**A PROJECT REPORT**

**ON**

**BLOCKCHAIN APPLICATION DEVELOPMENT**

Version 1.0

FOR

**BOT TRENDING AND BOTCHECK SMART CONTRACTS**

BY

Bui Thien Chu

Tran Anh vy

**Acknowledgements**

Mr. Ho Nguyen Phu Bao (industrial mentor) and Mr. Le Minh Hoa are to be thanked from the bottom of our hearts for providing us with the chance to work under their supervision and for assisting us in gaining enormously valuable professional experience. Mr. Do Thanh Sang, Project In charge and our internal project guide, provided us with invaluable feedback and suggestions from the selection of the project's theme until its successful conclusion.

We appreciate Mr. Vo Thanh Duy's continued support and encouragement in every way. Lastly, we would like to thank the whole Department of IT team for guiding our ideas and vision.

Without the assistance of our mentors, family, and friends, we would not have been able to complete our project successfully.

Bui Thien Chu

**Table of Contents**

[***Chapter 1 Introduction 1***](#_heading=h.30j0zll)

[**1.1 Introduction about the project subject 1**](#_heading=h.2et92p0)

[**1.2 Project objectives 2**](#_heading=h.tyjcwt)

[**1.3 Project plan 3**](#_heading=h.upglbi)

[**1.4 Project outcomes 4**](#_heading=h.1tuee74)

[**1.5 Project evaluation 5**](#_heading=h.4du1wux)

[***Chapter 2 Literature Review 6***](#_heading=h.2szc72q)

[**2.1 What is an E-Commerce Web Application? 6**](#_heading=h.184mhaj)

[**2.2 The related work when designing E-Commerce Web Application 7**](#_heading=h.3s49zyc)

[**2.3 The use of E-Commerce Web Application 8**](#_heading=h.279ka65)

[2.3.1 Accept orders 9](#_heading=h.meukdy)

[2.3.2 Accept payment 9](#_heading=h.36ei31r)

[2.3.3. Handle shipping and logistics 9](#_heading=h.1ljsd9k)

[2.3.4 Provide customer service 10](#_heading=h.45jfvxd)

[**2.4 How social media (Facebook, Google) and other big e-wallet companies currently support E-Commerce Web Application? 10**](#_heading=h.2koq656)

[**2.5 What are the differences between DusTin and another E-Commerce Web Application? 12**](#_heading=h.zu0gcz)

[**2.6 How is personal information protected in a project's application? 12**](#_heading=h.3jtnz0s)

[***Chapter 3 Technology and Tools 15***](#_heading=h.1yyy98l)

[**3.1 Front-end technologies 15**](#_heading=h.4iylrwe)

[**3.2 Back-end technologies 16**](#_heading=h.2y3w247)

[**3.3 Database 17**](#_heading=h.1d96cc0)

[***Chapter 4 Software Product Requirements 17***](#_heading=h.3x8tuzt)

[**4.1 Review/overview of other similar products 17**](#_heading=h.2ce457m)

[**4.2 Use Case Diagrams/User Stories 18**](#_heading=h.rjefff)

[**4.3 Use Case Specifications/Activity Diagrams & Context Diagrams/Sequence Diagrams 19**](#_heading=h.1qoc8b1)

[**4.4 ERD 33**](#_heading=h.3u2rp3q)

[**4.5 Sitemap 34**](#_heading=h.odc9jc)

[***Chapter 5 Review of Software Development Methodologies 34***](#_heading=h.1nia2ey)

[**5.1 Waterfall 35**](#_heading=h.47hxl2r)

[5.1.1.](#_heading=h.11si5id) Advantages of Waterfall methodology 36

[5.1.2.](#_heading=h.3ls5o66) Disadvantages of Waterfall methodology 36

[**5.2 Spiral 36**](#_heading=h.20xfydz)

[5.2.1.](#_heading=h.302dr9l) Advantages of Spiral methodology 37

[5.2.2.](#_heading=h.1f7o1he) Disadvantages of Spiral methodology 37

[**5.3 RAD 38**](#_heading=h.3z7bk57)

[5.3.1.](#_heading=h.thw4kt) Advantages of RAD methodology 39

[5.3.2.](#_heading=h.3dhjn8m) Disadvantages of RAD methodology 39

[**5.4 Agile 39**](#_heading=h.1smtxgf)

[5.4.1.](#_heading=h.4cmhg48) Agile process 40

[5.4.2.](#_heading=h.16x20ju) Advantages of Agile methodology 42

[5.4.3.](#_heading=h.3qwpj7n) Disadvantages of Agile methodology 42

[**5.5 Selected software development methodologies with justification 43**](#_heading=h.261ztfg)

[***Chapter 6 Design and Implementation of your demo product 46***](#_heading=h.l7a3n9)

[**6.1 Product Analysis and Design 46**](#_heading=h.356xmb2)

[**6.2 Features include with screenshots 56**](#_heading=h.4hr1b5p)

[**6.3 Product Implementation 62**](#_heading=h.2olpkfy)

[**6.4 Evaluation of the product 69**](#_heading=h.13qzunr)

[***Chapter 7 Conclusions 72***](#_heading=h.3nqndbk)

[**7.1 Learned from the project 72**](#_heading=h.22vxnjd)

[**7.2 Result of the project 73**](#_heading=h.i17xr6)

[**7.3 Further development of this project 75**](#_heading=h.320vgez)

[***REFERENCE 76***](#_heading=h.1h65qms)

**Table of Figures**

[Figure 1 Project plan 3](#_heading=h.4d34og8)

[Figure 2 Use case diagram 18](#_heading=h.49x2ik5)

[Figure 3 Use case login 19](#_heading=h.3o7alnk)

[Figure 4 Use case register 22](#_heading=h.1hmsyys)

[Figure 5 Use case order 24](#_heading=h.vx1227)

[Figure 6 Use case customize 27](#_heading=h.4f1mdlm)

[Figure 7 Use case recharge 29](#_heading=h.19c6y18)

[Figure 8 Use case management 31](#_heading=h.28h4qwu)

[Figure 9 ERD 33](#_heading=h.37m2jsg)

[Figure 10 Sitemap 34](#_heading=h.46r0co2)

