**INTERNATIONAL UNIVERSITY - VIETNAM NATIONAL UNIVERSITY**

**SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**

A blue and white logo

Description automatically generated

**PROJECT REPORT**

**Course: WEB APPLICATION DEVELOPMENT**

**Lecturer: Assoc. Prof. Nguyen Van Sinh**

**Lab instructor: Msc. Nguyen Trung Nghia**

**Group: Kuronami - Topic:** E-commerce

| **Name** | **ID** | **Contribution** |
| --- | --- | --- |
| Nguyễn Trần Duy Thức | ITITIU20318 | 25% |
| Nguyễn Trần Quốc Thịnh | ITITIU20312 | 25% |
| Nguyễn Xuân Sơn | ITITIU20295 | 25% |
| Nguyễn Công Sáng | ITITIU20292 | 25% |

Table of Contents

[1.](#_30j0zll) Executive Summary 7

[1.1.](#_1fob9te) Project Overview 7

[1.2.](#_3znysh7) Project Objectives and Scope 7

[1.3.](#_2et92p0) Key Achievements and Milestones 7

[1.4.](#_tyjcwt) Summary of Final Project 7

[1.5.](#_3dy6vkm) Outstanding Issues and Recommendations 8

[2.](#_1t3h5sf) Introduction 8

[2.1.](#_4d34og8) Background and Context of Project 8

[2.2.](#_2s8eyo1) Project Goals and Objectives 8

[2.3.](#_17dp8vu) Importance and Relevance of the Project 9

[2.4.](#_3rdcrjn) Brief Overview of the Report’s Structure 10

[3.](#_26in1rg) Project Planning 11

[3.1.](#_lnxbz9) Project Timeline and Milestones 11

[3.2.](#_1ksv4uv) Resource Allocation 12

[3.3 Risk Assessment and Mitigation Strategies 13](#_z337ya)

[4.](#_1y810tw) Requirements Analysis 16

[4.1.](#_4i7ojhp) Detailed description of project requirements 16

[4.2.](#_1ci93xb) User Stories, Use Cases, or Functional Specifications: 17

[4.2.1.](#_3whwml4) User Stories and Use Cases 17

[4.2.2.](#_1v1yuxt) Functional Specifications 26

[4.3.](#_2u6wntf) Changes and Updates to Technologies Throughout the Project 27

[5.](#_19c6y18) Design and Architecture 28

[5.1.](#_3tbugp1) Overview of the System Architecture 28

[5.2.](#_nmf14n) Diagrams illustrating system components. 30

[5.3.](#_2zbgiuw) Key Decision and Justification 38

[6.](#_1egqt2p) Development 40

[6.1.](#_3ygebqi) Description of the Development Process 40

[6.2.](#_2dlolyb) Programming languages, frameworks, and tools used. 42

[6.3.](#_kgcv8k) Challenges encountered during development. 53

[6.4.](#_34g0dwd) Code Management and version control 54

[7.](#_43ky6rz) Testing and Quality Assurance 55

8[.](#_upglbi) User Documentation 66

9[.](#_3x8tuzt) Project Evaluation 82

9[.1.](#_2ce457m) Evaluation Criteria Used to Measure Project Success: 82

9[.2.](#_rjefff) Assessment of Whether Project Goals Were Achieved: 83

9[.3.](#_3bj1y38) User Feedback and Satisfaction: 83

[10.](#_1qoc8b1) Lesson Learned 84

[10.1.](#_4anzqyu) Reflection on What Went Well: 85

[10.2.](#_2pta16n) Insights Gained from Project Challenges: 85

[10.3.](#_14ykbeg) Recommendations for Future Projects: 85

[11.](#_3oy7u29) Conclusion 86

[11.1.](#_243i4a2) Summary of Project Outcomes: 86

[11.2.](#_j8sehv) Reiteration of Key Achievements: 86

[11.3.](#_338fx5o) Closing Remarks: 87

[References](#_1idq7dh) 87

List of Figures

[Figure 4-1 Work Breakdown Structure](#_2xcytpi) 16

[Figure 4-2 Use Case Diagram](#_2bn6wsx) 17

[Figure 5-1 MVC Model](#_28h4qwu) 29

[Figure 5-2 Login Sequence Diagram](#_37m2jsg) 31

[Figure 5-3 Register Sequence Diagram](#_1mrcu09) 32

[Figure 5-4 View Order Sequence Diagram](#_46r0co2) 33

[Figure 5-5 Checkout Sequence Diagram](#_2lwamvv) 34

[Figure 5-6 Add to cart Sequence Diagram](#_111kx3o) 35

[Figure 5-7 Remove from Cart Sequence Diagram](#_3l18frh) 36

[Figure 5-8 Reduce from Cart Sequence Diagram](#_206ipza) 37

[Figure 5-9 Admin features Sequence Diagram](#_4k668n3) 38

[Figure 6-1 Visual Studio Code](#_sqyw64) 42

[Figure 6-2 Intellij IDEA](https://docs.google.com/document/d/1qzA_WcmAenMI5m_K_iVFRL0CCuV3rdl9xY8_IpQ90uw/edit?fbclid=IwZXh0bgNhZW0CMTAAAR0CSvoNXiVT_CYZiW3knw-d00q1pRWk4Pj9aXpTbfy_6_4zuexHxTGC_iE_aem_AfbUJPfF5vQzx_FrMNv8cURWRpIEI89c9gD-sMw-WqdHL2jAQrbuyUTgLYZ3qRcdDXrU3Hd3xlqwdwQ9eDFWYn-j#heading=h.sqyw64) 43

[Figure 6-3 HTML-CSS-JS-JAVA 4](#_3cqmetx)4

[Figure 6-4 React JS 4](#_1rvwp1q)5

[Figure 6-5 Spring boot 4](https://docs.google.com/document/d/1qzA_WcmAenMI5m_K_iVFRL0CCuV3rdl9xY8_IpQ90uw/edit?fbclid=IwZXh0bgNhZW0CMTAAAR0CSvoNXiVT_CYZiW3knw-d00q1pRWk4Pj9aXpTbfy_6_4zuexHxTGC_iE_aem_AfbUJPfF5vQzx_FrMNv8cURWRpIEI89c9gD-sMw-WqdHL2jAQrbuyUTgLYZ3qRcdDXrU3Hd3xlqwdwQ9eDFWYn-j#heading=h.1rvwp1q)6

[Figure 6-6 Vite JS 4](https://docs.google.com/document/d/1qzA_WcmAenMI5m_K_iVFRL0CCuV3rdl9xY8_IpQ90uw/edit?fbclid=IwZXh0bgNhZW0CMTAAAR0CSvoNXiVT_CYZiW3knw-d00q1pRWk4Pj9aXpTbfy_6_4zuexHxTGC_iE_aem_AfbUJPfF5vQzx_FrMNv8cURWRpIEI89c9gD-sMw-WqdHL2jAQrbuyUTgLYZ3qRcdDXrU3Hd3xlqwdwQ9eDFWYn-j#heading=h.sqyw64)7

Figure 6-7 MySQL [4](#_4bvk7pj)8

[Figure 6-8 DB schema 49](#_2r0uhxc)

[Figure 6-9 Canva 51](#_1664s55)

[Figure 6-10 Discord 52](#_3q5sasy)

[Figure 6-11 Google Chrome 53](#_25b2l0r)

[Figure 6-12 Git and Github](#_1jlao46) 55

[Figure 8-1 Homepage Interface 7](#_3ep43zb)2

[Figure 8-2 Signup 1 76](#_1tuee74)

[Figure 8-3 Signup 2 7](#_4du1wux)4

[Figure 8-4 Signup 3 7](#_2szc72q)4

[Figure 8-5 Sign in 1 7](#_184mhaj)5

[Figure 8-6 Sign in 2 7](#_3s49zyc)5

[Figure 8-7 Sign in 3 7](#_279ka65)6

[Figure 8-8 User Details 7](#_meukdy)7

[Figure 8-9 Search 7](#_36ei31r)8

[Figure 8-10 Product Details](#_1ljsd9k) 79

[Figure 8-11 Product Purchase 8](#_45jfvxd)0

[Figure 8-12 Order Product 8](#_2koq656)1

[Figure 8-13 FAQ 1 8](#_2y3w247)1

List of Tables

[Table 3-1 Hardware Resource 12](#_44sinio)

[Table 3-2 Software Resource 13](#_2jxsxqh)

[Table 3-3 Risk Assessment and Mitigation 16](#_3j2qqm3)

[Table 4-1 Use Case 001](#_qsh70q) 18

[Table 4-2 Use Case 002](#_3as4poj) 19

[Table 4-3 Use Case 003](#_1pxezwc) 20

[Table 4-4 Use Case 004](#_49x2ik5) 21

[Table 4-5 Use Case 005 22](#_2p2csry)

[Table 4-6 Use Case 006 23](#_147n2zr)

[Table 4-7 Use Case 007 24](#_3o7alnk)

[Table 4-8 Use Case 008](#_23ckvvd) 24

[Table 4-9 Use Case 009](#_ihv636) 25

[Table 4-10 Use Case 010](#_32hioqz)26

[Table 4-11 Functional and Non-Functional Requirements](#_4f1mdlm) 27

[Table 7-1 Test Case 001 55](#_2iq8gzs)

[Table 7-2 Test Case 002](#_xvir7l) 57

[Table 7-3 Test Case 003 5](#_3hv69ve)8

[Table 7-4 Test Case 004 5](#_1x0gk37)9

[Table 7-5 Test Case 005 59](#_4h042r0)

[Table 7-6 Test Case 006](#_2w5ecyt) 60

[Table 7-7 Test Case 007 60](#_1baon6m)

[Table 7-8 Test Case 008 61](#_3vac5uf)

[Table 7-9 Test Case 009 62](#_2afmg28)

[Table 7-10 Test Case 010 63](#_pkwqa1)

[Table 7-11 Test Case 011 6](#_39kk8xu)5

# Executive Summary

## Project Overview

Our project's core objective was to develop a comprehensive e-commerce website tailored for a retail store, complemented by a robust admin panel. The website aimed to significantly boost the store's online presence by providing customers with an intuitive interface to explore products, make purchases, and engage seamlessly with the store's offerings. Simultaneously, the admin panel was designed to empower store owners with powerful management tools, streamlining operational processes for enhanced efficiency.

## Project Objectives and Scope

Our primary focus encompassed crafting a responsive and visually captivating website equipped with intuitive navigation and an interactive product catalog. The admin panel's development aimed to deliver extensive functionalities, enabling store owners to effortlessly manage inventory, monitor sales metrics, and update content dynamically. Our scope included comprehensive front-end and back-end development, seamless database integration, and the implementation of robust security measures to safeguard customer information and transactions.

## Key Achievements and Milestones

Throughout the project lifecycle, we achieved several pivotal milestones:

* **Design and Planning:** Successfully completed website wireframes and design mock-ups.
* **Responsive Design:** Implemented a responsive web design ensuring an optimal user experience across all devices.
* **Admin Panel Development:** Created and deployed a fully functional admin panel featuring user-friendly interfaces.
* **Quality Assurance:** Conducted rigorous testing phases to identify and resolve potential issues, ensuring a reliable platform.

## Summary of Final Project

The culmination of our efforts resulted in the successful launch of a fully operational ecommerce website alongside an intuitive admin panel. The website boasts a visually appealing design, seamless navigation, and robust security features, significantly enhancing the online shopping experience for customers. The admin panel equips store owners with efficient tools for managing products, orders, and customer interactions, thereby optimizing business operations.

## Outstanding Issues and Recommendations

While the project achieved its primary objectives, ongoing attention is essential to address outstanding issues and areas for improvement. We recommend continuous monitoring and periodic updates to tackle evolving security challenges, integrate new features, and enhance overall system performance. Additionally, gathering user feedback and conducting usability testing will provide valuable insights for refining and optimizing the platform further.

The successful execution of this project has delivered a robust and functional ecommerce solution, empowering store owners with an efficient admin panel. Our commitment to ongoing improvements in security, user experience, and technological advancements ensures the sustained success and relevance of the platform in a competitive market landscape.

# Introduction

## Background and Context of Project

The landscape of e-commerce has witnessed significant growth, playing a pivotal role in the retail industry. Platforms like Shopee, Lazada, and Tiki in Vietnam exemplify the transformative impact of e-commerce, enabling businesses to reduce costs, expand customer reach, and enhance operational efficiency. For consumers, e-commerce offers unparalleled convenience, enabling price comparisons, extensive product choices, and customer reviews, thereby revolutionizing shopping experiences.

In today's competitive business environment, establishing a robust sales website is crucial for success. It not only meets market demands but also enhances brand reputation, collects valuable consumer data, and reduces advertising costs. A well-executed e-commerce platform is pivotal in driving business growth and maintaining relevance in the dynamic market.

## Project Goals and Objectives

**Project Goals**

* Develop an e-commerce website and admin panel to bolster the store's online presence and operational efficiency.
* Ensure the website is user-friendly, visually appealing, and equipped with seamless navigation and purchasing options.
* Integrate an admin panel to empower store owners with comprehensive tools for efficient business management.
* Create a unified experience for all stakeholders involved, from customers to administrators.

**Project Objectives**

The project aims to create a complete e-commerce website for a specific shop with customer-facing features and an admin panel for efficient store administration. Product browsing, comprehensive product information, user accounts, shopping cart, secure payment, order tracking, and a responsive admin dashboard for product, order, customer, and inventory management will all be available on the website.

* **Customer Experience:** Provide consumers with an intuitive and user-friendly interface that ensures easy navigation, detailed product information, and a smooth shopping experience from product selection to checkout.
* **Admin Efficiency:** Give business owners/admins comprehensive tools for managing items, orders, customers, and inventories. Streamline administrative chores to increase productivity and decision-making effectiveness.
* **Security:** Establish strong security measures, such as secure payment gateways and data encryption, to protect consumer data and prevent illegal access.
* **Scalability:** Create a scalable website design and database structure to enable future growth and accommodate additional traffic and goods.
* **Performance:** Enhance the user experience by optimizing website performance to ensure quick loading times, device responsiveness, and minimal downtime.

## Importance and Relevance of the Project

**Importance of the Project:**

* **Market Expansion:** Facilitates global market reach and attracts a diverse customer base.
* **Competitive Edge:** Provides a competitive advantage through tailored online features and efficient management tools.

**Relevance of the Project:**

* **Consumer Behavior:** Aligns with evolving consumer preferences by delivering a seamless online shopping experience.
* **Future-Proofing:** Future-proofs operations by adapting to emerging technologies and market trends.

## Brief Overview of the Report’s Structure

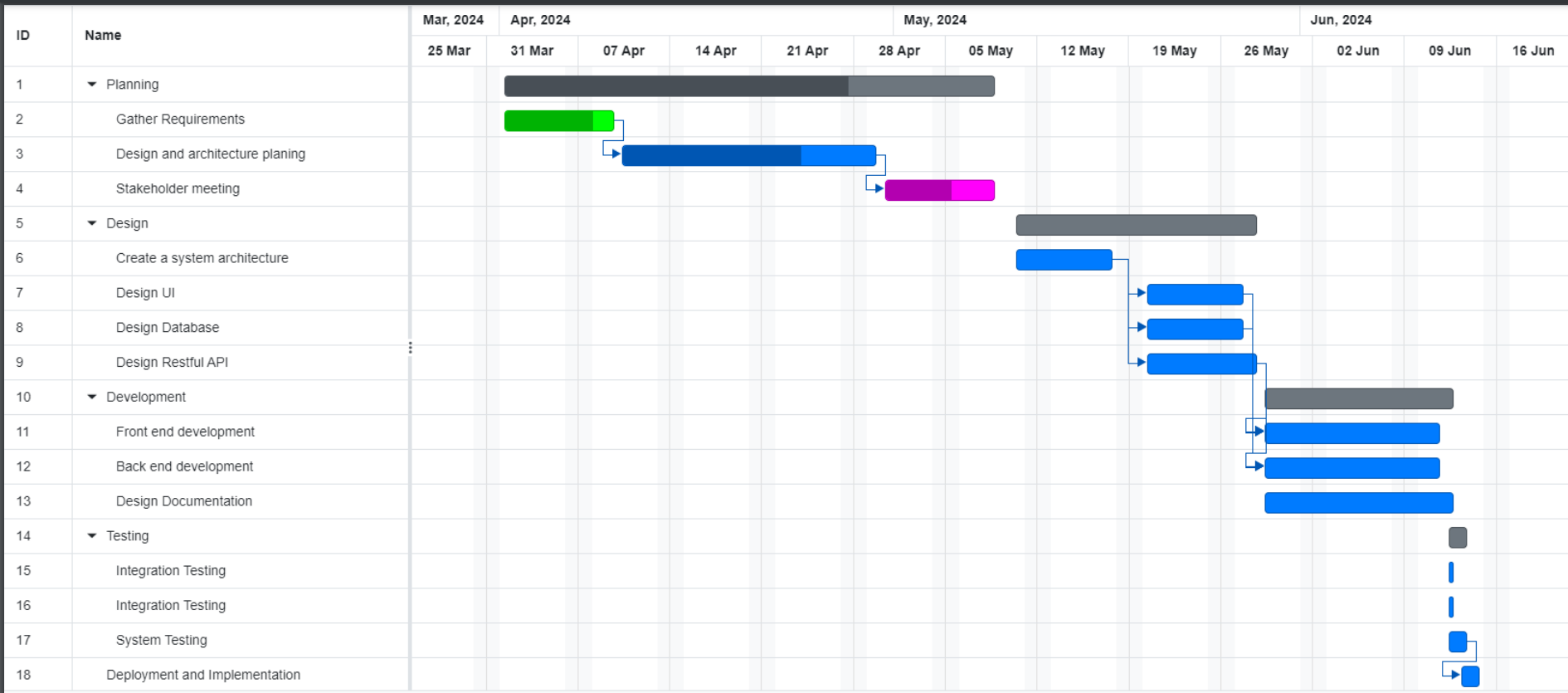
This report is structured to show a thorough walkthrough of the project. Following this introduction, the subsequent sections include phase 1 of the project, detailing the concise planning phase of the project, its requirements analysis, its design, and its architecture. The report reviews the project's development, testing and quality assurance, deployment, and implementation. The results and findings are then presented, discussing any challenges encountered and recommendations for future enhancements. The report concludes with an evaluation and its learned lessons for the broader knowledge of web development.

