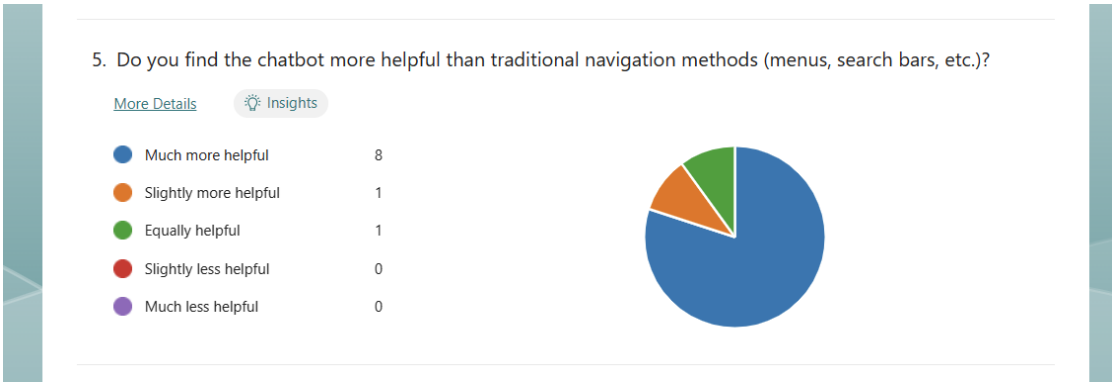

Figure 28 Survey result 5

- Influence on purchasing decisions: From the survey data, a great number of the respondents thought to some extent the chatbot had some influence over their buying decisions. This illustrates how the chatbot affects users' decision-making processes and could lead to greater satisfaction among users.

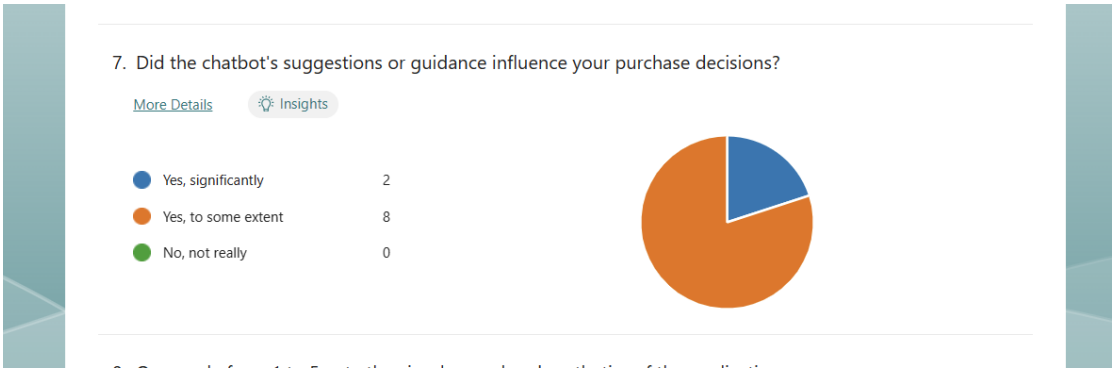

Figure 29 Survey result 6

In conclusion, the survey data clearly suggests that the chatbot dramatically improved customer engagement by easing navigation, promoting exploration, outperforming conventional techniques, and influencing users' purchase decisions. These findings highlight how important AI-driven support is to improving user experience and engagement on e-commerce platforms.

5.3 Research Question Three

Question: What strategies/design patterns can be employed to optimize content and layout for different screens?

Answer: The process of making content and layout more screen-friendly for a variety of devices involves several design techniques. In implementing the e-commerce application using responsive design approach, media queries, flexible images, and fluid grids were employed to dynamically adapt the page content to various screen sizes. Using this method guarantees that the layout will always be aesthetically pleasing and will remain readable and usable on any device—desktop, tablet, or smartphone.

Using adaptive layouts, which entail the creation of numerous fixed layout sizes suited to a particular device category, is another effective strategy. It can produce a more tailored and optimised experience for specific screen sizes, guaranteeing an appealing look on every device type, even though it is less adaptable than responsive design. Flexible grid systems are essential for maximising both layout and content. Grid systems that adjust to different screen sizes make it possible to arrange content in a logical and beautiful way. Because of this, interface designers may easily alter their designs without compromising their desired appearance or feel.