[Figure 11 Waterfall Methodology 35](#_heading=h.3l18frh)

[Figure 12 Spiral methodology 37](#_heading=h.1egqt2p)

[Figure 13 RAD methodology 39](#_heading=h.3cqmetx)

[Figure 14 Agile methodology 41](#_heading=h.3q5sasy)

[Figure 15 Home page GUI 46](#_heading=h.2iq8gzs)

[Figure 16 Profile GUI 48](#_heading=h.xvir7l)

[Figure 17 Shopping Cart GUI 50](#_heading=h.3hv69ve)

[Figure 18 Customize GUI 52](#_heading=h.ymfzma)

[Figure 19 Custom shoe list GUI 54](#_heading=h.4h042r0)

[Figure 20 Product list GUI 55](#_heading=h.2w5ecyt)

[Figure 21 Customize interface 56](#_heading=h.3vac5uf)

[Figure 22 Transfer UI 57](#_heading=h.2afmg28)

[Figure 23 Management UI 58](#_heading=h.pkwqa1)

[Figure 24 Create shoe UI 59](#_heading=h.39kk8xu)

[Figure 25 Update shoe information UI 60](#_heading=h.1opuj5n)

[Figure 26 Deleted shoe table 60](#_heading=h.48pi1tg)

[Figure 27 Login UI 61](#_heading=h.2nusc19)

[Figure 28 Shopping cart UI 62](#_heading=h.1302m92)

**Table of Tables**

[Table 1 Login use-case 19](#_heading=h.2pta16n)

[Table 2 Login use-case (case 1) 20](#_heading=h.14ykbeg)

[Table 3 Login use-case (case 2) 21](#_heading=h.3oy7u29)

[Table 4 Register use-case 22](#_heading=h.j8sehv)

[Table 5 Register use-case (case 1) 23](#_heading=h.338fx5o)

[Table 6 Order use-case 24](#_heading=h.42ddq1a)

[Table 7 Order use-case (case 1) 25](#_heading=h.2hio093)

[Table 8 Customize use-case 27](#_heading=h.3gnlt4p)

[Table 9 Recharge use-case 29](#_heading=h.4fsjm0b)

[Table 10 Management use-case 31](#_heading=h.1a346fx)

# Chapter 1 Introduction

## 1.1 Introduction about the project subject

With the development of modern technology, all tools and applications to help people become closer and easier to interact with are growing. Buying and selling is no exception when in recent years with the development of the Internet, online shopping websites and applications have been developed and used strongly and widely.

Through research and survey, we decided to design a web application to help people feel more comfortable and convenient when shopping online. Along with some basic essential functions that must be in an online shopping application, we join and implement some special functions to create a change in the thinking and understanding of everyone. people about the feature a trading application can do.

The project has been planned and designed since February 2022 and is implemented and developed to the present time with the participation of me - Nguyen Tien Thanh and Le Dinh Dong. The planned project is to create an online shoe selling application with 2 main roles as store owner and customer, along with basic functions such as payment by bank card, e-wallet or login. by social media applications.

With the understanding and knowledge that have been acquired in the learning process and through the lecturers at the university, this project we decided to use the Back-End technology of NodeJS along with HTML and CSS for the Front-End technology. Together with MongoDB for noSQL to store data into the database. The source code we use is hosted on Github and uses MVC models to structure the Folder. And last, the project is deployed by Heroku on <http://dustinshoe.herokuapp.com>.

For the software development methodologies, we choose waterfall to design the project and develop the application and also manage the work of each member in the team.

## 1.2 Project objectives

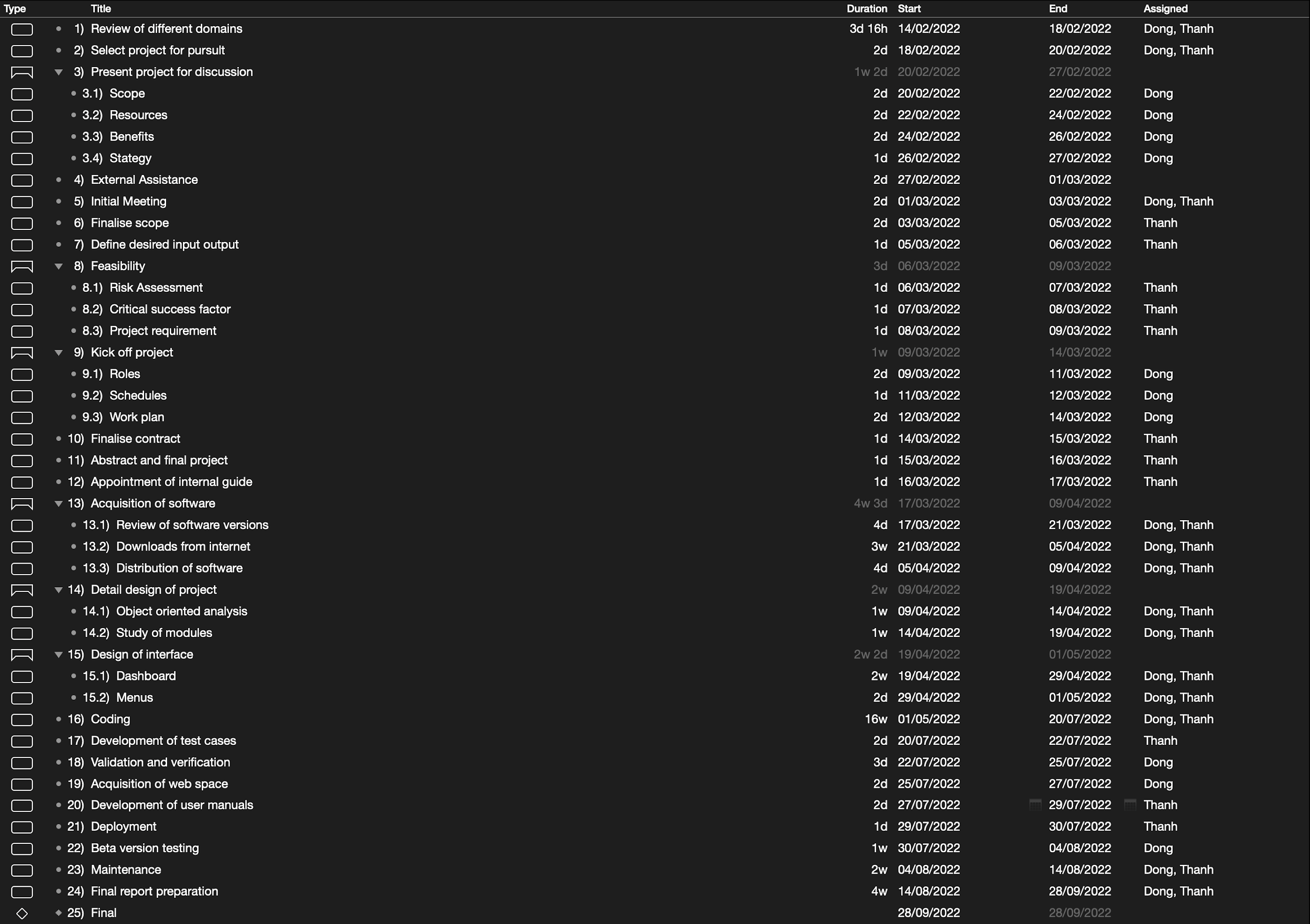
Current Web usability recommendations are damaging to the usability of web-based apps since they are mostly built on interactions inside a browser paradigm. Traditional internal software is distinguished by the fact that the user focuses completely on the program. All other software-related activities, including as administration, monitoring, and upgrades, are handled by the IT department or network administrator. Create websites for the Internet utilizing our learned information, skills, and talents.