The report will be organized in the following structure:

* **Chapter 1:** Executive Summary
* **Chapter 2:** Introduction
* **Chapter 3:** Project Planning
* **Chapter 4:** Requirement Analysis
* **Chapter 5:** Design and Architecture
* **Chapter 6:** Development
* **Chapter 7:** Testing and Quality Assurance
* **Chapter 8:** User Documentation
* **Chapter 9:** Project Evaluation
* **Chapter 10:** Lesson Learned
* **Chapter 11:** Conclusion
* **Reference**

# Project Planning

## Project Timeline and Milestones



*Figure 3-1 Gantt Chart*

Our project management approach embodies a decentralized structure designed to optimize efficiency and ensure comprehensive oversight across all project phases. With a dedicated Project Manager leading the initiative, our team of eight members is strategically organized into specialized groups, each contributing uniquely to project success.

1. **Project Manager:**
   * **Role and Responsibilities:** The Project Manager assumes a pivotal role in overseeing the entire project lifecycle. Responsibilities include strategic planning, resource allocation, stakeholder communication, and ensuring cohesive project execution. They serve as the central point of contact for all project-related matters, fostering collaboration and maintaining project coherence from inception to delivery.
2. **Web Development Team (Group of 2):**
   * **FrontEnd Responsibilities:** This team specializes in transforming design concepts into fully functional web interfaces. They focus on creating intuitive user experiences, implementing secure payment gateways, and ensuring seamless website functionality across different devices and browsers.
   * **BackEnd Responsibilities:** Focused on server-side operations, database management, and API integrations, this group ensures the robustness and scalability of the system. They collaborate closely with the FrontEnd team to integrate design elements seamlessly into the backend architecture, prioritizing performance optimization and data security.
3. **Database Administration and Quality Assurance (Group of 2):**
   * **Roles and Responsibilities:** This group manages database integration, ensuring data integrity and implementing stringent security protocols. They play a critical role in conducting comprehensive quality assurance testing to identify and resolve issues related to functionality, performance, and security. Their efforts are crucial in delivering a reliable and secure final product to our clients.

## Resource Allocation

Effective resource allocation is critical for the successful execution of our project, ensuring we have the necessary tools, budget, and personnel to achieve our objectives efficiently. This section details our comprehensive plan for resource allocation, encompassing budgetary considerations, software and hardware requirements, and the strategic management of resources to optimize productivity and collaboration.

**Hardware Resource**

| Items | Quantity | Description | Purpose | Price | Total Cost | Date Needed |
| --- | --- | --- | --- | --- | --- | --- |
| ASUS ROG | 2 | Development Devices | Team developmment and Management | $1,500/device | $3,000 | 5th May |
| Laptop gaming ASUS ROG Zephyrus G16 | 2 | Design Devices | Team Design | $2,000/device | $4,000 | 5th May |
| Internet Connection |  |  |  | $100/month | $200 | 5th May |
| Total cost: 14,000$ | | | | | | |

*Table 3-1 Hardware Resource*

**Software Resource**

| Application | Quantity | Description | Purpose | Price | Total Cost | Date Needed |
| --- | --- | --- | --- | --- | --- | --- |
| VisualCode | 4 | Design frontend tool | Development Phase | free | $0 |  |
| IntelliJ | 4 | Design backend tool | Design Phase | free | $0 |  |
| MySQL | 4 | Store database | Deployment Phase | free | $0 |  |
| Web Storm | 4 | Design frontend tool | Deployment Phase | free | $0 |  |
|  |  |  |  |  |  |  |
| Total cost: $0 | | | | | | |

*Table 3-2 Software Resource*

## 3.3 Risk Assessment and Mitigation Strategies

Risk management is paramount in ensuring the smooth execution and success of our project. This section presents a comprehensive analysis of potential risks across various domains and outlines proactive strategies to mitigate their impact, thereby safeguarding project continuity and achieving our objectives effectively.

| Risk ID | ﻿Risk Category | ﻿Risk Title | ﻿Affect | ﻿Probability | ﻿Impact | ﻿Risk response plan |
| --- | --- | --- | --- | --- | --- | --- |
| 001 | Human Resources | Inadequate Workforce Skills | Reduced productivity, errors, increased need for supervision | High | Medium | Implement regular training programs, conduct skills assessments, collaborate with educational institutions for skill development. |
| 002 | Human Resources | Legal Compliance Issues | Legal fines, damage to reputation, potential lawsuits | High | Low | Stay updated with employment laws, maintain clear HR policies, conduct regular compliance audits, seek legal counsel as needed. |
| 003 | Requirements | Unclear Project Requirements | Misaligned project deliverables, rework due to misunderstanding requirements, project completion delays, increased expenses, disgruntled stakeholders, and probable project failure. | Moderate | High | Conduct thorough requirement-gathering sessions, involve stakeholders, use tools like Use Cases and Traceability Matrix. |
| 004 | Requirements | Scope Creep | Project delays, increased costs, strained resources, potential decrease in project quality, frustrated team members, and dissatisfaction among stakeholders. | Moderate | High | Define and document clear project scope, establish change control procedures, ensure stakeholder alignment. |
| 005 | Organization | Poor Communication | Misunderstandings, conflicts among team members, delayed decision-making, lack of alignment, decreased team collaboration, and potential project scope misunderstandings. | High | Medium | Establish clear communication channels, encourage open dialogue, hold regular team meetings, provide communication training. |
| 006 | Organization | Lack of Cross-Functional Collaboration | Limited knowledge sharing, lack of diverse perspectives, missed opportunities for innovation, potential inefficiencies in processes, and reduced adaptability to change. | Moderate | Medium | Foster cross-functional collaboration, form interdisciplinary teams, organize regular knowledge-sharing sessions. |
| 007 | Management | Ineffective Leadership | Decreased team motivation, lack of direction, poor decision-making, higher turnover rates, reduced employee engagement, and potential lack of strategic vision. | Moderate | High | Provide leadership training, mentorship programs, and 360-degree feedback for managers. Encourage a leadership style that promotes employee growth, motivation, and productivity. |
| 008 | Management | Financial Mismanagement | Budget overruns, decreased project funding for critical activities, strained supplier relationships, potential project shutdown, inability to meet financial commitments, damage to the company's financial stability and credibility | Low | High | Implement strict financial controls, conduct regular audits, involve key stakeholders in financial decision-making processes. |

*Table 3-3 Risk Assessment and Mitigation*

# Requirements Analysis

## Detailed description of project requirements

Our project aims to develop a robust online platform that integrates a fully functional store website with an intuitive admin panel. This platform is designed to enhance the digital presence of the store while providing effective tools for managing operations.

A diagram of a company

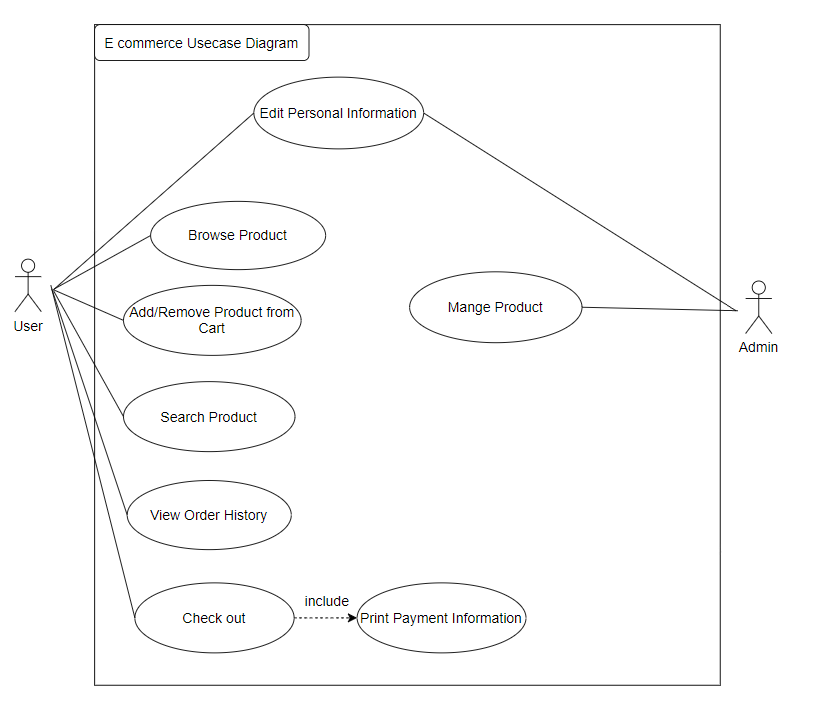
Description automatically generated

*Figure 4-1 Work Breakdown Structure*

## User Stories, Use Cases, or Functional Specifications:

### User Stories and Use Cases

Use cases provide a comprehensive exploration of various scenarios, detailing interactions between different actors, such as customers and store owners, and the system itself. For our project, use cases will outline the step-by-step processes involved in specific actions, such as order fulfillment, inventory management, and customer support. Use cases serve as a roadmap for the development team, guiding the implementation of features that contribute to a seamless and efficient online retail environment.



*Figure 4-2 Use Case Diagram*

**User**

**1.** **Login**

Title: User Account Authentication

Identifier: UC-001

User Story: As a customer, I want to create a new account or log in to my existing account so that I can access my personalized information and view my shopping history on the website.

Input/Output: Input**:** Username, password ,Output**:** User's account dashboard

Preconditions: The user is on the website's login page.

Basic Course:

| **Step** | **User Action** | **System Response** |
| --- | --- | --- |
| 1 | Users enter their username and password | - |
| 2 | - | The system verifies the user's credentials. |
| 3 | - | If valid, the user is directed to their account dashboard. If the credentials are invalid, the system displays an error message. |

*Table 4-1 Use Case 001*

Alternative Course:

Postconditions: The user is logged in and can access their account information.

**2.** **Browse Products**

Title: Product Browsing

Identifier: UC-002

User Story: As a customer, I want to browse the products available on the website to view details such as product name, price, description, and images.

Input/Output: Input: Product category; Output: List of products with details.

Preconditions: The user is on the website.

Basic Course:

| **Step** | **User Action** | **System Response** |
| --- | --- | --- |
| 1 | The user selects a product category or views all products | - |
| 2 | - | The system displays a list of products with details. |

*Table 4-2 Use Case 002*

Postconditions: The user can view product details.

**3.** **Search Products**

Title: Product Search

Identifier: UC-003

User Story: As a consumer, I would want to search for certain items using keywords, filters, or other search parameters.

Input/Output: Input: Search criteria; Output: List of matching products.

Preconditions: The user is on the website.

Basic Course:

| **Step** | **User Action** | **System Response** |
| --- | --- | --- |
| 1 | The user enters the search criteria | - |
| 2 | - | The system displays a list of products matching the criteria. |

*Table 4-3 Use Case 003*

Postconditions: The user can see the list of products that match their search.

**4.** **Add Product to Cart**

Title: Add Product to Cart

Identifier: UC-004

User Story: As a customer, I want to add one or more products to my shopping cart for purchase.

Input/Output: Input: Selected product(s); Output: Updated shopping cart.

Preconditions: The user is logged in and viewing a product.

Basic Course:

| **Step** | **User Action** | **System Response** |
| --- | --- | --- |
| 1 | The user selects a product and adds it to the cart | - |
| 2 | - | The system updates the shopping cart. |

*Table 4-4 Use Case 004*

Postconditions: The selected product(s) are in the user's shopping cart.

**5.** **Remove Product from Cart**

Title: Remove Product from Cart

Identifier: UC-005

User Story: As a customer, I want to remove one or more products from my shopping cart if I change my mind or decide not to make a purchase.

Input/Output: Input: Selected product(s); Output: Updated shopping cart.

Preconditions: The user is logged in and has products in the shopping cart.

Basic Course:

| **Step** | **User Action** | **System Response** |
| --- | --- | --- |
| 1 | The user selects product(s) for removal | - |
| 2 | - | The system updates the shopping cart by removing the selected product(s). |

*Table 4-5 Use Case 005*

Postconditions: The selected product(s) are removed from the shopping cart.

**6.** **Update Cart**

Title: Update Shopping Cart

Identifier: UC-006

User Story: As a customer, I want to update the quantity of products in my cart, apply coupons or discounts, and view the total cost of my order.

Input/Output: Input: Quantity, coupons, discounts; Output: Updated shopping cart and order summary.

Preconditions: The user is logged in and has products in the shopping cart.

Basic Course:

| **Step** | **User Action** | **System Response** |
| --- | --- | --- |
| 1 | The user adjusts the number of products, applies coupons, or adds discounts | - |
| 2 | - | The system updates the shopping cart and order summary. |

*Table 4-6 Use Case 006*

Postconditions: The shopping cart and order summary are updated.

**7.** **Checkout**

Title: Checkout

Identifier: UC-007

User Story: As a customer, I want to proceed to checkout, enter my shipping and payment information, review my order summary, and complete the purchase.

Input/Output: Input: Shipping and payment information; Output: Order confirmation.

Preconditions: The user is logged in and has products in the shopping cart.

| **Step** | **User Action** | **System Response** |
| --- | --- | --- |
| 1 | The user proceeds to checkout | - |
| 2 | The user enters shipping and payment information | - |
| 3 | - | The system displays an order summary. |
| 4 | The user confirms the order | - |

*Table 4-7 Use Case 007*

Postconditions: The order is placed, and the user receives an order confirmation.

**8.** **View Order History**

Title: View Order History

Identifier: UC-008

User Story: As a customer, I want to view my order history to see a list of my previous purchases, the date and time of the purchase, and the status of the order.

Input/Output: Input: None; Output: List of previous orders.

Preconditions: The user is logged in.

| **Step** | **User Action** | **System Response** |
| --- | --- | --- |
| 1 | The user navigates to the order history page | - |
| 2 | - | The system displays a list of previous orders with details. |

*Table 4-8 Use Case 008*

Postconditions: The user can view their order history.

**Admin**

**1.** **Manage Products**

Title: Product Management

Identifier: UC-009

User Story: As an admin, I want to manage the products offered for sale, including adding new products, editing product details, and removing products from inventory.

Input/Output: Input: Product information; Output: Updated product inventory.

Preconditions: The admin is logged in.

Basic Course:

| **Step** | **User Action** | **System Response** |
| --- | --- | --- |
| 1 | Admin adds, edits, or removes products | - |
| 2 | - | The system updates the product inventory. |

*Table 4-9 Use Case 009*

Postconditions: The product inventory is updated.

**2.** **Login (Admin)**

Title: Admin Account Authentication

Identifier: UC-010

User Story: As an admin, I want to log in to a special account with administrative privileges to manage the website.

Input/Output: Input: Admin credentials; Output: Admin dashboard.

Preconditions: The admin is on the website's admin login page.

Basic Course:

| **Step** | **User Action** | **System Response** |
| --- | --- | --- |
| 1 | Admin enters their username and password | - |
| 2 | - | The system verifies the admin's credentials. |
| 3 | - | If valid, the admin is directed to the admin dashboard. |

*Table 4-10 Use Case 010*

Postconditions: The admin is logged in and can access the admin dashboard.

### Functional Specifications

These website requirements include topics such as product catalog management, user identification, secure payment processing, and responsive design. For the admin panel, functional requirements will specify how store owners may manage inventory, track sales, and change content. These specs serve as a blueprint for the development team, outlining exactly what needs to be developed and how each component should work.

| Req. ID | Requirement Name | Detailed Description | Type |
| --- | --- | --- | --- |
| 001 | Register an account and authenticate | If the user doesn’t have an account, he will be asked to register. Users can reset their password. | Functional requirement |
| 002 | Admin/User login and authenticate | Admin and User after registration will able to login to the website with their account | Functional requirement |
| 003 | Manage products | Allow admin/owner of the page to add/remove/update a product and display it to the consumer | Functional requirement |
| 004 | Manage orders | Allow admin/owner of the page to add/remove/update an order and display it to the consumer | Functional requirement |
| 005 | Product Listings | Users can view the products with description and images | Functional requirement |
| 006 | Shopping Cart | Users can add and remove items from their shopping cart | Functional requirement |
| 007 | Search Functionality | Users should be able to search for products by keywords and see relevant results. | Functional requirement |
| 008 | Security Requirement | User data, including personal information and payment details, must be stored securely and encrypted. | Non-functional Requirement |
| 009 | Protection against common web vulnerabilities such as SQL injection and Cross-Site Scripting (XSS) | Non-functional Requirement |
| 010 | Usability Requirements | The user interface is intuitive and easy to navigate, promoting a positive user experience. | Non-functional Requirement |
| 011 | Data Backup and Recovery | Regular backups of user and product data should be maintained, and a data recovery plan should be in place. | Non-functional Requirement |

*Table 4-11 Functional and Non-Functional Requirements*

## Changes and Updates to Technologies Throughout the Project

During our project, the team faced significant challenges with the initially chosen technologies. Given that our group consists of beginners, the initial technology stack presented a steeper learning curve than anticipated. Recognizing the importance of a manageable and effective development process, we made a strategic decision to transition to a more accessible technology stack.

Our revised technology stack includes:

* **Backend**: Spring Boot
* **Frontend**: Vite with React.js

This change aims to provide a more accessible and learning-friendly environment for the team. Spring Boot offers a streamlined approach to backend development, while Vite.js enhances the speed and simplicity of the web application framework. React.js is known for its ease of use and efficiency in building interactive user interfaces.