Also, Scalable images are also essential for preserving visual appeal on a variety of screens. Images can be made to render correctly and load quickly on a variety of devices by using scalability-supporting image formats, responsive image approaches like ‘srcset’ and sizes attributes in HTML, or CSS to adjust image proportions. All this can be achieved by using a CSS framework to streamline the building process. Tailwind CSS is a useful tool for developers who want to leverage utility classes and a modular styling approach to create user interfaces that are flexible, customisable, and efficient.

5.4 Limitations

A number of technical difficulties surfaced during the implementation process, especially when integrating the OpenAI model into the application. It was difficult for me to comprehend and integrate the chatbot because of my lack of knowledge with AI tools. But researching into the documentation and carrying out in-depth reviews helped navigate through this obstacle. Further attention to detail was needed when integrating the Stripe API for payment processing, which resulted in additional complexity that were managed in a similar way—by depending on documentation and fixing bugs that were faced during implementation.

Limitations in the assessment data during the evaluation phase could somehow have affected the accuracy of the findings. The survey had only 10 participants, and it is possible that the data collected for user feedback and interaction with the application might have been limited in terms of sample size or diversity. A small sample size may have an impact on how various user preferences or behaviours are represented, which could distort the evaluation's overall findings.

In addition to these technical limitations, the project's scope was also affected by time constraints. It was difficult to completely explore and implement more functions like using cloud service, hosting the application in a production environment, populating the database with more product catalogue etc proved difficult because of the limited timeframe available. The limitations encountered during the development process were also a result of scarce resources, both in terms of technical background and accessible experience which might have an impact on the scope and depth of the project’s features. However, efforts were made to lessen them through efficient time management and targeted use of available resources.

5.5 Ethical Considerations/Issues

Throughout the project, maintaining data privacy and confidentiality was of utmost importance. Participants in the survey were guaranteed total anonymity, preventing any personal identity from being connected to their answers. Participants were informed in advance through an information sheet about how their data will be handled, guaranteeing consent and transparency. Users were notified immediately that they were interacting with an AI-powered assistant, encouraging openness, and understanding on the nature of their exchanges.

Another ethical focus was addressing potential biases in AI interactions. To reduce bias and ensure fair and unbiased interactions with users, OpenAI continuously monitors and refines the training data used by their AI. Their dedication to impartiality and fairness is reinforced by the regular audits and assessments carried out to identify and address any biases that might be presented in the use of their models.

6. CONCLUSION AND FUTURE WORK

The implementation of responsive web design had a significant effect on customer satisfaction and engagement inside the e-commerce platform. The user experience was significantly improved by the visually appealing and consistent interface that catered to a variety of devices, confirming the effectiveness of putting such design concepts into practice. The survey highlighted the pleasant experiences that users had,

emphasising the visual coherence and strong aesthetic appeal of the application on various platforms.

Despite the project's success, there were technological difficulties, most notably when integrating the chatbot and the Stripe API, which required more time and debugging. The project scope was also constrained by technological difficulties and resource limitations, which provided valuable insights for future endeavours.

Looking ahead, the potential enhancements for the chatbot are promising. Enhancing user interactions might be further increased by giving the chatbot the ability to refer consumers to other websites and handle the sales process for products that are not sold in the shop. Moreover, incorporating voice prompts into the chatbot's functionality in addition to text inputs is in line with the survey's recommendations, guaranteeing a more flexible and easily navigable user interface. Subsequent research endeavours may delve more deeply into the optimisation of content and layout for diverse screens, investigating innovative design patterns to enhance user experiences even more. For the application to continue developing and provide a strong and complete e-commerce platform, it is essential to increase its functionality and strengthen its security measures.

Continued refinement of the chatbot's abilities based on user feedback will be essential in promoting user engagement and satisfaction. These advancements correspond with user demands and changing technology patterns, which in the end enhance the user experience in the e-commerce domain.

7. ACKNOWLEDGMENT

I express my gratitude to God Almighty for bestowing upon me the knowledge and comprehension necessary to complete this project. I also want to take this opportunity to thank Dr. Muhammad Zahid Iqbal, my academic supervisor, for all his help, advice, and inspiring ideas during the project's implementation. Finally, I want to express my gratitude to my family for their unwavering support in my professional development.

8. REFERENCES

[1] E. S. A. Majid, N. Kamaruddin and Z. Mansor (2015) 'Adaptation of usability principles in responsive web design technique for e-commerce development', 2015 International Conference on Electrical Engineering and Informatics (ICEEI). Available at: 10.1109/ICEEI.2015.7352593.