The primary application component is the emphasis of a SaaS (Software as a Service) application, which also contains duties that comprise the SaaS application and business model as a whole. A SaaS application is aimed at the client much more narrowly than a typical application, which is aimed at the entire public. This includes registering, purchasing the product, using assistance, and personalizing the application. Consequently, design and interaction are increasingly appreciated.

That's why our project is focused on interface design and user experience, the main goals that we have set for this project before it is planned and implemented are:

* Use fundamental design principles to present ideas, information, products, and services on websites.
* When developing websites, use fundamental programming concepts.
* Utilize the tools available to you to efficiently manage website projects.
* demonstrate our ability to present, communicate, and manage services.
* Complete the necessary preparations by producing goods, selling them online, designing custom shoe models, and trading.
* Apply fundamental knowledge, self-management skills, and teamwork skills to our job search.

## 1.3 Project plan



*Figure 1 Project plan*

## 1.4 Project outcomes

An e-commerce website DusTin Store was created to improve the existing service for those living on college campuses by making it more dependable, efficient, and convenient. Maintaining consumer satisfaction and happiness through a dependable and practical purchasing approach is the only way to maintain business and draw in more clients. The frontend and the backend are its two parts. Users can view things that are offered in-store, place orders, and maintain their profiles from the front end. The system's backend keeps track of both customer orders and sales that have been made. It oversees the home delivery service for all clients that live on campus.

Additionally, the system maintains track of expenditures and purchases made for various items. It may determine the overall quantity of purchases and sales as well as the most and least popular things. Thus, this will lessen the effort of the service providers by enabling them to obtain an instant statistic while conducting a cost-benefit analysis of their company.

By putting the DusTin project into practice, we hope to advance the aforementioned idea and give users the possibility to create web pages online without having to install any special software to do so. It is a stand-alone product that enhances the conditions that now exist in web development.

* User freedom and control: A user will be able to support undoing and then redoing an activity to fix mistakes.
* DusTin strives to make the user's working environment and experience as aesthetically pleasing as possible. DusTin is therefore created as an RIA (Rich Internet Application)
* Help, prompts, and documentation are all provided by DusTin in the form of embedded, clear-cut instructions and notifications.

## 1.5 Project evaluation

To evaluate this e-commerce website project, I will briefly review 3 main parts: security, user experience and outstanding features.

In terms of security, the website was planned and implemented and completed within 6 months from the date of starting the project. However, the project only ensures the full implementation of the basic features of an e-commerce website and design an interface that is close and easy to use for customers. Do not focus too much on increasing the security of the website's personal information and data. With limited time, the project also does not focus too much on improving the security of the project, making the actual implementation will face many risks such as stealing and destroying customer and system information.

In terms of user experience, we have designed the website as a Website Applcation so it will be accessible on many popular devices such as PC, Laptop, Tablet and Mobile. With our understanding of user behavior and research from previous projects, we design user-friendly interfaces and notifications, tools and dashboards with notes and hints. make it easy for users to access and use. We also choose to use English instead of Vietnamese because English is currently one of the international languages ​​and is used by many people.

In terms of outstanding features, we choose Customize to design a function that helps users create a pair of shoes according to their personal preferences. From there, users can go to their personal inventory or order shoes, and the store will send information about the modified shoes to the manufacturer so they can design it according to the customer's requirements. Because of the limited time, we only implemented it in 2D with the RGBA color palette from the website. However, we are currently researching and developing a 3D design feature with 16 million colors from RGBA and some colors suggested by the system, and can engrave on the shoes that customers want to customize. Along with that, there will be a page to display hot custom created by users for customers to rate and then appear on the top of the page for other customers to choose and order.

# Chapter 2 Literature Review

## 2.1 What is an E-Commerce Web Application?

Before finding out what E-Commerce Web Application and what can it be used in the present with the strong development of the Internet, we will learn about what is E-commerce.

E-commerce, commonly referred to as electronic commerce, is the process of buying and selling goods over the internet as well as the data and money transfers necessary to execute those transactions. E-commerce wasn't well-known in the beginning, but as the use of mobile phones increased and more people expressed interest in shopping online, it quickly gained popularity.

Types of E-commerce: E-commerce business models primarily come in four different types.

* **B2C (Business to consumer):** It is an online firm, and utilizing this model, we may offer things to consumers.
* **B2B (Business to business):** In contemporary modern times, the majority of e-commerce belongs into this category. It is a business that will occur between huge firms, organizations, and businesses.
* **C2B (Consumer to Business):** This form of online trade enables people to sell their products to businesses. Individuals would assign tasks to perform in the allotted period using this kind of online commerce. The clearest illustration of this kind of employment is freelancing. Customers can establish the price for their services on websites or other electronic platforms.
* **C2C (Consumer to Consumer):** This form of e-commerce links consumers to consumers so that they can exchange things and earn money by charging transactions.