The new technology stack aligns well with our group's learning curve:

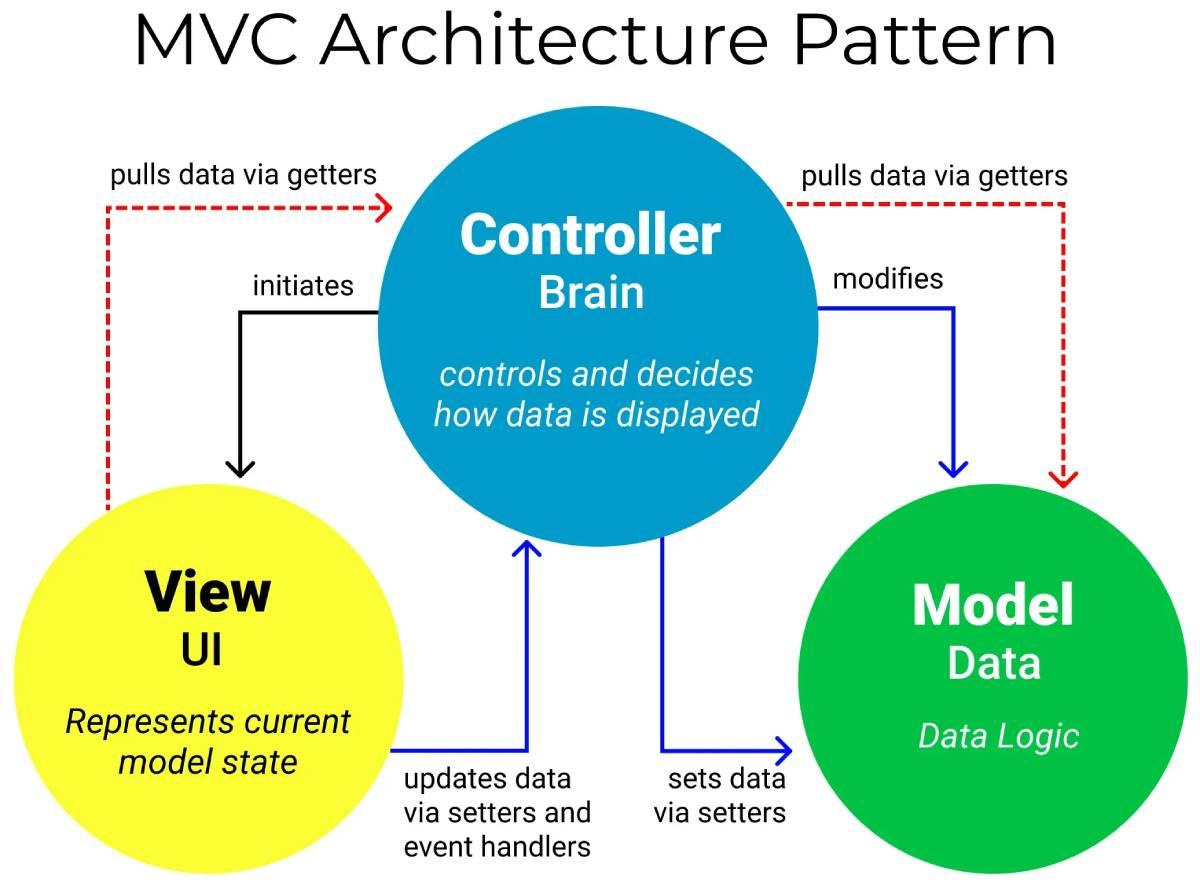
* **Spring Boot**: Well-suited for building robust backend services and simplifies many complexities associated with backend development.
* **Vite**: Offers a fast and modern build tool that significantly improves the development experience with its rapid hot module replacement (HMR) and minimal configuration.
* **React.js**: Provides a component-based architecture that facilitates the creation of interactive and dynamic user interfaces, making it ideal for building a feature-rich e-commerce.

This adjustment is comprehensively documented, outlining the motivations for the change, the impact on project timelines, and the strategies employed to ensure a smooth transition. Embracing this technology shift not only addresses the skill level of our team but also positions us for a more efficient and successful implementation of the website and admin panel. It reflects our commitment to adaptability and underscores the collaborative nature of our team as we navigate the complexities of technology implementation in a learning-intensive environment.

# Design and Architecture

## Overview of the System Architecture

Our project endeavors to build a comprehensive website for a store, complemented by an efficient admin panel for streamlined management. To achieve this, we have adopted the Model-View-Controller (MVC) architecture, known for its structured organization and facilitation of modular development practices.



*Figure 5-1 MVC Model*

**Model:**

The **Model** serves as the backbone of our system, handling core data management and business logic. It encompasses functionalities crucial to the store's operations, including product management, inventory tracking, and user data management. By interfacing with the database, the Model ensures reliable storage and retrieval of essential information. For instance, it manages updates to product quantities, monitors sales performance, and maintains user accounts securely.

**View:**

The **View** component focuses on user interface presentation across both the store website and admin panel. For customers visiting the store website, the View delivers intuitive interfaces for browsing products, exploring detailed product information, and completing purchases seamlessly. In contrast, the admin panel's View empowers store owners with tools to visualize sales analytics, update product catalogs efficiently, and manage user accounts effectively. Our emphasis lies in crafting visually appealing interfaces that enhance user experience and operational efficiency.

**Controller:**

Serving as the intermediary, the **Controller** manages user interactions and orchestrates data flow between the Model and View components. On the store website, it processes user actions such as product searches, cart management, and checkout processes. Within the admin panel, the Controller facilitates tasks such as inventory updates, report generation, and user permissions management. This role ensures smooth communication between different parts of the system, optimizing performance and maintaining consistency in user interactions.

By adopting the MVC architecture, our development approach promotes scalability and maintainability. Each component (Model, View, and Controller) operates independently, allowing for targeted updates and expansions without disrupting other system functionalities. This modular design enhances agility in adapting to evolving requirements and facilitates easier troubleshooting and enhancement of features over time.

Our commitment to leveraging MVC architecture underscores our dedication to delivering a robust, flexible, and user-centric platform that meets the dynamic needs of both customers and administrator

## Diagrams illustrating system components.

These diagrams aim to provide a clear visual representation of the system's architecture and the relationships between its components. Adjustments and additional details can be made based on the specific technologies and intricacies of the project.

**LOGIN**

**A picture containing text, diagram, parallel, plan

Description automatically generated**

*Figure 5-2 Login Sequence Diagram*

**Description:** The sequence diagram illustrates the interactions involved in the user login process. It shows how the server handles the rendering of the login form, validates user input, performs authentication, manages the user's shopping cart by saving it to their account or loading it into the session, and manages the redirection of the user to the appropriate page based on the outcome of the login process.

**REGISTER**  
A screenshot of a diagram

Description automatically generated

*Figure 5-3 Register Sequence Diagram*

**Description:** The sequence diagram illustrates the interactions involved in the user signup process. It shows how the server handles the rendering of the signup form, validates user input, creates a new user account, performs authentication, saves the shopping cart (if applicable), and manages the redirection of the user to the appropriate page based on the outcome of the signup process.

**VIEW ORDER HISTORY:**

A screenshot of a computer program

Description automatically generated

*Figure 5-4 View Order Sequence Diagram*

**Description:** The sequence diagram illustrates the process of a user viewing their order history on a website. The user initiates the process, and the view/template checks if the user is logged in. If the user is logged in, the user route queries the database for the user’s order history. The user model identifies the redirect page and responds with the order history. Finally, the view/template renders and displays the order history to the user. This diagram helps visualize the flow of data and interactions between different components in the process.

**CHECKOUT**

A black screen with white text

Description automatically generated

*Figure 5-5 Checkout Sequence Diagram*

**Description:** This sequence diagram provides a visual representation of the checkout process on an e-commerce website. It begins with the user clicking on the checkout button. If the user is not logged in, the view/template renders a login form and the user is redirected to the login page. If there’s an error in identifying the user, an error message is displayed. Once the user is logged in, the view/template renders and displays the payment information. The user route and user model handle the backend processes, including validating the user’s input and processing the order. The diagram effectively illustrates the flow of data and interactions between the user and the various components of the website during the checkout phase.

**ADD TO CART**

A screenshot of a computer

Description automatically generated

*Figure 5-6 Add to cart Sequence Diagram*

**Description:** The sequence diagram illustrates the process of adding a product to the cart in an E-Commerce website application using the MVC pattern.

* **User:** The user initiates the request by clicking the “Add to Cart" button on a product page.
* **Controller:** The controller receives the request and handles the business logic for adding the product to the cart.
* **Model:** The model represents the application's data, including the cart data.
* **View:** The view displays the user interface, including the shopping cart.

**REMOVE FROM CART**

A screenshot of a computer

Description automatically generated

*Figure 5-7 Remove from Cart Sequence Diagram*

**REDUCE FROM CART**

A screenshot of a computer

Description automatically generated

*Figure 5-8 Reduce from Cart Sequence Diagram*

**Description:** Sequence diagram for reducing and removing a product from the cart using the MVC pattern:

* **User:** The user initiates the request by clicking a "Remove/ Reduce from Cart" button on a product page.
* **Controller:** The controller receives the request and handles the business logic for reducing the product quantity in the cart.
* **Model:** The model represents the application's data, including the cart data.
* **View:** The view displays the user interface, including the shopping cart.

**ADMIN MANAGEMENT (Including Order, Customer, Product and Categories)**

**A diagram of a software system

Description automatically generated**

*Figure 5-9 Admin features Sequence Diagram*

**Description:**The sequence diagram depicts the general flow of interactions within the Model-View-Controller (MVC) architecture for various admin actions. Initiated by the admin through the user interface, the View communicates user actions to the Controller, which, in turn, interacts with the Model for data retrieval or updates. The Model, representing the data and business logic, performs necessary operations, such as database interactions or business rule executions. Subsequently, the Model notifies the Controller of the results, and the Controller updates the View accordingly. This sequence ensures a coordinated and updated representation of data for actions like managing orders, customers, products, and categories, demonstrating the cohesive collaboration between the MVC components in handling diverse administrative functionalities.

## Key Decision and Justification

**MVC Architecture:**

**Justification:** The adoption of the Model-View-Controller (MVC) architecture is a fundamental design decision. This separation of concerns enhances maintainability by organizing code into distinct modules: Model for data and logic, View for user interface, and Controller for managing interactions. This design promotes scalability and ease of future updates or expansions.

**Responsive Web Design:**

**Justification:** Opting for a responsive web design ensures that the store website is accessible and functional across various devices and screen sizes. With the increasing use of mobile devices, this decision improves the overall user experience, broadening the reach of the store to a diverse audience.

**Database Management System (DBMS) Selection:**

**Justification:** Choosing an appropriate Database Management System - MongoDB, is crucial for efficient data storage and retrieval. The selection is based on factors like data structure, scalability requirements, and the need for relational or NoSQL capabilities, ensuring optimal performance and data integrity.

**Scalability Through Microservices:**

**Justification:** Considering the potential growth of the store and admin functionalities, adopting a microservices architecture allows for scalability. Breaking down the system into smaller, independent services facilitates easier deployment, maintenance, and updates. It also enables the use of different technologies for specific services based on their requirements.

**Security Measures:**

**Justification:** Implementing robust security measures, including HTTPS, encryption, and secure user authentication, is paramount. This ensures the protection of user data, financial transactions, and sensitive information. Regular security audits and updates will be implemented to address emerging threats and vulnerabilities.

**User-Friendly Admin Panel:**

**Justification:** The design of the admin panel will prioritize user-friendliness and intuitiveness. Store owners should be able to manage products, track sales, and update inventory effortlessly. A clean and well-organized interface, coupled with informative dashboards, ensures efficient website management.

**Regular Backups and Disaster Recovery Plan:**

**Justification:** Implementing a robust backup strategy and disaster recovery plan is essential to safeguard against data loss or system failures. Regular backups of the database and application code, coupled with a well-defined recovery plan, mitigate the impact of unforeseen events, and ensure business continuity.

By making these key design decisions, our project aims to deliver a robust, scalable, and secure solution that meets the needs of both end-users visiting the store website and store owners managing their online business through the admin panel. Each decision is carefully justified to align with the project's goals and long-term sustainability.

# Development

## Description of the Development Process

Throughout the development phase, several significant milestones were achieved, marking crucial progress in the project:

The development process for our project followed an iterative and collaborative approach, emphasizing the principles of Agile development. Here is a breakdown of the key stages:

1. **Requirements Analysis:** A thorough analysis of stakeholder requirements was conducted, ensuring a clear understanding of the project scope and objectives.
   * Gathered detailed requirements from stakeholders, including features for the store website and admin panel.
   * Created user stories and prioritized features based on business goals and user needs.
2. **System Design:** The design and development phase were initiated promptly, focusing on creating intuitive user interfaces and robust backend functionalities.

* ***Architectural Blueprints:***
  + Architectural blueprints were created, mapping out the system components and their interactions. The Model-View-Controller (MVC) architecture was chosen for its modularity and scalability benefits.
* ***Wireframes and Mockups:***
  + Detailed wireframes and mockups were designed to visualize the user interfaces for both the store website and the admin panel. This helped in aligning design concepts with stakeholder expectations.

1. **Database Design:** The database management system was implemented using MongoDB, enabling seamless data storage, retrieval, and management.

* ***Database Schema Definition:***
  + The database schema was meticulously defined, outlining the structure and relationships between data entities. This step ensured the efficient storage and retrieval of data in accordance with the needs of the Model component.
  + Established relationships between entities to ensure data integrity.

1. **Development Sprints:** The development sprints were executed in iterative cycles, facilitating continuous integration, testing, and feedback.

* ***Agile Methodology:***
  + Agile principles guided the development process, with regular sprints enabling the implementation of features in small, manageable increments.
* ***Incremental Feature Development:***
  + Features were developed incrementally, allowing for continuous testing and validation. This iterative approach ensured that each sprint delivered tangible progress.
* ***Collaborative Development:***
  + Cross-functional teams collaborated closely, fostering communication between developers, designers, and stakeholders. This collaborative environment supported the timely resolution of challenges and effective knowledge sharing.

1. **Testing:** The testing phase played a crucial role in ensuring the reliability and functionality of the developed system.

* ***Comprehensive Testing Suite:***
  + Unit testing, integration testing, and user acceptance testing were conducted comprehensively. This multifaceted approach aimed to identify and address issues at various levels of the application.
* ***Bug Tracking and Resolution:***
  + Bugs and issues discovered during testing were logged and addressed promptly. This proactive approach ensured that the final product met the specified requirements and quality standards.

1. **Deployment:** The deployment phase marked the transition from development to a live, accessible application.

* ***Hosting Platform Integration:***
  + The web application was deployed on a chosen hosting platform, ensuring accessibility to users. Compatibility checks were performed to guarantee optimal performance across different browsers and devices.
* ***Continuous Monitoring:***
  + Monitoring tools were implemented to track the application's performance post-deployment. This allowed for real-time identification of potential issues and swift resolution.

1. **User Feedback and Iteration:** The incorporation of user feedback and iterative improvements was a dynamic and ongoing process.

* ***Feedback Gathering:***
  + Feedback was actively sought from users and stakeholders through various channels, including surveys, usability testing, and direct communication.
* ***Iterative Refinement:***
  + Based on received feedback, iterative refinements were made to the user interfaces, functionalities, and overall user experience. This continuous improvement loop ensured that the system evolved in response to user needs and preferences.

## Programming languages, frameworks, and tools used.

**Development Environment:**

For the front end of our Online E-commerce Website project, we will harness the capabilities of Visual Studio Code (VSCode). VSCode is renowned for its versatility and robust feature set tailored for web development. It offers advanced code editing functionalities, comprehensive syntax highlighting, intelligent code completion, and powerful debugging tools. Moreover, its seamless integration with version control systems and extensive library of plugins enhance productivity and facilitate collaborative coding efforts.



*Figure 6-1 Visual Studio Code*

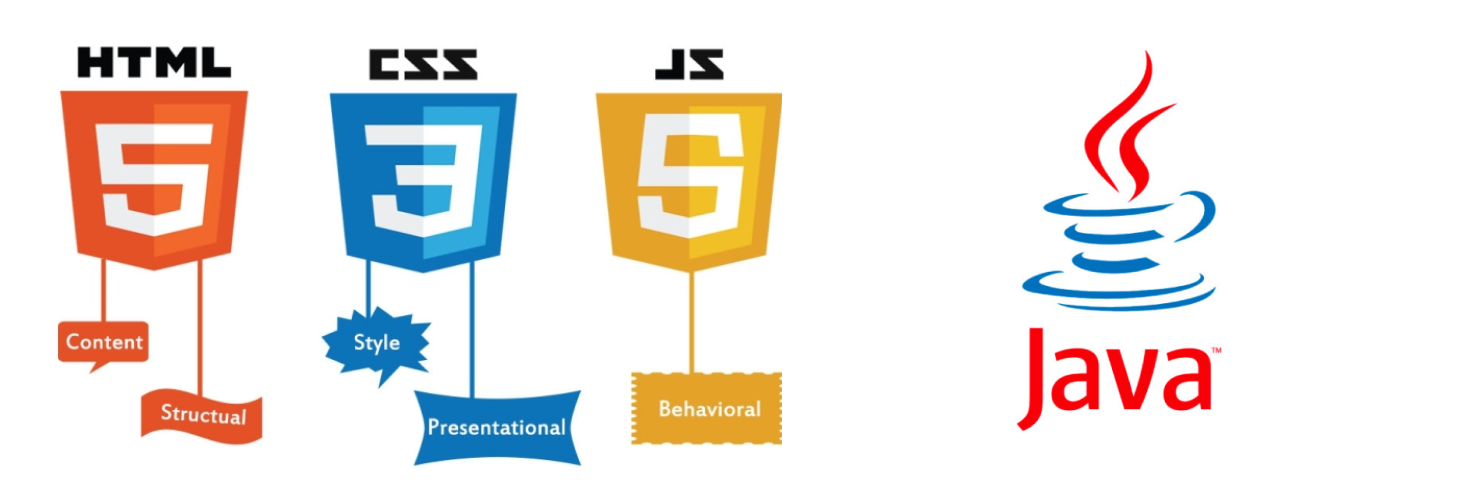
On the back end, we will utilize IntelliJ IDEA, a leading integrated development environment (IDE) specifically designed for Java development. IntelliJ IDEA provides an intuitive and efficient environment for writing, debugging, and deploying backend applications. It offers robust support for Java frameworks, intelligent code assistance, built-in version control, and seamless integration with various build tools. With its extensive plugin ecosystem and developer-friendly features, IntelliJ IDEA ensures a streamlined and productive development experience for backend programming tasks in our e-commerce project.



*Figure 6-2 Intellij IDEA*

**Programming Language**

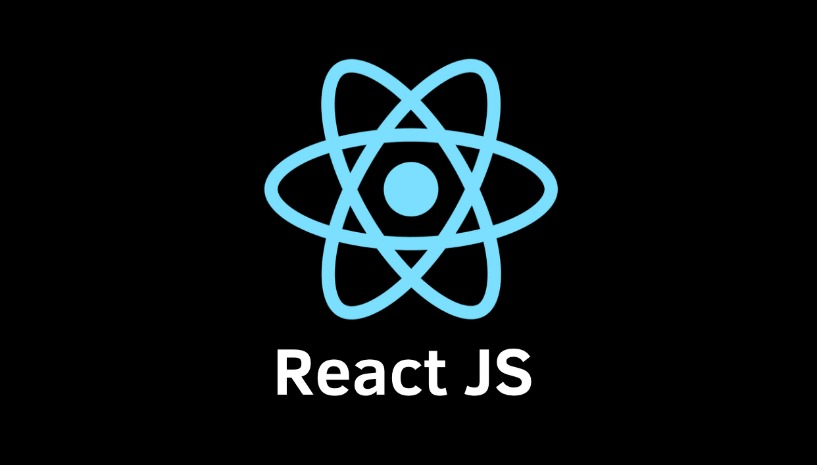
As we undertake our project, our technology stack revolves around four foundational programming languages: HTML (Hypertext Markup Language), CSS (Cascading Style Sheets), JavaScript, and Java. HTML lays the groundwork for structuring and organizing the content of our website, while CSS enhances its appearance and visual presentation. JavaScript assumes a pivotal role by introducing interactivity and dynamic functionalities to our web applications. Its versatility empowers us to implement diverse features crucial for our project, such as enabling search capabilities, managing shopping carts, and ensuring robust user authentication. Additionally, Java provides robust backend support, facilitating secure data handling and powering the server-side functionalities essential for our e-commerce's operations.