[2] R. A. Sukamto, Y. Wibisono and D. G. Agitya (2020) 'Enhancing The User Experience of Portal Website using User-Centered Design Method', 2020 6th International Conference on Science in Information Technology (ICSITech). Available at: 10.1109/ICSITech49800.2020.9392044.

[3] W. Hasim, S. Wibirama and H. A. Nugroho (2019) 'Redesign of E-Participation using User-Centered Design Approach for Improving User Experience', 2019 International Conference on Information and Communications Technology (ICOIACT). Available at: 10.1109/ICOIACT46704.2019.8938545.

[4] K. Gustiana Sugosha, R. Andreswari and M. Hardiyanti (2021) 'Design and Implementation of User Interface and User Experience in Online Sales Applications At Sugosha Pharmacy With User Centered Design Method', 2021 International Conference on Advanced Computer Science and Information Systems (ICACSIS). Available at: 10.1109/ICACSIS53237.2021.9631344.

[5] H. T. Maulana and U. Rosyidah (2022) 'User Experience Model using Concise User- Centered Design in Small and Medium Enterprise E-Commerce', 2022 International Seminar on Application for Technology of Information and Communication (iSemantic). Available at: 10.1109/iSemantic55962.2022.9920381.

[6] S. Hadzhikoleva, T. Rachovski and E. Hadzhikolev (2018) 'Generalized Net Model for Building Responsive Design of Web Pages', 2018 20th International Symposium on Electrical Apparatus and Technologies (SIELA). Available at: 10.1109/SIELA.2018.8447100.

A. Pinandito et al. (2017) 'Analysis of web content delivery effectiveness and efficiency in responsive web design using material design guidelines and User Centered Design', 2017 International Conference on Sustainable Information Engineering and Technology (SIET). Available at: 10.1109/SIET.2017.8304178.

[7] Wei Jiang et al. (2014) 'Responsive web design mode and application', 2014 IEEE Workshop on Advanced Research and Technology in Industry Applications (WARTIA). Available at: 10.1109/WARTIA.2014.6976522.

[8] S. Mohorovičić (2013) 'Implementing responsive web design for enhanced web presence', 2013 36th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO).

[9] J. Lee et al. (2015) 'Responsive Web Design According to the Resolution', 2015 8th International Conference on u- and e-Service, Science and Technology (UNESST). Available at: 10.1109/UNESST.2015.11.

[10] N. Li and B. Zhang (2019) 'The Design and Implementation of Responsive Web Page Based on HTML5 and CSS3', 2019 International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI). Available at: 10.1109/MLBDBI48998.2019.00084.

[11] J. -P. Voutilainen, J. Salonen and T. Mikkonen (2015) 'On the Design of a Responsive User Interface for a Multi-device Web Service', 2015 2nd ACM International Conference on Mobile Software Engineering and Systems. Available at: 10.1109/MobileSoft.2015.16.

[12] Sudiana, Y. U. Chandra and L. Angela (2021) 'Key Success Factors for a Better User Experience in E-Commerce Website', 2021 International Conference on Information Management and Technology (ICIMTech). Available at: 10.1109/ICIMTech53080.2021.9535076.

[13] Y. Cheng and B. Dong (2016) 'Analysis and design of campus E-commerce system', 2016 International Conference on Logistics, Informatics and Service Sciences (LISS). Available at: 10.1109/LISS.2016.7854564.

[14] Chan, F.K.Y. and Thong, J.Y.L. (2009) 'Acceptance of agile methodologies: A critical review and conceptual framework', Decision Support Systems, 46(4), pp. 803-814. Available at: https://doi.org/10.1016/j.dss.2008.11.009

[15] C. Matthies (2018) 'Scrum2Kanban: Integrating Kanban and Scrum in a University Software Engineering Capstone Course', 2018 IEEE/ACM International Workshop on Software Engineering Education for Millennials (SEEM).

[16] S. Gupta and N. Gayathri (2022) 'Study of the Software Development Life Cycle and the Function of Testing', 2022 International Interdisciplinary Humanitarian Conference for Sustainability (IIHC). Available at: 10.1109/IIHC55949.2022.10060231.

[17] L. Liang et al. (2017) 'Express supervision system based on NodeJS and MongoDB', 2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS). Available at: 10.1109/ICIS.2017.7960064.

[18] Dave Westerveld (2021) API Testing and Development with Postman: A practical guide to creating, testing, and managing APIs for automated software testing.