In the research on design and development of E-Commerce Web Application for Cooperative Store, Sangay Tenzin and his team conclude that: "Much like a traditional physical retail store, e-commerce websites allow consumers and businesses to buy and sell to one another on a designated platform. The main difference between e-commerce and physical commerce, however, is that e-commerce transactions occur entirely over the internet rather than at a brick-and-mortar location" (Design and Development of E-Commerce Web Application for Cooperative Store et al., 2022, p. 843).

Customers may purchase online using the Cooperative Store Management System, an e-commerce web application, without having to physically visit the store. By automating record entry, the company sought to eliminate salesperson effort and human mistakes. Customers may dramatically cut expenses and lost time by using a system. In addition, since the services are accessible from home, users may enjoy greater service and make purchases at their leisure. This will contribute to the company's long-term success by attracting new customers and retaining current ones. To collect all of the essential data for a system, team meetings, interviews, and brainstorming sessions were conducted. Depending on their needs, customers may evaluate the products and make orders using the frontend interface. Using the backend interface, store personnel may perform Create, Read, Update, and Delete (CRUD) operations. The document will provide information on the techniques utilized during the application's development. In addition, it will describe the system's methodology, design, and general functioning.

## 2.2 The related work when designing E-Commerce Web Application

S. E. Ullah, T. Alauddin and H. U. Zaman suppose that the most affordable method of conducting business is through the use of an e-commerce platform, which also lowers the price of advertising goods and services (Developing an E-Commerce Website., 2016). Following that, the primary goal of E-Commerce Web Applications development of the application is to draw more users in order to boost sales. M. S. Kandhari, F. Zulkernine and H. Isah (2018) have created a speech recognition-based e-commerce web application to take advantage of accessibility for consumers who are visually challenged. This enables people with disabilities to utilize the application via voice commands. Therefore, this one also makes the developers need to design more features for the website. The increase of their code also makes the website become low performance. Additionally, applications with SRS capabilities can improve usability for all users and enable consumers to submit requests in natural language.

A web application for online shopping using the MERNstack technology, which consists of MongoDB, the Express.JS framework, the Bootstrap library, and the Node.JS platform (Naidu et al., 2021). Along with offering several views for users and administrators, the application also incorporates a payment gateway for checkout. Seeing that benefits from developing the website with those technologies and framework, we decided to make our website – Dustin by them. The accuracy and business performance of e-commerce applications have been improved by P. Umberto in 2015 by developing a price-sensitive recommendation engine. It was discovered that good recommendation engine modeling enhances business performance. Additionally, in Vietnam, where physical buying is prohibited because to COVID-19 around the globe, internet shopping and home delivery have grown commonplace. In the research of M. B. Subba, D. Pem (2020) in their own project more than 20 stores were given permission to cater and offer services, such as food and grocery delivery. Kuensel (2019) states that a platform called Vietnam buy was established to enable online purchases of Vietnamese goods by all Vietnamese residing anywhere in the world.

A website that allows users to buy and sell tangible items, services, and digital products online as opposed to in person. Through an e-commerce website, a business can process orders, accept payments, manage shipping and logistics, and provide customer service.

## 2.3 The use of E-Commerce Web Application

When it comes to selling goods and services online, every e-commerce site must have a few essential features.

### 2.3.1 Accept orders

If Dustin owns a company that sells things, the e-commerce website must be able to accept orders from customers. Therefore, each time a purchase is made, the website must:

* Maintain client information (name, address, etc.)
* Obtain consumer signatures on the terms of service.
* Perform the appropriate calculations for any applicable taxes.
* Apply discounts or coupons
* Generate tracking and order numbers
* Provide delivery information
* Process billing data Allow customers access to a payment gateway

### 2.3.2 Accept payment

The website must transition effortlessly to a payment gateway. Typically, payment gateways allow customers to:

* Select a payment method.
* Provide details such as payment card numbers and CVV codes
* For secure transactions, employ multi-factor authentication.

### 2.3.3. Handle shipping and logistics

After a website has processed a consumer's purchase and payment, the next step is delivery. It is your obligation as a seller to ensure that customers get the proper item on time and in perfect condition. Either the website may handle shipping and logistics internally, or it can collaborate with a third-party firm to expedite the process. The website should be able to reliably and consistently initiate the procedure regardless of the approach used.

On the website, a mechanism for accepting returned items must be established. This method, also known as reverse logistics, is just as important as outgoing logistics.

### 2.3.4 Provide customer service

Inevitably, the customers will need to communicate with the proprietor about their purchase or experience. They may be required to update their personal data, make a purchase modification, or return a damaged product. These customers will visit the website to contact the proprietor. The more straightforward developers make it for users to contact the site's owner—via a contact form or chat service, for example—the easier it will be to quickly fix their difficulties.

## 2.4 How social media (Facebook, Google) and other big e-wallet companies currently support E-Commerce Web Application?

The world's expanding online market in recent years is evidence that digitization is progressing quickly. Vietnamese youth are particularly open to technology. Therefore, the potential of the mobile business market continues to rise as more individuals utilize social media and the internet on their smartphones, tablets, and other mobile devices. The use of social media platforms has improved audience reach. With the expansion of well-known websites like Facebook and Google, there is more content than ever. Because of this, it's critical for everyone to be able to target their marketing approach and differentiate themselves from the competition.

Aside from boosting traffic and revenue, social media also enables you to sell cheaply while demonstrating to clients how the goods or service will improve their lives. The most genuine and advantageous parts of the brand should preferably be highlighted in the material that you produce as part of your social marketing approach. You can use online tools like social analytics to monitor the effectiveness of social marketing strategies. Social media platforms are much more than just places for users to exchange and find information. They have developed into commercial platforms that are being used by more and more companies. Social media aids marketing strategy by making it easier to attract customers, close deals, and launch new marketing initiatives.

The ability of social media to reach consumers where they are is a crucial feature. Social media platforms can be used to better understand the audience. This makes creating campaigns that are appropriate for needs considerably simpler. Social media platforms are where consumers go to learn about goods and services. A good first impression for online business will be made by optimized profiles and helpful information.