*Figure 6-3 HTML-CSS-JS-JAVA*

**Framework**

To enhance the user experience and ensure a dynamic frontend interface for our project, we will leverage React.js. React.js is a powerful JavaScript library for building user interfaces, known for its component-based architecture and efficient rendering capabilities. By adopting React.js, we aim to create responsive and interactive web pages that dynamically update as users interact with the application. React's virtual DOM and reusable components will streamline development, allowing us to maintain consistency across different parts of the application while ensuring optimal performance. Its robust ecosystem and extensive community support make React.js an ideal choice for delivering a modern and intuitive user interface.



*Figure 6-4 React JS*

For the backend infrastructure of our project, we will adopt Spring Boot, a robust Java-based framework renowned for its simplicity and productivity. Spring Boot simplifies the development of stand-alone, production-grade Spring-based applications, offering a comprehensive ecosystem that includes security, data access, and management features out of the box. With Spring Boot, we can rapidly develop RESTful APIs, manage database interactions through Spring Data, and ensure scalability and reliability. This framework's integration capabilities with various enterprise systems and cloud platforms will empower us to build a resilient backend architecture that meets the ecommerce's operational demands effectively.

****

*Figure 6-5 Spring boot*

**Additional Tool: Vite js**

To optimize the frontend development process further, we will integrate Vite.js into our technology stack. Vite.js is a fast and modern build tool specifically designed for frontend development with React.js and other JavaScript frameworks. It offers blazing-fast hot module replacement (HMR), optimized builds, and lightning-fast startup times, enhancing the developer experience and accelerating iteration cycles. By leveraging Vite.js, we aim to maximize productivity in frontend development and deliver a performant and responsive user interface for our project.



*Figure 6-6 Vite JS*

**Database Management System**

MySQL stands out as a highly reliable choice for website projects, renowned for its robust relational data model. Designed to efficiently handle structured data, MySQL is particularly suitable for e-commerce applications where data consistency and integrity are paramount. Its adherence to ACID (Atomicity, Consistency, Isolation, Durability) properties ensures transactions are processed reliably, making it ideal for managing financial and transactional data securely.

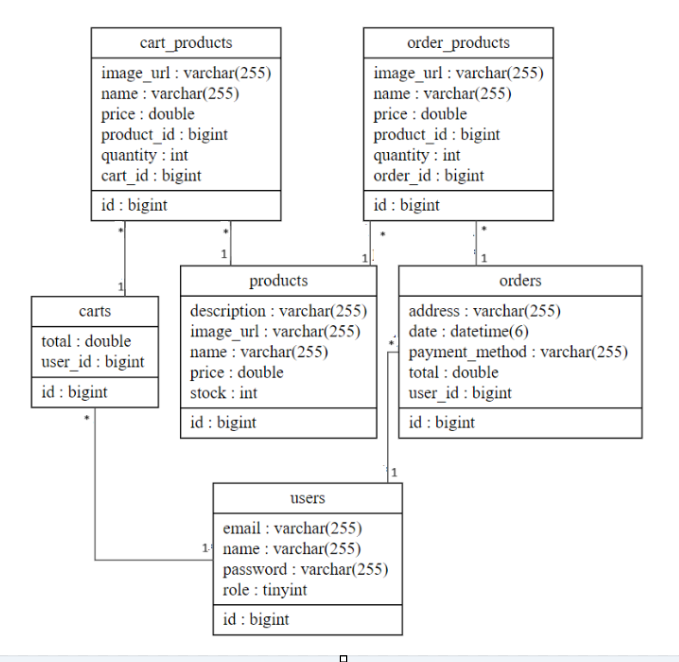
MySQL offers exceptional performance through its efficient indexing mechanisms, optimizing data retrieval and query execution. Its comprehensive support for SQL (Structured Query Language) enables complex querying and data manipulation operations, facilitating seamless integration with various programming languages and frameworks.

Scalability is another key advantage of MySQL, supporting horizontal scaling through replication and sharding strategies. This capability allows the database to grow alongside increasing data volumes and user demands, ensuring consistent performance under heavy loads.

Moreover, MySQL benefits from a mature ecosystem with extensive documentation, active community support, and a wide range of tools and utilities. This ecosystem enhances development efficiency and troubleshooting capabilities, making MySQL a dependable choice for building robust and scalable database solutions for our website project.



*Figure 6-7 MySQL*



*Figure 6-8 DB Schema*

**Relationships:**

cart\_products -- carts: Many to One

* Each cart\_product relates to one cart, but each cart is associated with many cart\_products.

cart\_products -- products: Many to One

* Each cart\_product corresponds to one product, but each product can be linked to many cart\_products.

carts -- users: Many to One

* Each cart belongs to one user, but each user can have multiple carts.

order\_products -- orders: Many to One

* Each order\_product is linked to one order, but each order can have multiple order\_products.

order\_products -- products: Many to One

* Each order\_product corresponds to one product, but each product can be associated with many order\_products.

orders -- users: Many to One

* Each order belongs to one user, but each user can have multiple orders

**Documentation and Presentation Tools**

Within our group, Google Docs and Canva play integral roles as indispensable tools for our documentation and presentation needs. Google Docs serves as a collaborative platform that allows us to generate, edit, and share documents in real-time. This fosters efficient collaboration among team members, enabling seamless version control and simultaneous work on project documentation. Furthermore, the inclusion of commenting and revision history features empowers us to provide feedback and track changes with ease. On the other hand, Canva empowers us to craft visually captivating presentations, infographics, and other graphical elements for our website project. Its user-friendly interface and vast array of templates, fonts, and graphics facilitate the creation of visually stunning visuals that effectively convey our ideas and concepts in a professional manner.



*Figure 6-9 Canva*

**Communication Tools**

Discord serves as our main communication tool, essential for fostering seamless collaboration and efficient team interactions. This platform allows team members to communicate through text chat, video conferencing, and voice calls, ensuring effective real-time communication no matter where we are located. Discord's integrated file-sharing features streamline the process of exchanging project documents, images, and other files, enhancing team coordination. With its robust set of communication and collaboration capabilities, Discord significantly boosts our group's productivity and ensures the smooth exchange of information



*Figure 6-10 Discord*

**Web Browsers**

To thoroughly test our website project, we rely on Google Chrome as our web browser of choice. Google Chrome provides an optimal testing environment with its robust developer tools and compatibility features. It allows us to inspect and debug code, analyze network performance, and assess the website's responsiveness across different devices and screen sizes. Google Chrome's compatibility with modern web standards ensures that our website functions correctly and offers an optimal user experience for a wide range of users. By utilizing Google Chrome for testing, we can identify and address any issues or inconsistencies promptly, ensuring that our website performs flawlessly across various browsers and platforms.



*Figure 6-11 Google Chrome*

## Challenges encountered during development.

**Ensuring Code Quality and Consistency**: Throughout the development cycle, maintaining uniform code quality across diverse team members posed a significant challenge. Variations in coding styles and practices could potentially lead to confusion and errors.

*Solution Implemented*: To mitigate these challenges, our team implemented robust measures. We instituted regular code reviews, leveraging version control systems to enforce consistency in coding standards. Additionally, we crafted comprehensive coding guidelines and best practices documents, fostering a unified approach among developers. Automated code analysis tools were integral in identifying and rectifying any deviations from established standards, thereby upholding a high level of code quality throughout the project's lifecycle.

**Navigating Third-Party Integrations**: Integrating disparate third-party services, such as payment gateways and analytics platforms, presented another hurdle. Differences in APIs, documentation formats, and data handling protocols often complicated seamless integration.

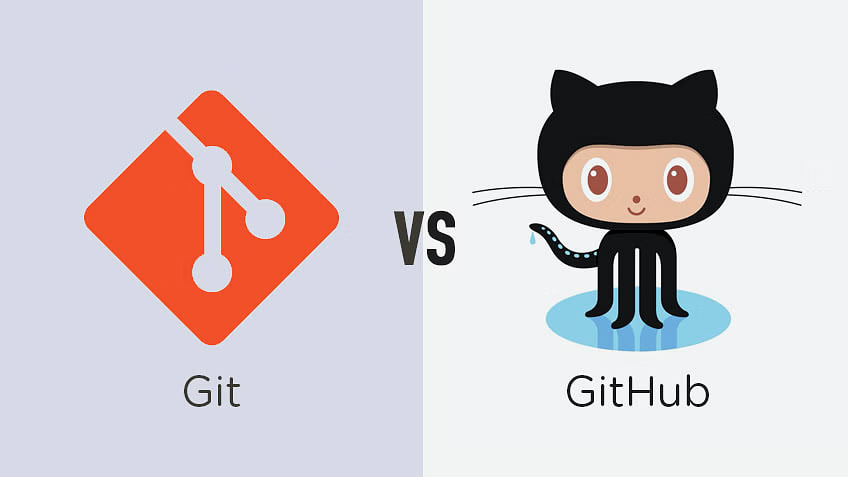
*Solution Implemented*: To address this, our strategy involved meticulous planning and collaboration. We allocated dedicated phases for API research and rigorous testing, ensuring compatibility and reliability before full-scale integration. Close partnerships with third-party providers facilitated swift resolution of integration challenges, supported by effective communication channels established for prompt issue resolution. Prior to deployment, thorough simulations and mock testing of API interactions preemptively identified potential communication gaps, enhancing the overall integration process's smoothness and reliability.

**Adapting to Evolving Requirements**: Responding agilely to evolving stakeholder demands proved essential but challenging, as changes could impact established timelines and development strategies.

*Solution Implemented*: Embracing agile methodologies proved pivotal in navigating shifting project requirements. Regular sprint planning sessions and retrospective meetings enabled continuous evaluation and adjustment of our development roadmap. Transparent communication channels with stakeholders fostered a deep understanding of evolving needs, empowering our team to implement necessary adaptations efficiently. By prioritizing flexibility and responsiveness, we effectively accommodated changes while minimizing disruption to project timelines and ensuring alignment with overarching project goals.

## Code Management and version control

Our team relies on Git and GitHub for version control and collaboration. Git, a distributed version control system, becomes useful for effectively tracking and managing code changes. It allows us to work on the same project at the same time, integrate changes seamlessly, and quickly rollback to prior versions as needed. On the other hand, GitHub provides a web-based hosting infrastructure for our Git repositories. It creates a collaborative environment that improves several areas of our productivity. We may monitor bugs, propose and approve modifications via pull requests, and promote effective coding cooperation among team members.



*Figure 6-12 Git and Github*

# Testing and Quality Assurance

*Table 7-1 Test Case 001*

| **Test Case #:1** | **Test Case Name: User register** |
| --- | --- |
| **System: E-commercial** |  |
| **Designed by: Group** | **Design Date: 08/6/2024** |
| **Executed by: Group** | **Execution Date: 13/6/2024** |
| **Short Description: Test user account creation** |  |



| **Step** | **Action** | **Expected System Response** | **Pass/Fail** | **Comments** |
| --- | --- | --- | --- | --- |
| 1 | Click the create an account button | The system navigate to Sign up page | pass |  |
| 2 | Submit register form with all field empty | The Welcome user should be displayed at the top-left corner and the system navigate to user profile page | pass |  |
| 3 | Submit form with invalid data | Proper validation messages should be displayed, and the form should not be submitted | pass |  |
| 4 | Access the User Profile page after registration | The user’s profile information should be correctly displayed | pass |  |



*Table 7-2 Test Case 002*

| **Test Case #:** 2 | **Test Case Name: Product browser** |
| --- | --- |
| **System: E- commercial** |  |
| **Designed by: Group** | **Design Date: 09/6/2024** |
| **Executed by: Group** | **Execution Date: 13/6/2024** |
| **Short Description: Test product link** | |



| **Step** | **Action** | **Expected System Response** | **Pass/Fail** | **Comments** |
| --- | --- | --- | --- | --- |
| 1 | Click All on the menu bar | The product listing page should be displayed with all product | Pass |  |
| 2 | Click on product | The product details page should be displayed with accurate information | Pass |  |
| 3 | Go back to the product listing page from product details | The user should be able to continue browsing products seamlessly | Pass |  |



*Table 7-3 Test Case 003*

| **Test Case #: 3** | **Test Case Name: Check out** |
| --- | --- |
| **System: E-comercial** |  |
| **Designed by: Group** | **Design Date: 1/5/2024** |
| **Executed by: Group** | **Execution Date: 13/5/2024** |
| **Short Description: Test check out system** | |

**Pre-conditions:**

- Ensure that the user is logged in and has added at least 1 product to the cart

- Currently in shopping cart page

| **Step** | **Action** | **Expected System Response** | **Pass/Fail** | **Comments** |
| --- | --- | --- | --- | --- |
| 1 | Click checkout button | The checkout page should be displayed with fields for payment information | Pass |  |
| 2 | Updating valid payment information | The system navigate to ordered page with all information about the items have completed purchase | Pass |  |
| 3 | Updating invalid payment information | The error message should be displayed, and the form should not be submitted | Pass |  |
| 4 | Review order summary | The order summary should match the purchased products | pass |  |
| 5 | Go back to the product listing page from order summary | The user should be able to continue browsing products seamlessly | Pass |  |



*Table 7-4 Test Case 004*

| **Test Case #: 4 Test Case Name: Website Navigation**  **System: E-commercial**  **Designed by: Group Design Date: 10/6/2024**  **Executed by: Group Executed Date: 14/6/2024**  **Short Description: Test All links are correct and navigate smoothly** |
| --- |

| **Pre-conditions:**  **The system displays main menu** |
| --- |

| **Step** | **Action** | **Expected System Response** | **Pass/Fail** | **Comments** |
| --- | --- | --- | --- | --- |
| 1 | Click on the button “Product” | The system shows all the products with the images and price/each with the responding time is around 1 second | Pass |  |
| 2 | Click on the button “Home” | The system shows navigate to home page | Pass |  |
| 3 | Click on the button “Account” | The system shows the products that are labeled football with the responding time is around 1 second | Pass |  |

*Table 7-5 Test Case 005*

| **Test Case #: 5 Test Case Name: User login**  **System: E-commercial**  **Designed by: Group Design Date: 11/6/2024**  **Executed by: Group Executed Date: 14/6/2024**  **Short Description: Test user login** |
| --- |

| **Pre-conditions:**  **The current email:** [**123@gmail.com**](mailto:123@gmail.com)  **The current password: 123**  **Post-conditions**  **The system display login page** |
| --- |

| **Step** | **Action** | **Expected System Response** | **Pass/Fail** | **Comments** |
| --- | --- | --- | --- | --- |
| 1 | Click on “Sign in” | An interface appears asking the user to enter email and password and a link leads to the registration interface | Pass |  |
| 2 | Input email 123 | System reminds “Please include the ‘@’ in email address | Pass |  |
| 3 | Input email 123@gmail.com | Nothing appears | Pass |  |
| 4 | Input password 1 | The system restarts the login page and warm user “Wrong password” | Pass |  |
| 5 | Input password 123 | The system move to home page and change the interface login into “Welcome ‘username’ “ | Pass |  |

*Table 7-6 Test Case 006*

| **Test Case #: 6 Test Case Name: Search**  **System: E-commercial**  **Designed by: Group Design Date: 11/6/2024**  **Executed by: Group Executed Date: 14/6/2024**  **Short Description: Test product search system works well** |
| --- |

| **Step** | **Action** | **Expected System Response** | **Pass/Fail** | **Comments** |
| --- | --- | --- | --- | --- |
| 1 | Click to search box and type “asdjfijhaojw” | how all products if they have the keyword “asdjfijhaojw” in the name. |  |  |
| 2 | Type invalid characters in the name like !@# | System will show white page |  |  |

*Table 7-7 Test Case 007*

| **Test Case #: 7 Test Case Name: Shopping cart**  **System: E-commercial**  **Designed by: Group Design Date: 07/6/2024**  **Executed by: Group Executed Date: 13/6/2024**  **Short Description: Test adding/removing product to shopping cart** |
| --- |

| **Pre-conditions:**  **Already have 1 runner comfort with price is $21 in cart.**  **The number at shopping icon is 1**  **The final total price is $21**  **Post-conditions:**  **The system display shopping cart page with the plus and minus button in the right of product’s image, then quantity, total, subtotal and “removel” icon** |
| --- |

| **Step** | **Action** | **Expected System Response** | **Pass/Fail** | **Comments** |
| --- | --- | --- | --- | --- |
| 1 | Click plus button | The system shows 2 in quantity, $21 in total and $42 in subtotal display. At the same time, the shopping icon right where the username is also displayed 2, and the total at the end of page is $42 |  |  |
| 2 | Click minus button | The system shows 1 in quantity, $21 in total and $21 in subtotal display. At the same time, the shopping icon right where the username is also displayed 1, and total at the end of page is $21 |  |  |
| 3 | Click remove icon | The system shows the notification “No item in the cart” |  |  |
| 4 | Click to the image of product at menu | The system shows the information of product |  |  |
| 5 | Click “Add to Cart” | The shopping icon display 1 and shopping cart page display quantity: 1, total: $21, subtotal: $21 and final total: $21 before checkout |  |  |

*Table 7-8 Test Case 008*

| **Test Case #: 8 Test Case Name: Order tracking**  **System: E-commercial**  **Designed by: Group Design Date: 04/6/2024**  **Executed by: Group Executed Date: 13/6/2024**  **Short Description: Test tracking payment status and**  **Information of user** |
| --- |

| **Pre-conditions:**  **Admin email:** [**admin@gmail.com**](mailto:admin@gmail.com)  **Admin password: admin**  **Post-conditions:**  **The system display the Admin page** |
| --- |

| Step | Action | Expected System Response | Pass/Fail | Comments |
| --- | --- | --- | --- | --- |
| 1 | Click on the drop down arrow next to the account button, it will display a list of options click on My orders | The system shows the list order of users |  |  |
| 2 | Click on any order | The system shows all information of user, production information, total quantity, total cost, payment method and shipping address |  |  |

*Table 7-9 Test Case 009*

| **Test Case 09: Manage Product - Add Product**  **System: E-commercial**  **Designed by: Group**  **Executed by: Group**  **Short Description: Verify that a product can be successfully added.** | **Test Case Name: Add Product**  **Design Date: 07/6/2024**  **Execution Date: 12/6/2024** |
| --- | --- |

| **Pre-conditions:**   * **The admin is logged in and on the "Home page" page.** |
| --- |

| **Step** | **Action** | **Expected System Response** | **Pass/Fail** | **Comments** |
| --- | --- | --- | --- | --- |
| 1 | Click on "Add Product" button | New product form is displayed | Pass |  |
| 2 | Fill in product details | Data is entered without errors | Pass | Ensure all required fields are filled |
| 3 | Click on "Add Product" button | Product is added successfully | Pass |  |
| 4 | Verify product presence in the system | Product is displayed in the list | Pass | Check the product in the product listing |

| **Post-Conditions:**   * **The new product is successfully added to the system.** |
| --- |

*Table 7-10 Test Case 010*

| **Test Case 10: Manage Product - Remove Product**  **System: E-commercial**  **Designed by: Group**  **Executed by: Group**  **Short Description: Verify that a product can be successfully removed.** | **Test Case Name: Remove Product**  **Design Date: 07/6/2024**  **Execution Date: 12/6/2024** |
| --- | --- |

| **Pre-conditions:**   * **The admin is logged in and on the “Home" page.** |
| --- |

| **Step** | **Action** | **Expected System Response** | **Pass/Fail** | **Comments** |
| --- | --- | --- | --- | --- |
| 1 | Click “Edit” button in product | Product details are displayed | Pass | Ensure a product is selected |
| 2 | Click on "DELETE Product" button | Confirmation prompt appears | Pass | Confirm the intention to remove |
| 3 | Verify product absence in the system | Product is not listed anymore | Pass | Ensure the product is removed |

| **Post-Conditions:**   * **The selected product is successfully removed from the system.** |
| --- |



*Table 7-11 Test Case 011*

| **Test Case 11: Manage Categories - Update Product**  **System: E-commercial**  **Designed by: Group**  **Executed by: Group**  **Short Description: Verify that a category can be successfully updated.** | **Test Case Name: Update Product**  **Design Date: 07/6/2024**  **Execution Date: 12/6/2024** |
| --- | --- |

| **Pre-conditions:**   * **The admin is logged in and on the "Home" page.** |
| --- |

| **Step** | **Action** | **Expected System Response** | **Pass/Fail** | **Comments** |
| --- | --- | --- | --- | --- |
| 1 | Click “Edit” button in product | Product details are displayed | Pass | Ensure a product is selected |
| 2 | Fill in the information that needs to be changed in the fields in the form | Data is entered without errors | Pass | Ensure all required fields are filled |
| 3 | Click on "Save Changes" button | Confirmation prompt appears | Pass | Confirm the intention to update |
| 4 | Verify product changes in the system | Product is displayed changes in the list | Pass | Ensure the product is updated |

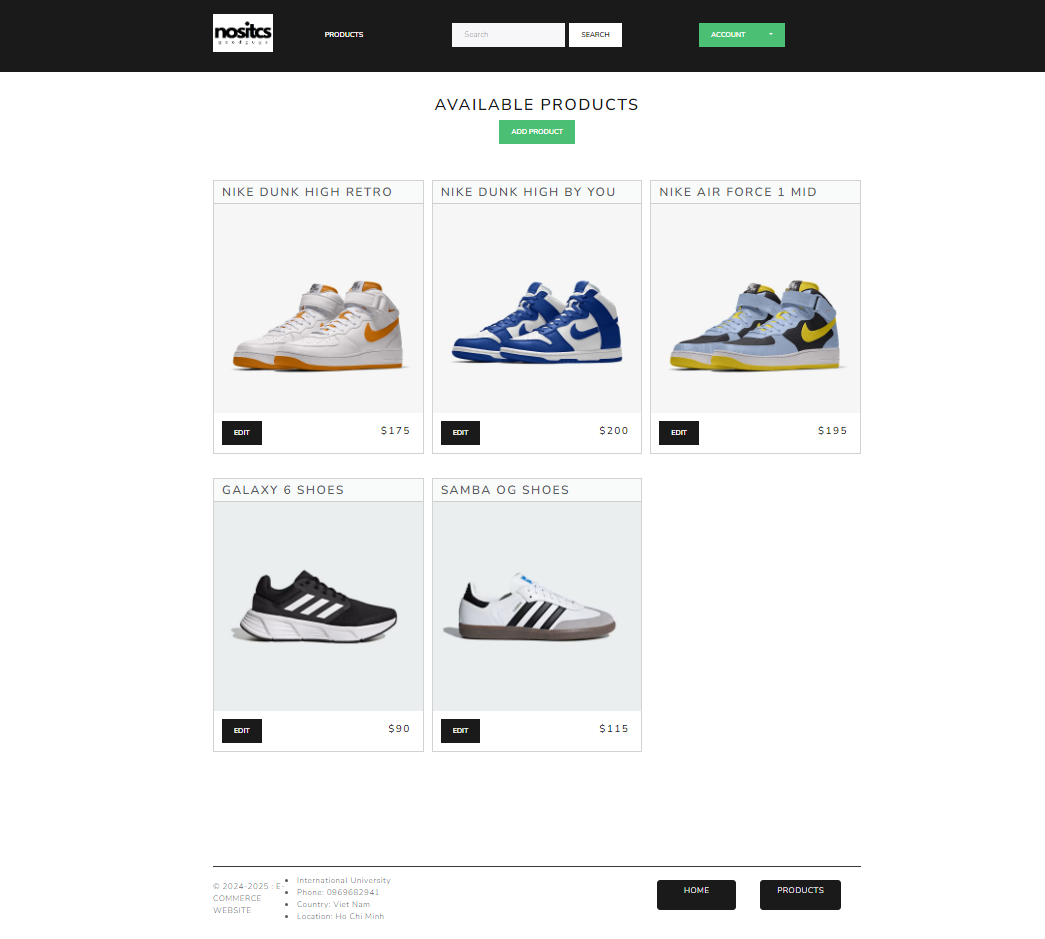
| **Post-Conditions:**   * **The selected product is successfully updated from the system.** |
| --- |

# User Documentation

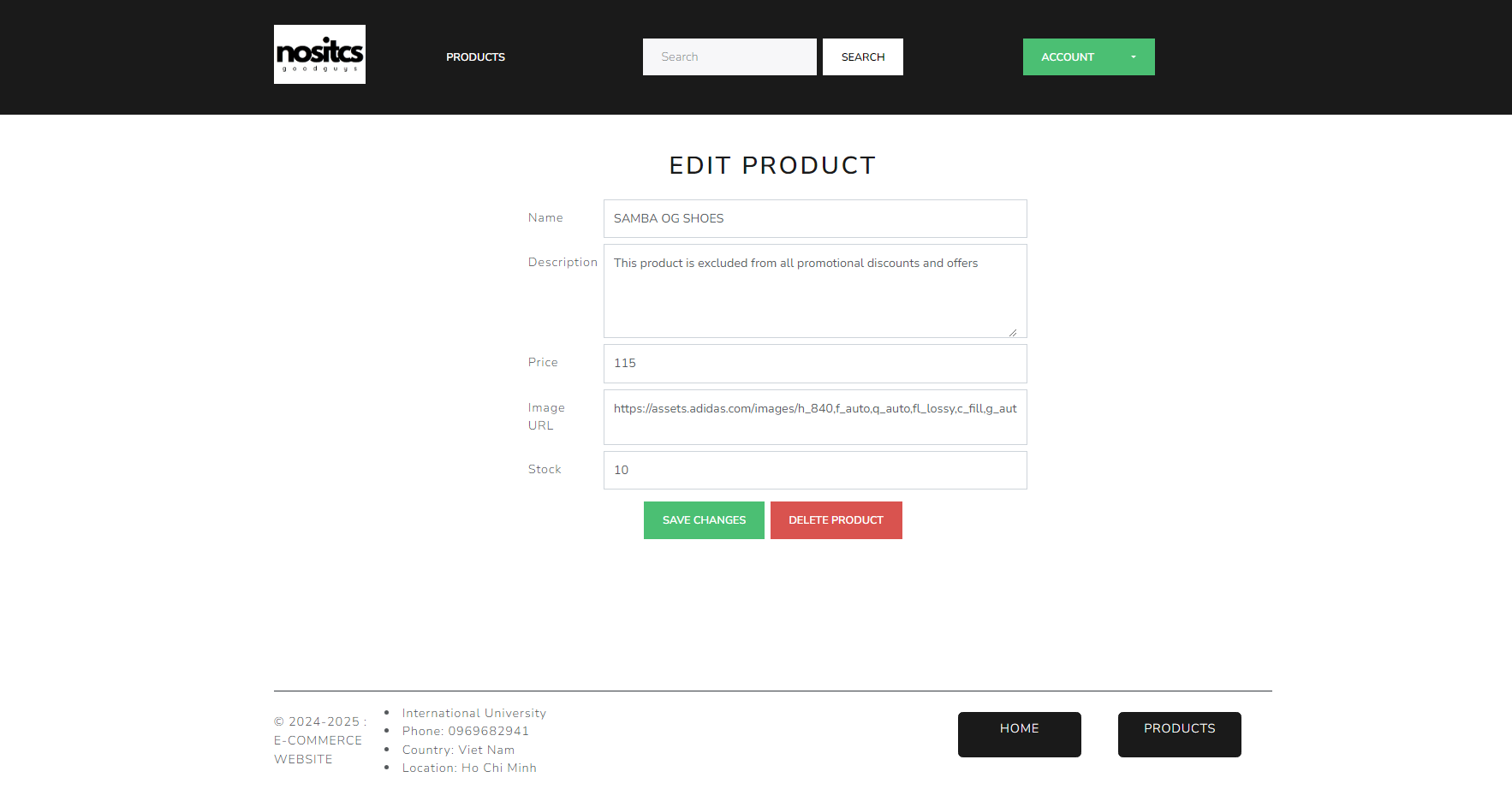
1. **Introduction:**

* The [E-commerce Website] (Shoes) is a web-based software designed to efficiently manage operations related to the sale of footwear. It encompasses features for configuring the software to meet the specific needs of the footwear retail business, overseeing policy administration, managing customers and orders, and facilitating order processing. This guide serves as a manual for understanding and utilizing the functionalities of the software, focusing on user-friendly instructions rather than intricate technical details. The Contents section acts as a roadmap, offering direct links to each major chapter and its corresponding sub-chapters. By clicking on the content title (in the online version), users are seamlessly directed to the relevant section of interest.