Social media and e-wallets greatly influence an online business website today because of the benefits it brings.

* **Login with social media**: This is one of the important benefits that social media brings because with just one account you can login in many web, app, e-wallet platforms today. This helps users not to have to remember many accounts leading to forgetting accounts and information. Some people choose to use social media login sites because they feel lazy and uncomfortable creating a new account on that site, which is also an important reason to attract customers.
* **Payment by e-wallet**: As digital technology develops, people choose to use e-wallet more than regular banking because it is fast and convenient. For example, a customer who chooses to buy a product from a website only needs to choose e-banking payment and enter a certain password. However, it also has disadvantages if the website is a scam and will steal the information of the customer's e-banking card.
* **Advertising by social media**: When users choose to use social media, the website may ask to get some of their personal information necessary for marketing convenience. For example, if the user is a Nike shoe enthusiast, the site will display more Nike shoes. This is a marketing method used by many large websites such as Facebook, Google, YouTube, etc. to attract users and enhance the experience. For example, if you search with Google for Adidas shoes, Google will send that information to your shop website from which you can show or suggest to that user.

## 2.5 What are the differences between DusTin and another E-Commerce Web Application?

Basic features with a business website are important, so not only DusTin but all websites today have such as: Registration, login, shopping cart, order, payment, etc.

The difference that Dustin wants to make is Custom shoes. In the past 10 years, young people have become too familiar with Sneakers (sports shoes, commonly referred to as shoes designed for sports or outdoor activities). They are willing to pay large sums to collect rare, limited-edition shoes. This has created a movement about street style not only in Vietnam and around the world. Young people are ready to 'camping' to hunt for new or limited-edition shoes that a store or brand creates. However, for some people who do not like to wear a shoe like others, they have redrawn the textures, changing the texture of an ordinary shoe to make it special and a unique product of their own. From there the term 'Custom Shoes' was born.

So, with our website instead of selling ordinary shoes, we have created a special function called 'Customize' so that users can be creative in terms of styles and colors on shoes. they want to then be able to order the store to make one. With limited time, we are currently developing the 2D customizer function first, and will be 3D in the future.

## 2.6 How is personal information protected in a project's application?

Before answering that question, we will learn about what personal information is normally included in a business website. The first indispensable will be the account and password. However, if you log in with social media, the account will probably be an account with social media (email or phone number), some pages allow users to create their own password accounts and then link to social media to have social media accounts. can get some other information like name, gender, date of birth, etc. After that, the user's personal information will also be saved in the database such as name, address, date of birth, phone number, gender, avatar, etc.

According to DPIA, all personal data needs to be protected to avoid theft of user information and destruction. For example, if users have their phone numbers stolen, they can hack in and get OTP codes or simply tele sales that annoy them personally. Or the home address, which they will use to place an order, owning that information may be underestimated in the stores that were used to place the order.

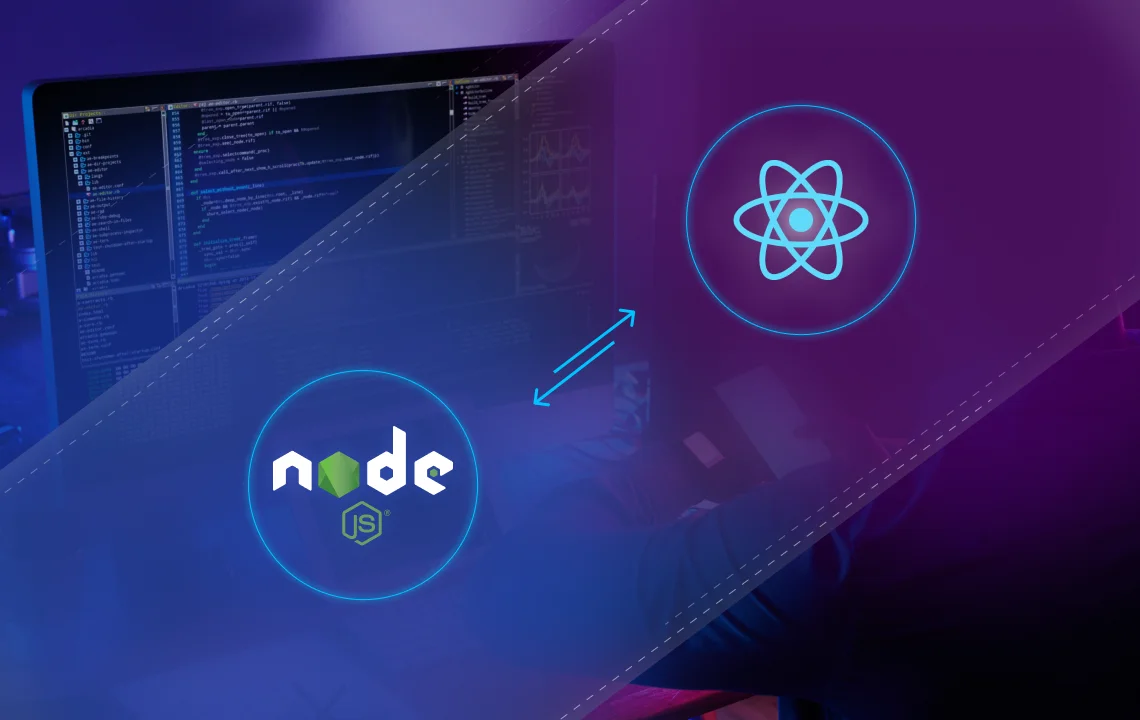
Regarding passwords, many projects choose to employ a password manager to generate and remember complicated passwords, similar to Google's current practice. LastPass and 1Password are the password managers preferred by the Wirecutter. Both can generate passwords, monitor accounts for security breaches, suggest weak passwords, and synchronize passwords between a computer and a mobile device. When you login in to your account, the password manager will save your password and prompt you to change any duplicate or weak passwords. Use two-step authentication so that even a hacker who knows your password and account cannot access it without authorization from your device.

For instance, the first step is to log in to Facebook using your login and password. In step two, Facebook provides you a temporary code by text message or, preferably, via an app such as Google Authenticator, which you input to sign in.