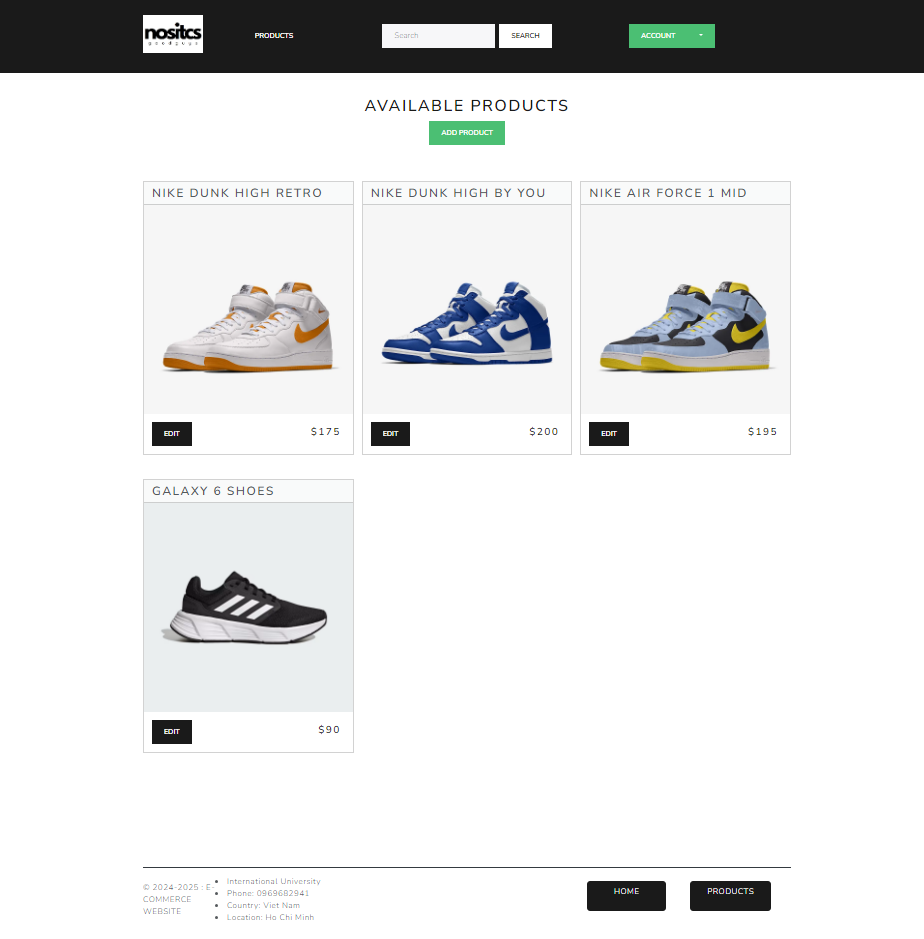
1. **‘Admin’ View Home Page**

****

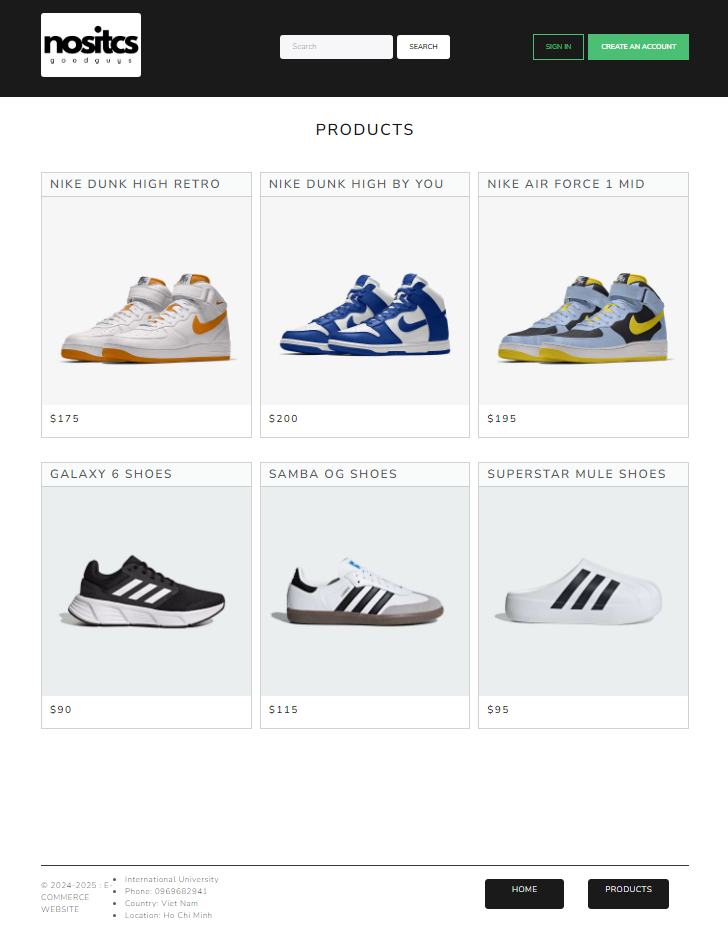
1. **‘Admin’ Edit**

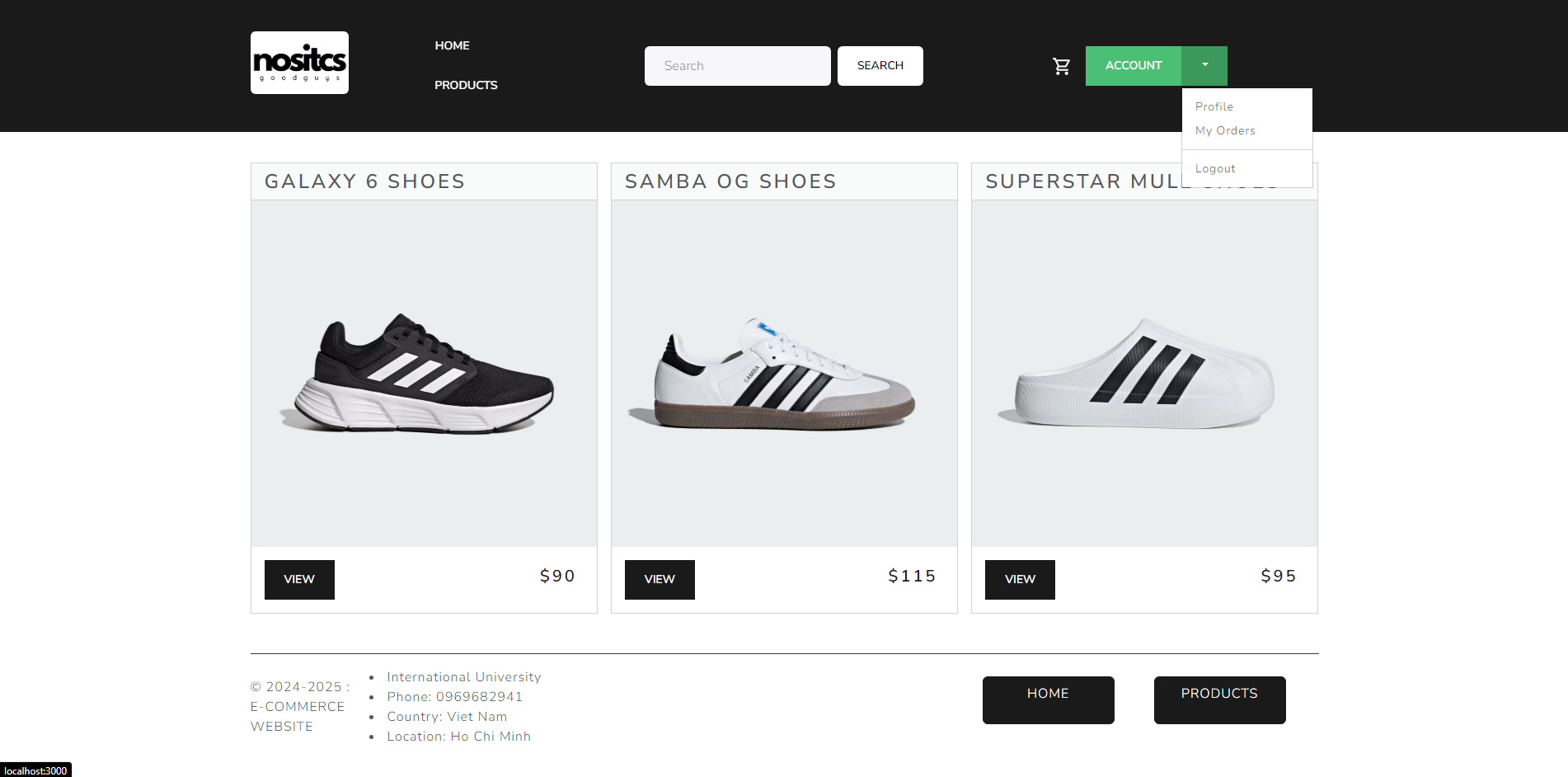
****

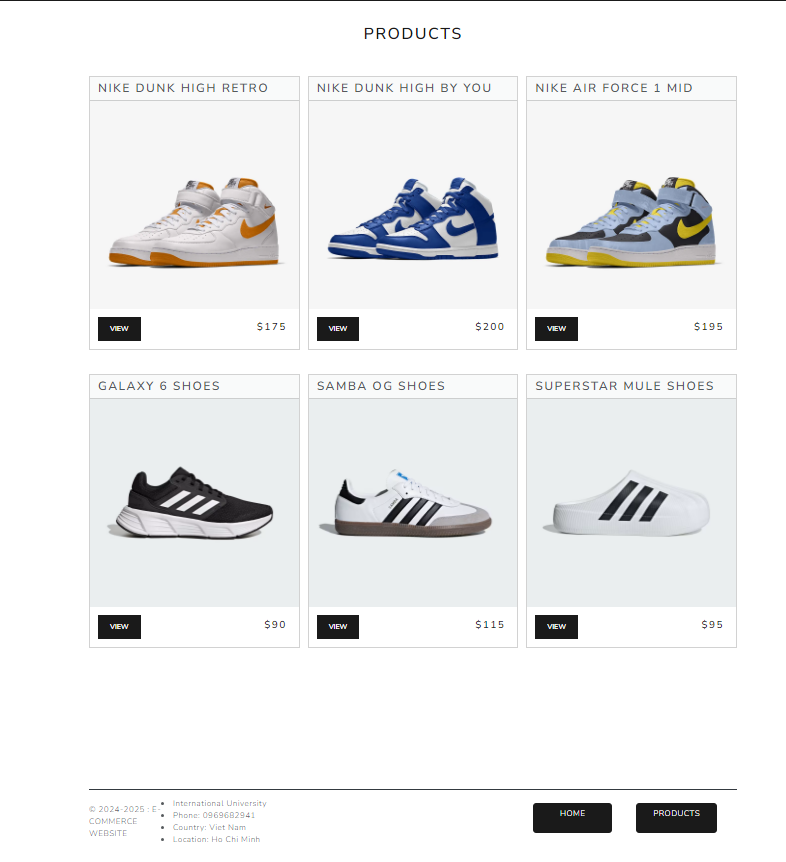
**Delete ones**

****

1. **View Home Page**

****





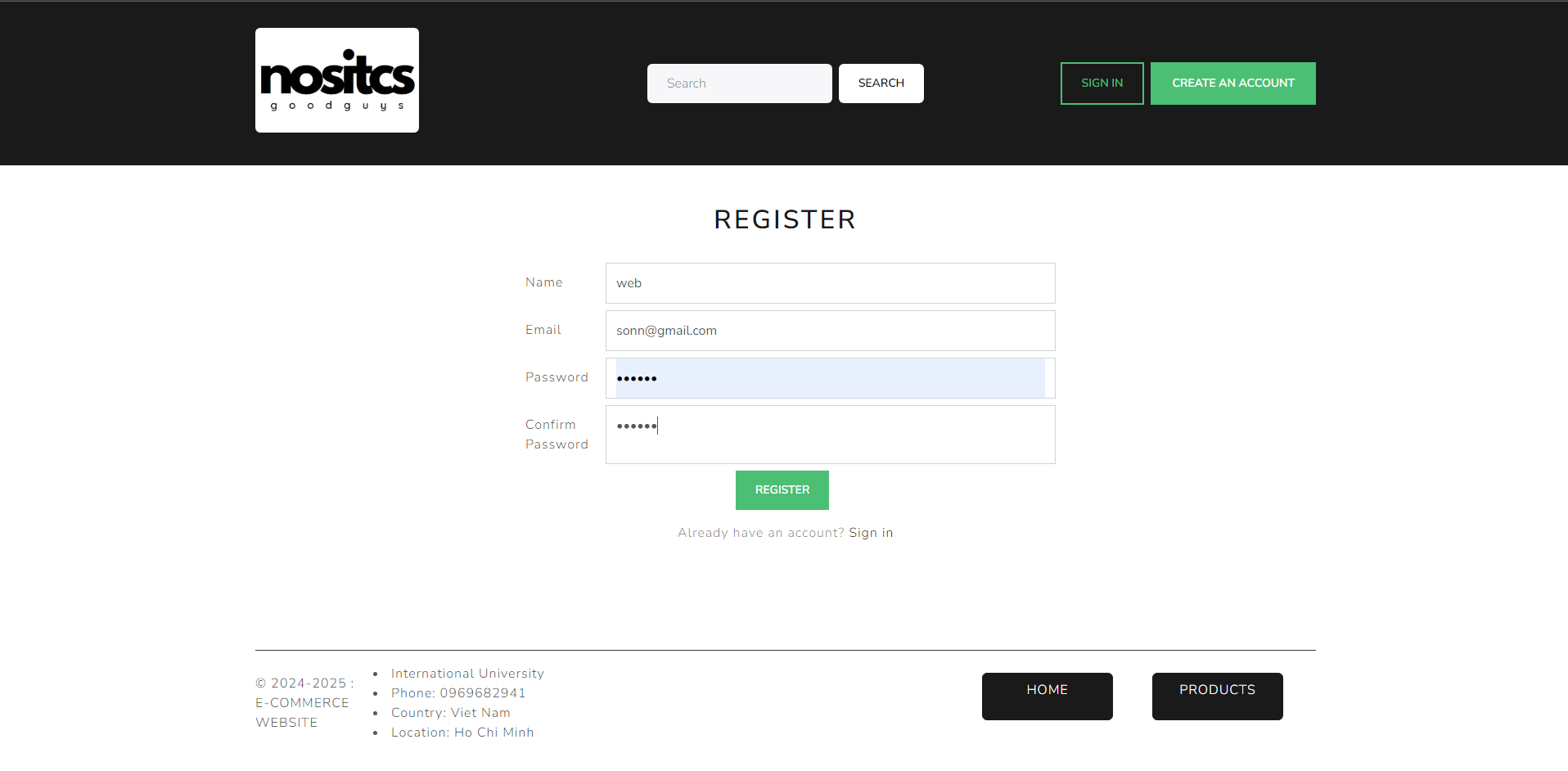
*Figure 8-1 Homepage Interface*

On the homepage interface, users can interact with the website through some basic features such as logging in or registering an account. Additionally, users can explore specific product categories that interest them, such as basketball shoes. If they have a specific type of shoe in mind, they can use the search bar to search more specifically, allowing them to easily find the type of shoe they want. Moreover, the website also recommends you the latest products so you can easily choose the appropriate product.

1. **Sign Up / Sign In**

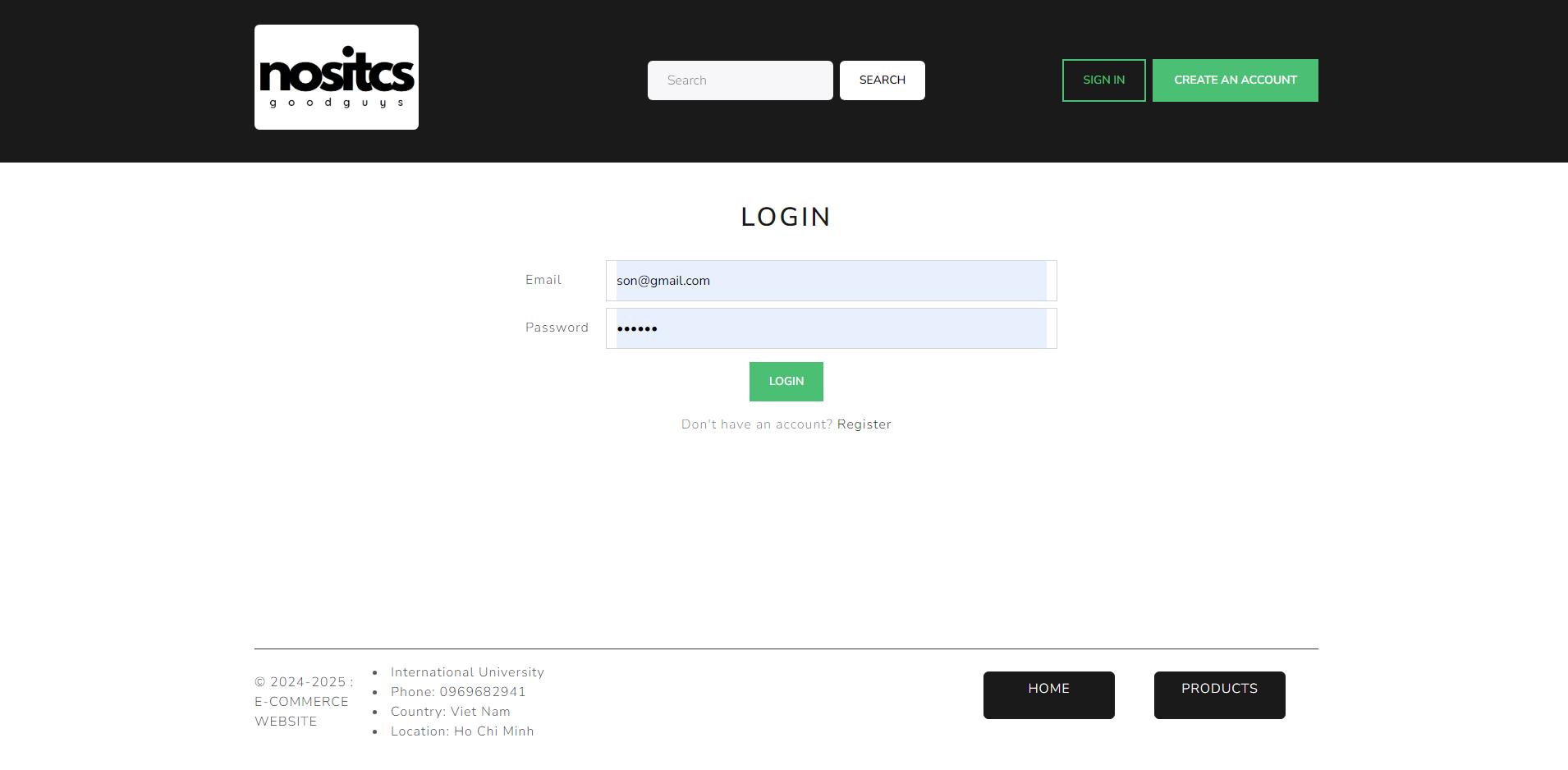
* **Sign Up**

**Step 1:** Click on the 'Create an account' function on the right-side bar to register an account.



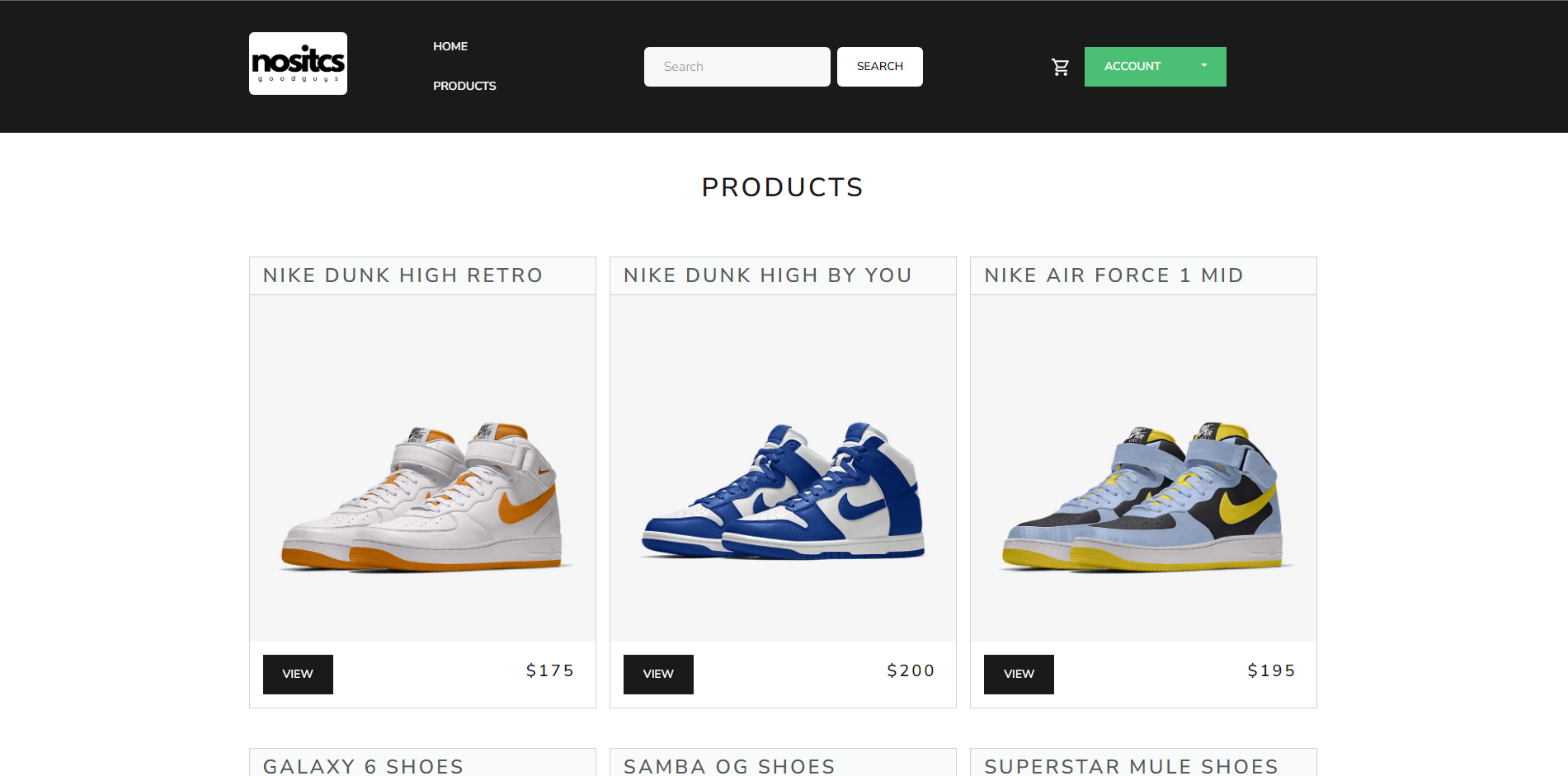
*Figure 8-2 Signup 1*

The Sign Up will display after you perform this action like below.



*Figure 8-3 Signup 2*

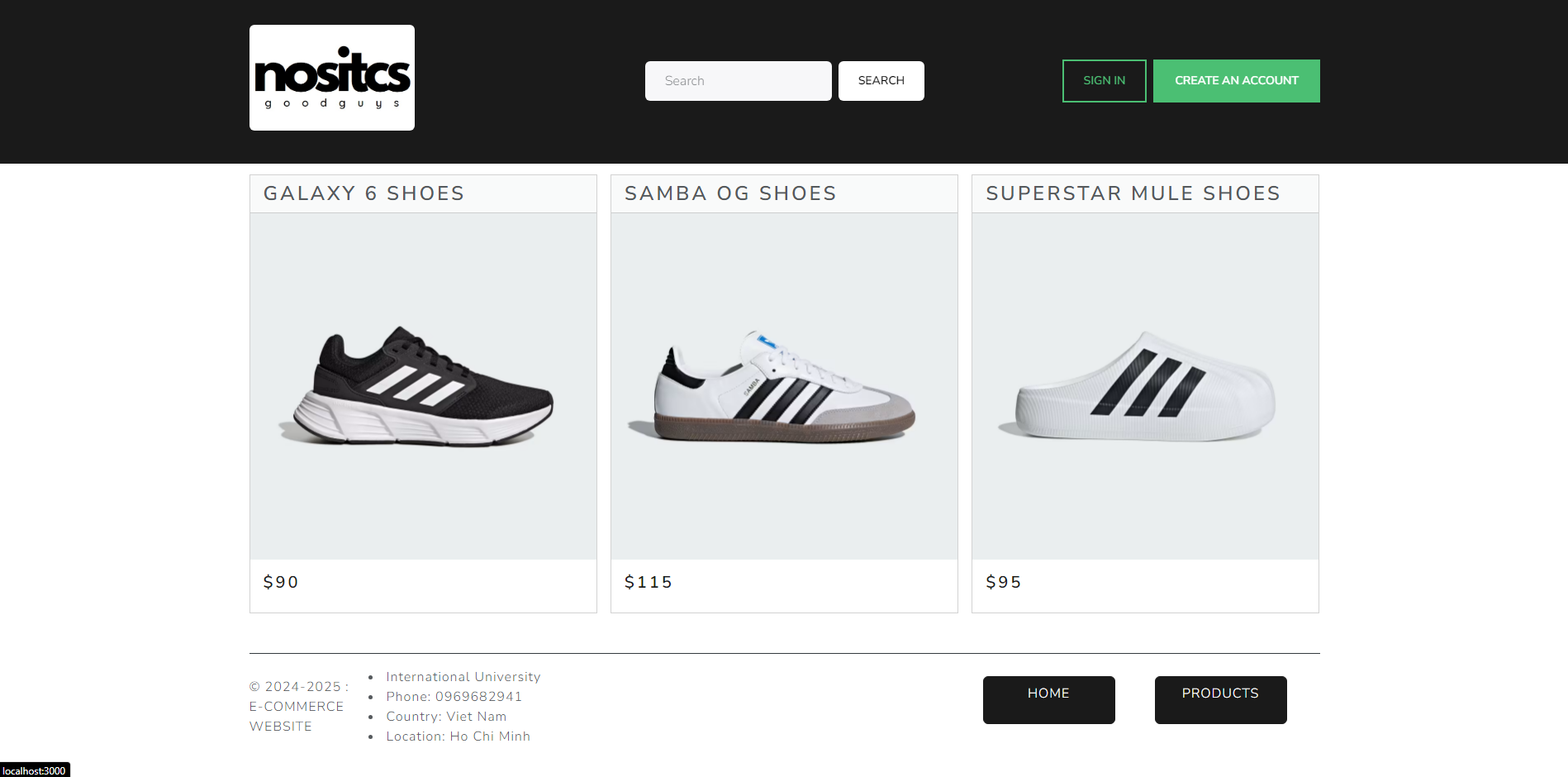
**Step 2:** User logged in



*Figure 8-4 Signup 3*

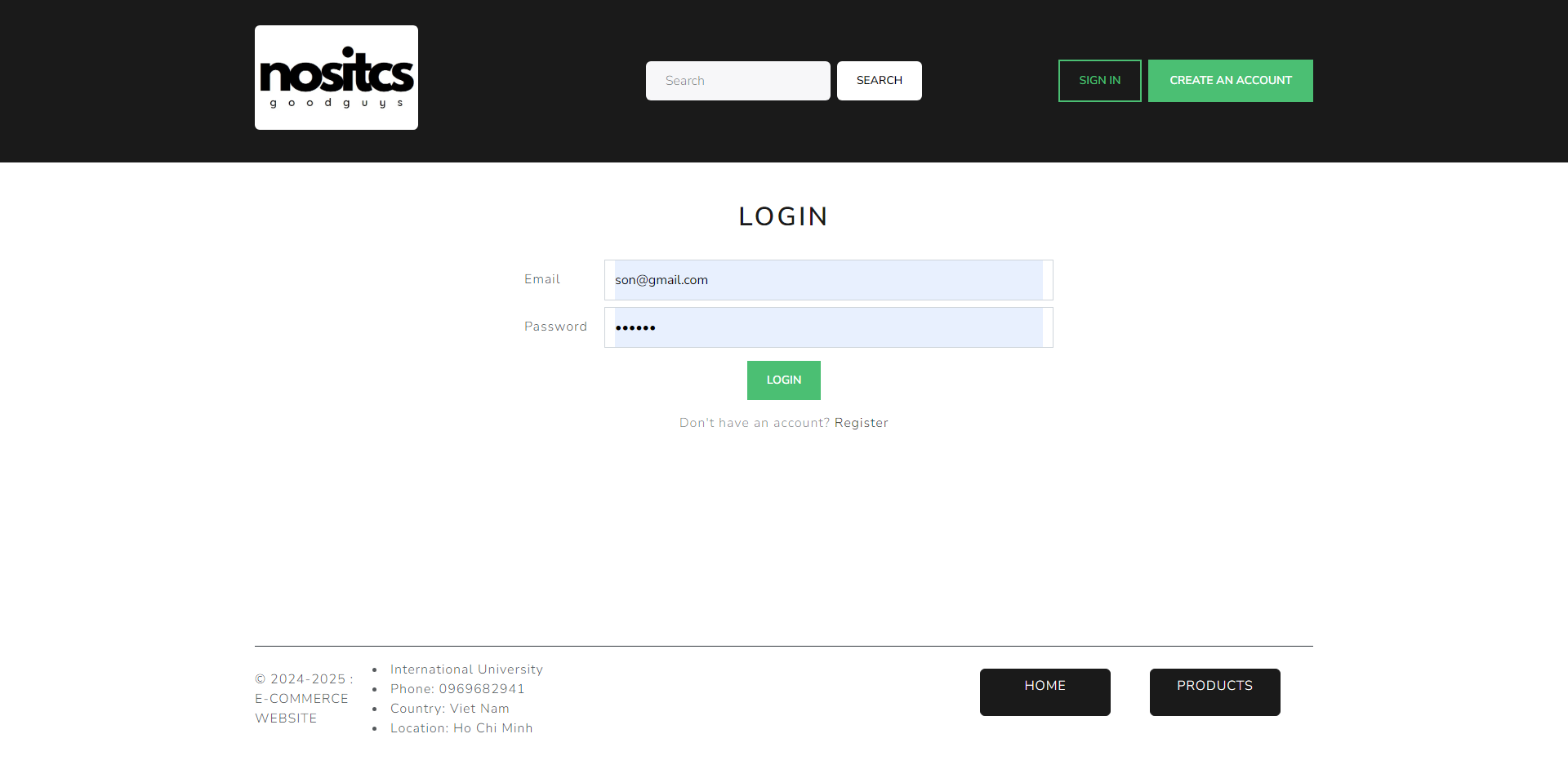
* **Sign In**

**Step 1:** Click on the 'Sign In' function on the right-side bar to register an account.

****

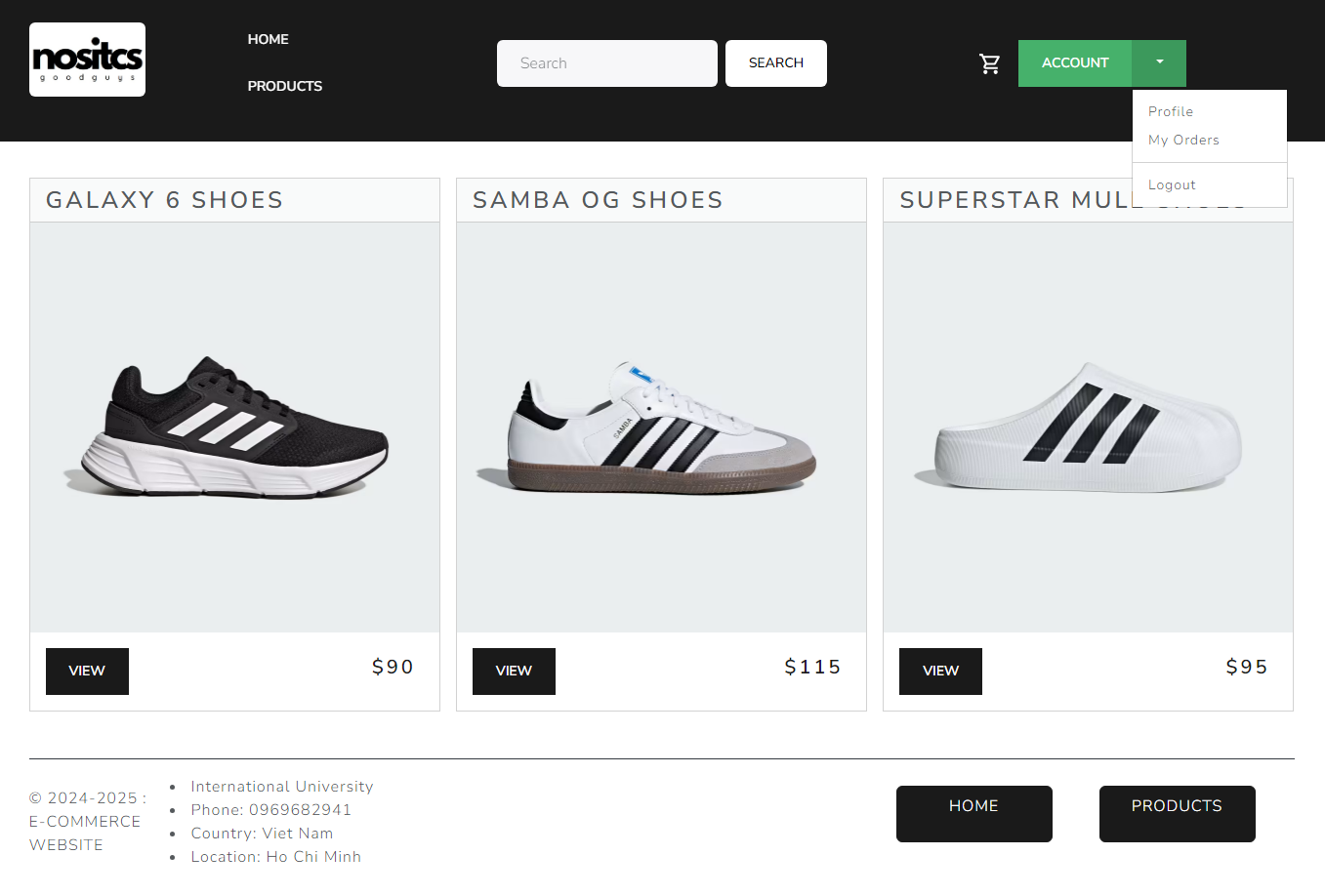
*Figure 8-5 Sign in 1*

The Sign In will display after you perform this action like below.

****

*Figure 8-6 Sign in 2*

**Step 2:** We have just registered to log in to the website system and clicked the "Sign In" button. Additionally, if we don't have an account, we can still access the Sign-Up page on the Login page. And just like on the Sign-Up page, the system will take you to the home page.

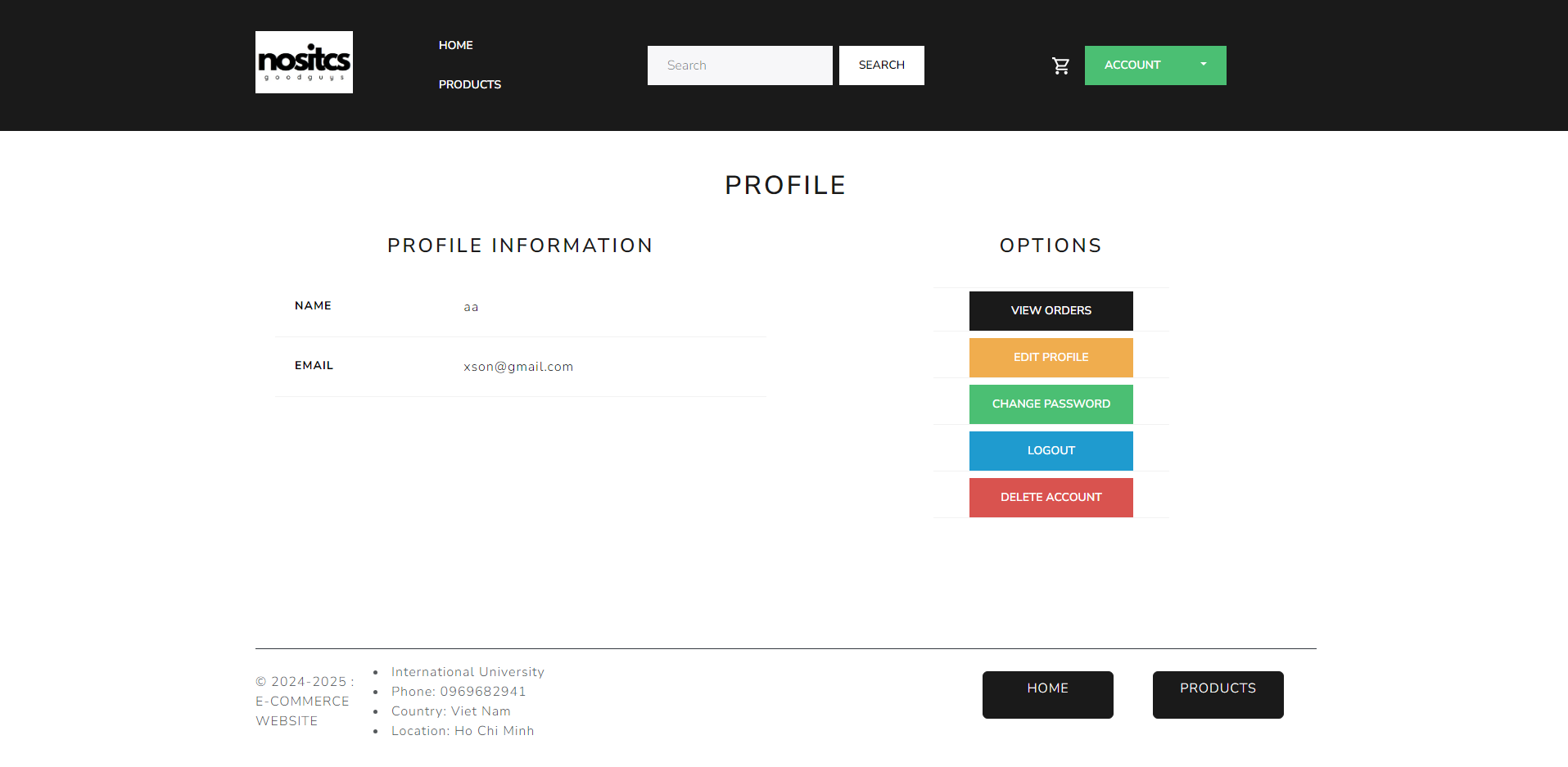


*Figure 8-7 Sign in 3*

1. **User detail**

This is the User detail page: There are Profile information. Some option like:

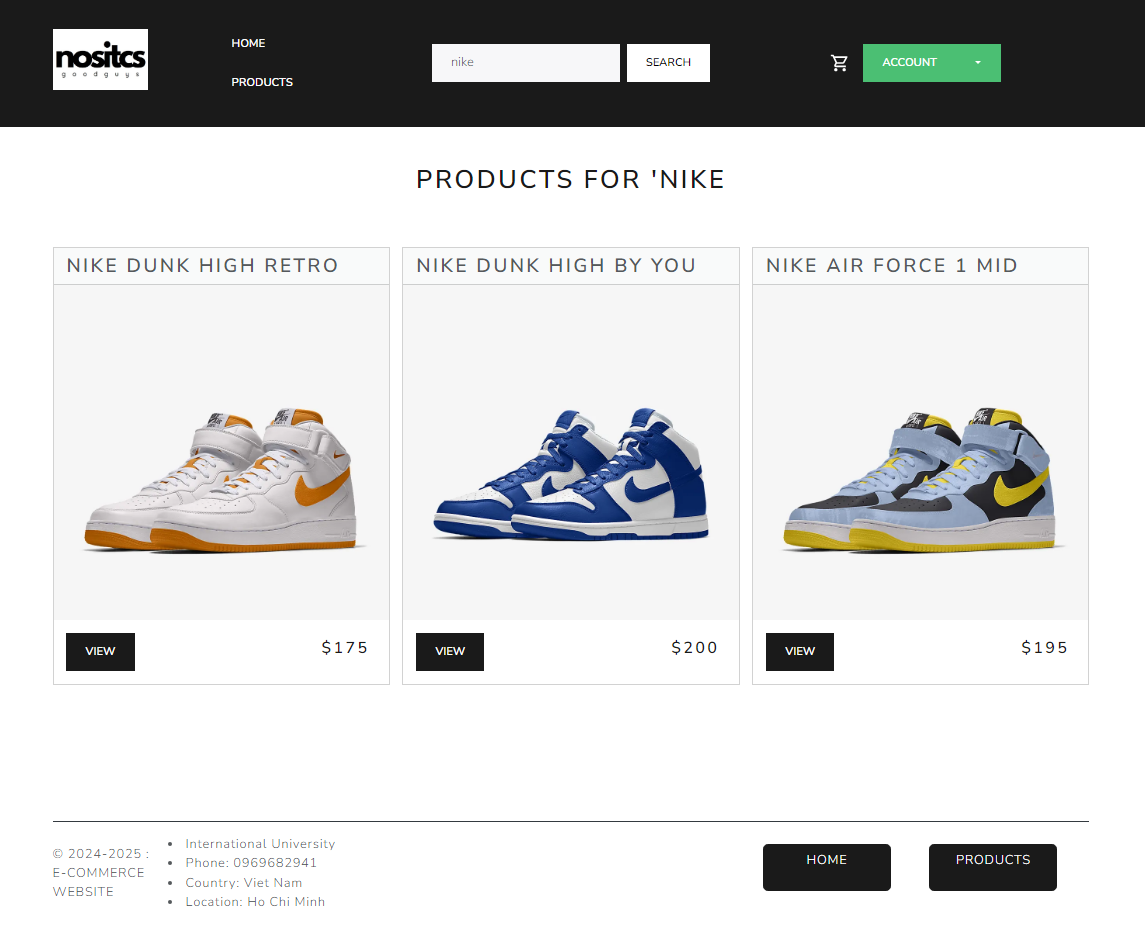
View, Edit, Change, Logout, Delete.

****

*Figure 8-8 User Details*

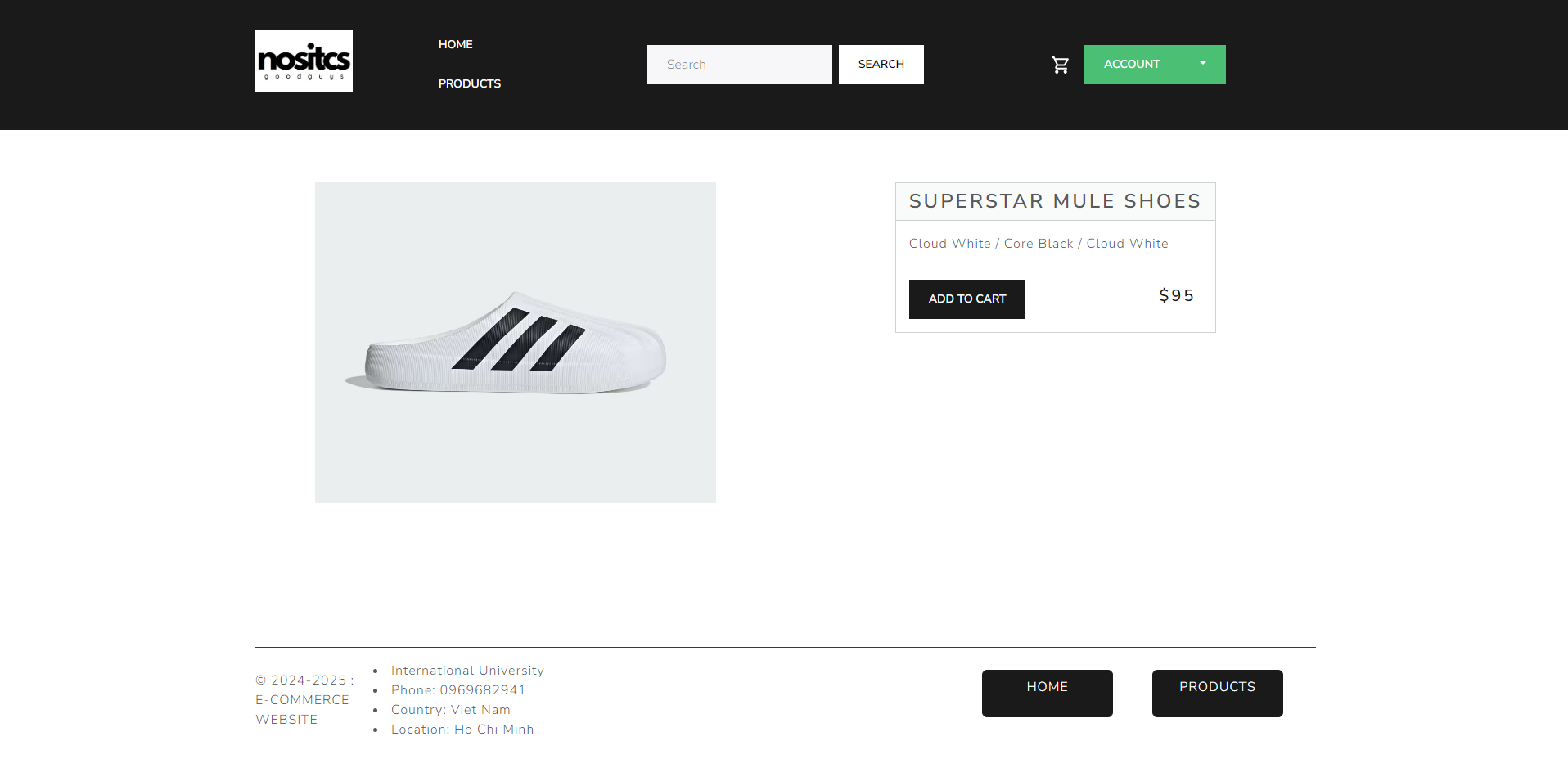
1. **Search**

Click keyword on sidebar in head of website. Example” nike “. The products relate with keyword “nike ” will be appearance.