There are several storage and administration options available for safeguarding your data. You may use solutions to regulate access, monitor activity, and combat threats. The following are some of the most prominent approaches and tools:

* Discovery of data is the first stage in data protection. It involves identifying the existing data sets, those that are business-critical, and those that include sensitive data that may be subject to compliance rules.
* Data loss prevention (DLP) is a collection of strategies and tools that may be used to avoid data loss, theft, and accidental deletion. Data loss prevention solutions often include a variety of tactics for avoiding and recovering from data loss.
* Storage with built-in data protection - contemporary storage technology has disk clustering and redundancy built in. For example, Cloudian's Hyperstore provides minimum RTO/RPO, quick access, and up to 14 nines of durability at a cheap cost, enabling the storage of massive amounts of data. Our safe data storage guide has further details.
* By creating copies of data and keeping them independently, data backup allows data restoration in the case of data loss or modification. When original data is accidentally lost, destroyed, or corrupted, or when it is purposefully harmed, backups are essential for preserving business continuity. Our guide on data availability has further details.
* A snapshot is a full picture of a protected system, including data and system files. It is comparable to a backup. Using a snapshot, a system may be restored to a former state.
* Replication is the process of frequently transferring data from one safe system to another. This provides a copy of the data that is continually updated and allows for quick failover to the copy in the event that the main system fails.
* Firewalls are network traffic monitoring and filtering technologies. Firewalls may be used to ensure that only authorized users can see or transfer data.
* Authentication and authorisation are controls that let you validate credentials and guarantee that user rights are applied appropriately. In combination with role-based access restrictions as part of an identity and access management (IAM) system, these measures are often used (RBAC).
* Encryption modifies the content of data using a method that can only be reversed with the correct encryption key. Even if data is stolen, encryption renders it unreadable, preventing unauthorized access. The data encryption guide has further details.
* Endpoint security protects the ports, routers, and connected devices that serve as access points to your network. Using endpoint protection software, you can generally monitor the perimeter of your network and filter traffic.
* Data erasure decreases liability by removing obsolete information. This is possible when data is processed and analyzed, or periodically when data is no longer relevant. The erasure of unnecessary data is a requirement of several compliance requirements, notably the GDPR. Check out our guide for more GDPR information: Regarding privacy under the GDPR.
* Disaster recovery is a collection of policies and techniques that regulate how an organization reacts to a catastrophe, such as a natural disaster, cyberattack, or catastrophic equipment failure. Setting up a remote disaster recovery site with backups of catastrophe-resistant systems and transferring operations to these systems in the case of a disaster are typical phases in the recovery process.

# Chapter 3 Technology and Tools



**3.1.Front-End**

## -React: React is an open-source JavaScript library for building user interfaces. It is commonly used for building single-page applications and mobile applications. React allows developers to create reusable UI components and efficiently update the user interface in response to data changes. It is maintained by Facebook and has a large and active community of developers.

-HTML , CSS, Tiwins , Javascrip

## 3.2.Back-End

**-Node.js:** Node.js is an open-source JavaScript runtime environment that allows developers to run JavaScript code outside of a web browser. It is built on Chrome's V8 JavaScript engine and is commonly used for server-side development. Node.js provides a rich ecosystem of libraries and frameworks, making it a popular choice for building scalable and high-performance applications.

-**Solidity:** Solidity is a statically-typed programming language for writing smart contracts on the Ethereum blockchain. It is similar to JavaScript and provides a set of features for building secure and reliable smart contracts. Solidity is commonly used for building decentralized applications (dApps) and decentralized finance (DeFi) applications.

-**Truffle:** Truffle is a development framework for Ethereum that provides a suite of tools for building, testing, and deploying smart contracts. It includes a development environment, a testing framework, and a deployment tool. Truffle supports multiple programming languages, including Solidity, and provides a convenient way to manage the development lifecycle of smart contracts.

-**Ganache:** Ganache is a personal blockchain for Ethereum development. It allows developers to create a local blockchain environment for testing and development purposes. Ganache provides a set of features for managing accounts, deploying smart contracts, and inspecting transaction data. It is commonly used in conjunction with Truffle for Ethereum development.

-**Web3.js:** Web3.js is a JavaScript library for interacting with Ethereum nodes. It provides a set of features for sending transactions, querying blockchain data, and managing accounts. Web3.js is commonly used for building decentralized applications (dApps) and integrating Ethereum functionality into web applications.

-**Remix Ethereum IDE:** (Integrated Development Environment) is a popular tool for developing, testing, and deploying smart contracts on the Ethereum blockchain. It is an open-source browser-based IDE that supports multiple programming languages, including Solidity

**-Infura:** Infura is a cloud-based Ethereum node service that provides a scalable and reliable way to interact with the Ethereum blockchain. It provides a set of APIs for sending transactions, querying blockchain data, and managing accounts. Infura is commonly used for building decentralized applications (dApps) and integrating Ethereum functionality into web applications.

**-Ether.js:** Ether.js is a JavaScript library for interacting with Ethereum nodes. It provides a set of features for sending transactions, querying blockchain data, and managing accounts. Ether.js is similar to Web3.js but provides a more modular and flexible architecture. It is commonly used for building decentralized applications (dApps) and integrating Ethereum functionality into web applications.

**-Meta Mask:** provides a convenient way for users to manage their Ethereum accounts, private keys, and digital assets. It allows users to interact with decentralized applications (dApps) and smart contracts on the Ethereum blockchain, without the need for a full Ethereum node.

## -Etherscan.io: Etherscan.io is a blockchain explorer for Ethereum. It provides a set of APIs for querying blockchain data, including transaction data, account balances, and smart contract events. Etherscan.io is commonly used for debugging and monitoring Ethereum applications, as well as for integrating Ethereum functionality into web applications.

## These technologies are commonly used in the development of decentralized applications and blockchain-based systems. By mastering these technologies, developers can build innovative and cutting-edge applications that leverage the power of blockchain technology.

## 3.1 Front-end technologies

**HTML**: HTML is used to create the main text of a website, giving it structure. We begin by writing words, which we then decorate with tags or other ornaments. Our website can then be filled with content, as the web browser will be able to interpret the page's heading, paragraphs, and page boundaries.