*Figure 8-9 Search*

1. **Product Details**

**Step 1:** In this section, users need to clearly identify the product they want to buy. Then we double click on that product. The page product details screen will look like this. Here we will give an example of the product: "superstar mule shoes".

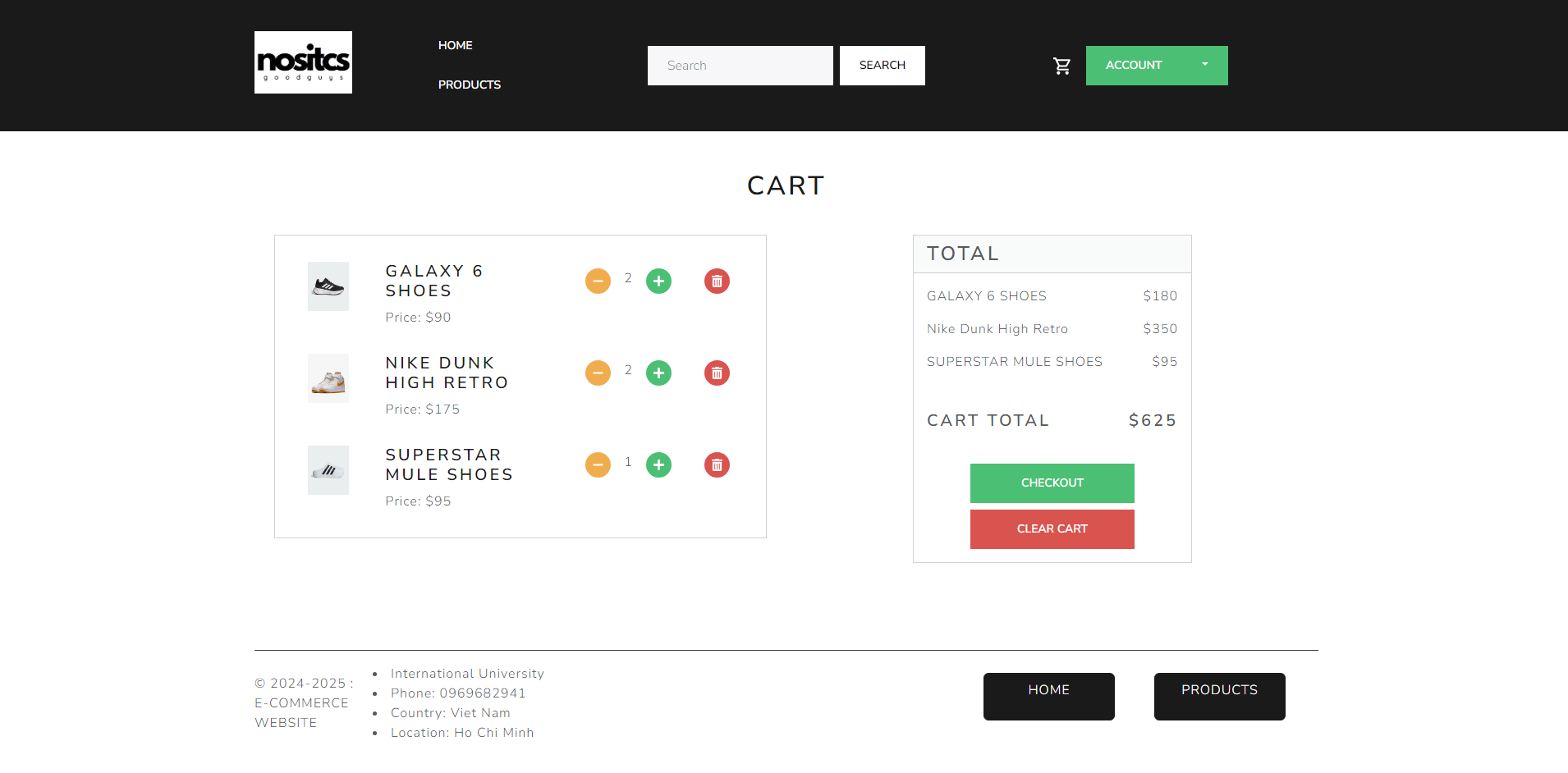


*Figure 8-10 Product Details*

**Step 2:** On the product detail page, we will see product details including Product Image, Price of product, Description, Category, Brand, Product-code, Date added to the store, stock notification and finally the "Add to cart" button.

1. **Marking Purchase**

CART page : you can see products that be chosen ( quantity, name, price and cart total ….). You can delete and add quantity from example below:

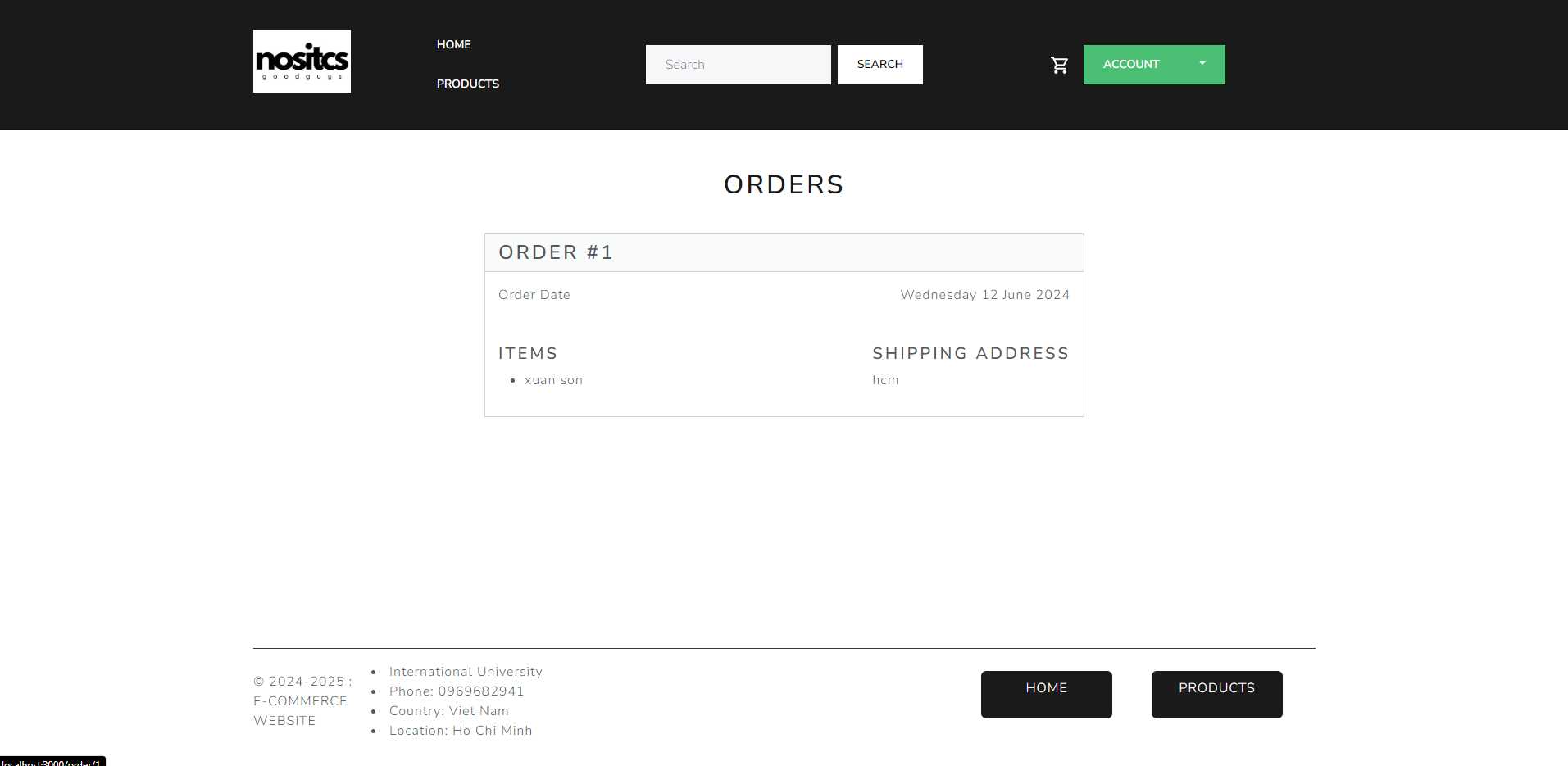


*Figure 8-11 Product Purchase*

1. **Order Products**

The order page show table include:

* Date ordered.
* Name of items.
* Shipping address



*Figure 8-12 Order Product*

1. **Frequently Asked Questions (FAQ):**

* If you encounter any issues or have any questions that need clarification, please don't hesitate to reach out to our support team. We would be delighted to assist you and ensure that any concerns you may have been addressed promptly and effectively.

You can contact us via the link at the bottom of the website (specifically the Contact Us section) if you encounter any problems.



*Figure 8-13 FAQ 1*

# Project Evaluation

## Evaluation Criteria Used to Measure Project Success:

1. **Completion of Deliverables:**

* **Evaluation:** The project has successfully delivered a fully functional e-commerce website and admin panel, meeting all specified objectives.
* **Measurement:** Deliverables include a user-friendly website with comprehensive features, an efficient admin panel, and robust security protocols, all of which meet the initial project specifications and goals.

1. **Adherence to Timeline:**

* **Evaluation:** The project adhered to the proposed timeline, achieving significant milestones as planned.
* **Measurement:** The Gantt chart and project timeline reflect the timely achievement of key phases, indicating effective project management and scheduling practices.

1. **Budget Adherence:**

* **Evaluation:** The project was completed within the allocated budget.
* **Measurement:** Financial reports show efficient resource management, ensuring all expenses remained within the planned budget, demonstrating prudent financial oversight.

1. **Stakeholder Satisfaction:**

* **Evaluation:** Stakeholder feedback has been overwhelmingly positive, recognizing the project's user-friendly design and robust functionality.
* **Measurement:** Surveys and direct feedback from stakeholders, including clients and team members, confirm high satisfaction levels with the project outcomes.

1. **Quality of Deliverables:**

* **Evaluation:** The project maintained high-quality standards in code, documentation, and overall implementation.
* **Measurement:** Regular code reviews, adherence to coding standards, and automated quality checks have ensured a high standard of code quality, comprehensive documentation, and well-implemented functionalities.

1. **User Adoption:**

* **Evaluation:** The project has achieved successful user adoption, evidenced by features such as secure payment gateways and effective order tracking.
* **Measurement:** User engagement metrics and positive user feedback indicate successful adoption and utilization of the website's features.

## Assessment of Whether Project Goals Were Achieved:

1. **Goal Alignment:**

* **Assessment:** The project goals were consistently aligned with the overall business objectives, focusing on enhancing customer experience, admin efficiency, and security measures.
* **Measurement:** Documentation and final deliverables confirm that project goals were aligned with and met the overarching business objectives.

1. **Functional Goals:**

* **Assessment:** The implemented functionalities met the predefined project goals, including seamless product browsing, secure payment processing, and efficient order tracking.
* **Measurement:** Comparison of achieved functionalities with initial requirements confirms that all functional goals were met.

1. **Client Satisfaction:**

* **Assessment:** Clients are highly satisfied with the final deliverables, appreciating the user-friendly interface and robust security features.
* **Measurement:** Client feedback, collected through meetings and surveys, reflects high satisfaction and approval of the project's outcomes.

1. **Operational Efficiency:**

* **Assessment:** The project has enhanced operational efficiency by streamlining product, order, and inventory management.
* **Measurement:** Key performance indicators (KPIs) demonstrate significant improvements in operational processes, validating the project's impact on efficiency.

## User Feedback and Satisfaction:

1. **Usability and User Experience:**

* **Assessment:** User feedback highlights a positive experience, with users appreciating the intuitive navigation and smooth purchasing process.
* **Measurement:** Surveys, interviews, and usability tests consistently show high user satisfaction with the website's design and functionality.

1. **Feature Satisfaction:**

* **Assessment:** Users are particularly satisfied with specific features, including product browsing, secure payment methods, and order tracking.
* **Measurement:** Detailed user feedback analysis reveals positive responses to these key features, indicating their effectiveness and user appeal.

1. **Response Time and Performance:**

* **Assessment:** Performance testing confirms the website's responsiveness and stability, even during high traffic periods.
* **Measurement:** User feedback on system responsiveness and performance testing results align, demonstrating the website's reliability and efficiency.

1. **Problem Resolution:**

* **Assessment:** Minor issues and bugs identified during testing were promptly resolved, ensuring a smooth user experience.
* **Measurement:** Tracking the resolution time for reported issues shows effective problem-solving and maintenance, contributing to overall user satisfaction.

1. **Overall User Satisfaction:**

* **Assessment:** Users are highly satisfied with the e-commerce website and admin panel, finding them user-friendly and efficient.
* **Measurement:** Aggregated feedback from various channels provides a comprehensive satisfaction score, reflecting the project's success in meeting and exceeding user expectations.

# Lesson Learned

Reflecting on our journey to develop a complete e-commerce website and admin panel, we have gained numerous insights that highlight our successes, challenges, and areas for future improvement.

## Reflection on What Went Well:

Our team successfully managed the transition to React JS, and Spring boot, showcasing our resilience and adaptability. Our iterative approach to requirements analysis and updates proved effective, allowing us to integrate valuable feedback from users and stakeholders seamlessly. Additionally, our commitment to open and transparent communication fostered a collaborative environment, ensuring alignment with project objectives and facilitating smooth adaptation to changes.

## Insights Gained from Project Challenges:

One significant challenge was the steep learning curve associated with the initial technology stack. This prompted us to switch to more beginner-friendly technologies, underscoring the importance of aligning the tech stack with the team's skill levels. This experience highlighted the need for regular skill assessments and proactive adjustments to our technical foundations to maintain efficiency and progress.

## Recommendations for Future Projects:

Future improvements and enhancements for the E-commerce website can include the following aspects:

1. **Reviews and Ratings of Products:** Implement a customer review and rating system where shoppers can provide feedback and rate the products they have purchased. This feature helps build trust, provides social proof, and assists other customers in making informed purchase decisions.
2. **Sorting Products:** Enhance the website's search functionality to allow customers to sort and filter products based on various criteria such as price, color, brand, size, and popularity. This improves the user experience by enabling more efficient product discovery.
3. **Manufacturer Information and Reviews:** Provide detailed information about the manufacturers of the sneakers, including their reputation, history, and brand values. Incorporate customer reviews and ratings specific to the manufacturers to help customers assess the quality and reliability of the products.
4. **Analytics and Reporting for Admin Page:** Develop an analytics and reporting system for the admin page to track key performance indicators, sales data, customer behavior, and other relevant metrics. This data can provide valuable insights for decision-making, identifying trends, and optimizing the website's performance.
5. **Returns and Refunds Policy for Checkout:** Implement a clear and comprehensive returns and refunds policy, including procedures for customers to initiate returns, request refunds, and track the status of their returns. This helps streamline the customer support process and ensures a smooth experience in case of product returns.
6. **Scalability and Performance Upgrades:** As the business grows, focus on scalability and performance improvements to handle increasing traffic and transactions. This may involve optimizing server infrastructure, implementing caching mechanisms, and utilizing content delivery networks (CDNs) to enhance website speed and reliability.
7. **Admin Panel** :The Advanced Inventory Management System is designed to provide the admin with robust tools for overseeing and managing the entire inventory of the e-commerce platform. This feature ensures that inventory levels are accurately tracked, stock levels are optimized, and inventory-related tasks are streamlined to improve operational efficiency and reduce manual errors.
8. **Additional Features and Integrations:** Continuously assess customer needs and industry trends to identify potential additional features and integrations. For example, incorporating virtual try-on functionality, interactive shoe size guides, integration with social media platforms, or collaborations with influencers or athletes to promote the brand.

By focusing on these future improvements and enhancements, the E-commerce website can enhance the user experience, increase customer satisfaction, and stay competitive in the market.

# Conclusion

In concluding our project journey to build a comprehensive website for a store , we reflect on the significant outcomes, key achievements, and the collective efforts that brought this endeavor to fruition.

## Summary of Project Outcomes:

The project culminated in the successful development of a fully functional online store. The website offers an intuitive user experience, enabling customers to explore, select, and purchase products.

## Reiteration of Key Achievements:

Our achievements stem from a collaborative and adaptable approach. The iterative development process, guided by user stories and use cases, ensured the end-user experience remained central. The decision to pivot to more beginner-friendly technologies showcased our team's resilience and commitment to overcoming challenges. Clear communication channels, documented changes, and an agile project management approach were crucial to our success.

## Closing Remarks:

As we conclude this project, we express our gratitude for the dedicated efforts of each team member. The project's success is a testament to our collective commitment to learning, adaptability, and innovation. We view the lessons learned not as obstacles but as stepping stones for continuous improvement. The skills developed, challenges overcome, and relationships built throughout this project lay a strong foundation for future endeavors.

# References

1. **Spring Boot**
   * Official website: [Spring Boot](https://spring.io/projects/spring-boot)
2. **Spring MVC**
   * Official documentation: [Spring MVC](https://docs.spring.io/spring-framework/reference/)
3. **MySQL**
   * Official website: [MySQL](https://www.mysql.com/)
4. **React**
   * Official website: [React](https://reactjs.org/)
5. **JWT (JSON Web Tokens)**
   * JWT Introduction: [JWT.io](http://jwt.io)
6. **RESTful API Design**
   * Richardson Maturity Model: [RESTful API Design](https://viblo.asia/p/restful-api-design-best-practices-LzD5dLVW5jY)
7. **Bootstrap**
   * Official website: [Bootstrap](https://getbootstrap.com/)
8. **HTML/CSS/JavaScript**
   * Tutorials and references: [W3Schools Online Web Tutorials](https://www.w3schools.com/)
9. **Vite JS**
   * Tutorials and references: [Vite](https://vitejs.dev/guide/)