Every browser that exists today supports HTML, which is used on virtually every website. It can be rather simple to learn how to code, I don't need any licenses, and I don't have to pay for it.

**CSS (**Cascade Styling Sheets): CSS is the skin that covers HTML, which is likened to the body's bones. It is used for background color, styling, layout, borders, and shadowing—all the crucial design elements that give a website a polished, sophisticated appearance. With CSS, you may alter the look and feel of HTML elements to differentiate between presentation and content.

By transforming how content appears on a webpage and what else is on it to complement that content, CSS has made a significant contribution to web design in terms of presentation and usability. Although it is usually used in conjunction with HTML, it may be used with any XML-based markup language and is independent of HTML.

**JavaScript** (the JS library used for front end development is jQuery)**:** It's incredibly adaptable and easy to learn. Other web technologies, such HTML and CSS, work well with it. JavaScript can be used to create both front-end and back-end web applications. JavaScript is frequently used to create straightforward web games. JavaScript can be used to make mobile applications.

## 3.2 Back-end technologies

**NodeJS**: A server-side JavaScript run-time environment is Node.js. It is open-source and contains a core library, libuv for cross-platform compatibility, and the V8 engine from Google. It is significant that Node.js does not offer a global "window" object since it is single-threaded and does not execute in a browser. Node.js is generally used for event-driven, non-blocking servers. Although it was designed with real-time, push-based architectures in mind, it is also used for standard web pages and back-end API services.

By using the ExpressJS as a framework in NodeJS, the web is built also with handlebars for API view in NodeJS.

**Handlebars (hbs)**: A more capable and user-friendly templating engine than the ejs module in node.js is handlebars.js. It guarantees minimal templating and is a logicless engine that maintains the separation of the view and the code. It is available through npm as the hbs module, which may be used with express. Handlebars can be used to produce web pages from data on the server side to the client side.

## 3.3 Database

**MongoDB:** For easy use in Rest API, MongoDB REST API is excellent for unstructured data because it's easy to set up and lets you store and retrieve documents.

One popular method of putting the MongoDB REST API idea into practice is by using Express JS as the backend web server and MongoDB as the document store. This method easily connects the Document Model of MongoDB with the JSON-based payloads of the REST API. Express enables me to create a Node.js-based backend middle tier that exposes REST API routes to my application.

# Chapter 4 Software Product Requirements

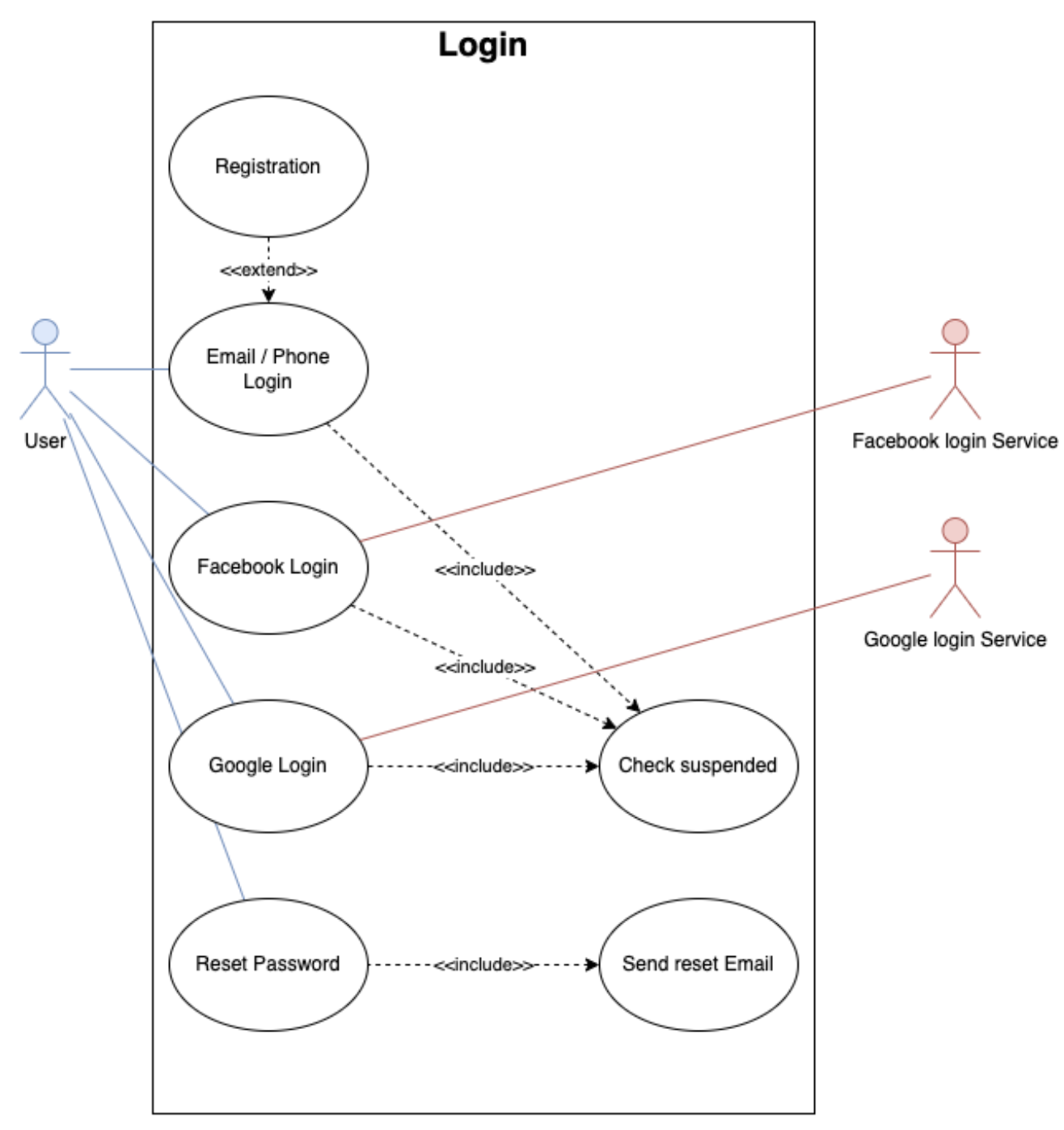
## 4.1 Review/overview of other similar products

## 4.2 Use Case Diagrams/User Stories



*Figure 2 Use case diagram*

## 4.3 Use Case Specifications/Activity Diagrams & Context Diagrams/Sequence Diagrams



*Figure 3 Use case login*

*Table 1 Login use-case*

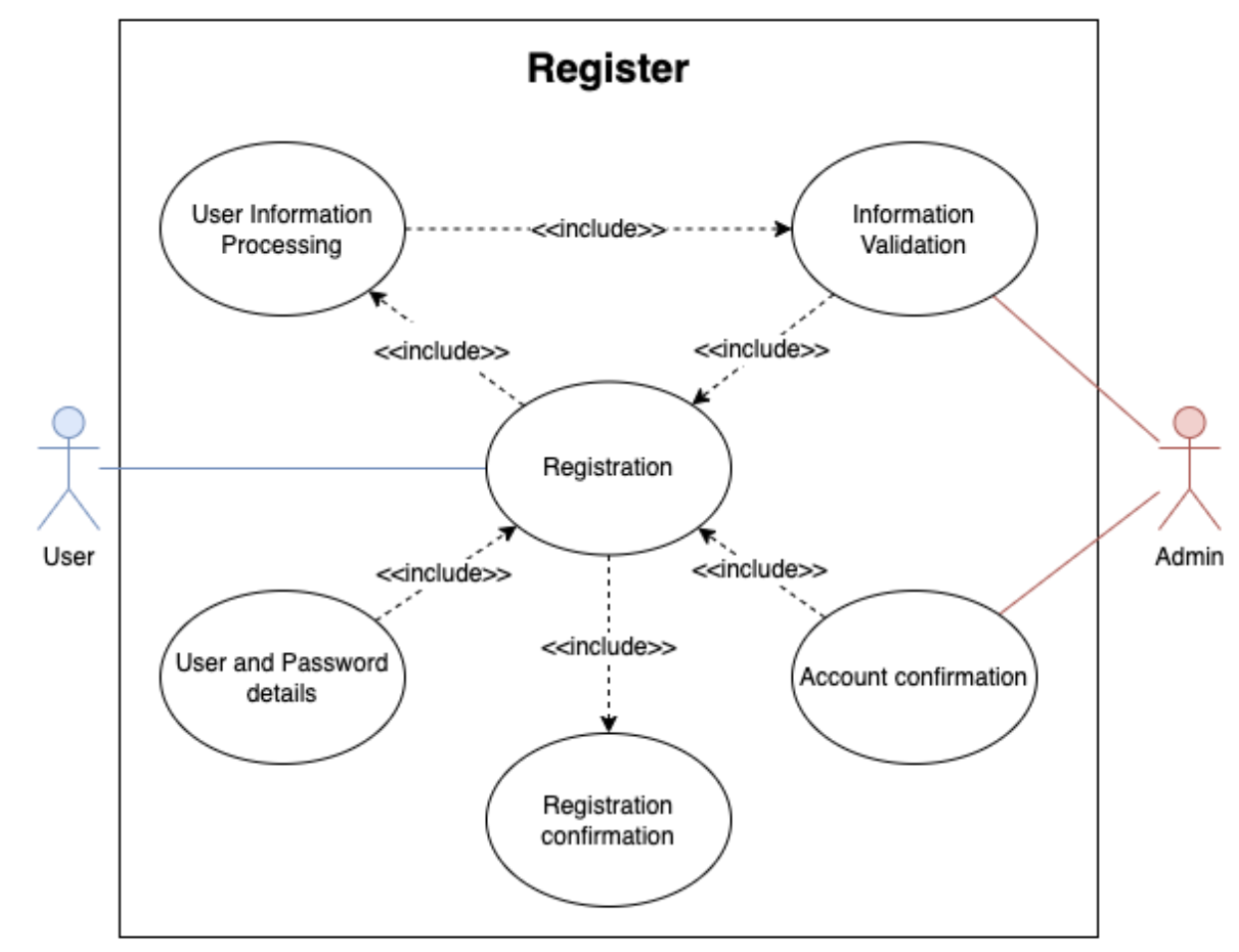
|  |  |
| --- | --- |
| **Use Case** | Login |
| **ID** | UC1 |
| **Description** | Users sign in to website |
| **Primary Actors** | Customers, Admins |
| **Secondary Actors** | None |
| **Preconditions** | User has already has an account in the database of the web application. |
| **Main Flow** | 1. When a user clicks the "Login" tab in the navigation bar, the use case begins. 2. Users can log in by entering their username, email address, and password on the login screen that is displayed by the system. 3. The user enters the information and selects "Sign In." 4. The program verifies the information in the database. 5. The user's profile page is shown by the system. |
| **Post Conditions** | The user has access to the system |
| **Alternative Flows** | Invalid Field(s)  Login by social media |

*Table 2 Login use-case (case 1)*

|  |  |
| --- | --- |
| **Use Case** | Login (Login by social media) |
| **ID** | UC1.1 |
| **Description** | Users sign in to website by social media |
| **Primary Actors** | Customers, Admins |
| **Secondary Actors** | None |
| **Preconditions** | User has already had an account on the social media |
| **Main Flow** | 1. When a user clicks the "Login by Facebook/Google" tab in the navigation bar, the use case begins. 2. Users can log in by Facebook, Google on the login screen that is displayed by the system. 3. The program verifies the information in the database. 4. The user's profile page is shown by the system. |
| **Post Conditions** | None |
| **Alternative Flows** | None |

*Table 3 Login use-case (case 2)*

|  |  |
| --- | --- |
| **Use Case** | Login (Invalid Field(s)) |
| **ID** | UC1.2 |
| **Description** | User tries to sign into the system with invalid details |
| **Primary Actors** | Customers, Admins |
| **Secondary Actors** | None |
| **Preconditions** | User has tried to sign in. |
| **Main Flow** | 1. The alternate flows begin after step 3 of the primary flow. 2. The system cannot locate any matching information in the database. 3. The login page is re-displayed with an invalid login message 4. The user may re-enter the proper information. |
| **Post Conditions** | None |
| **Alternative Flows** | None |



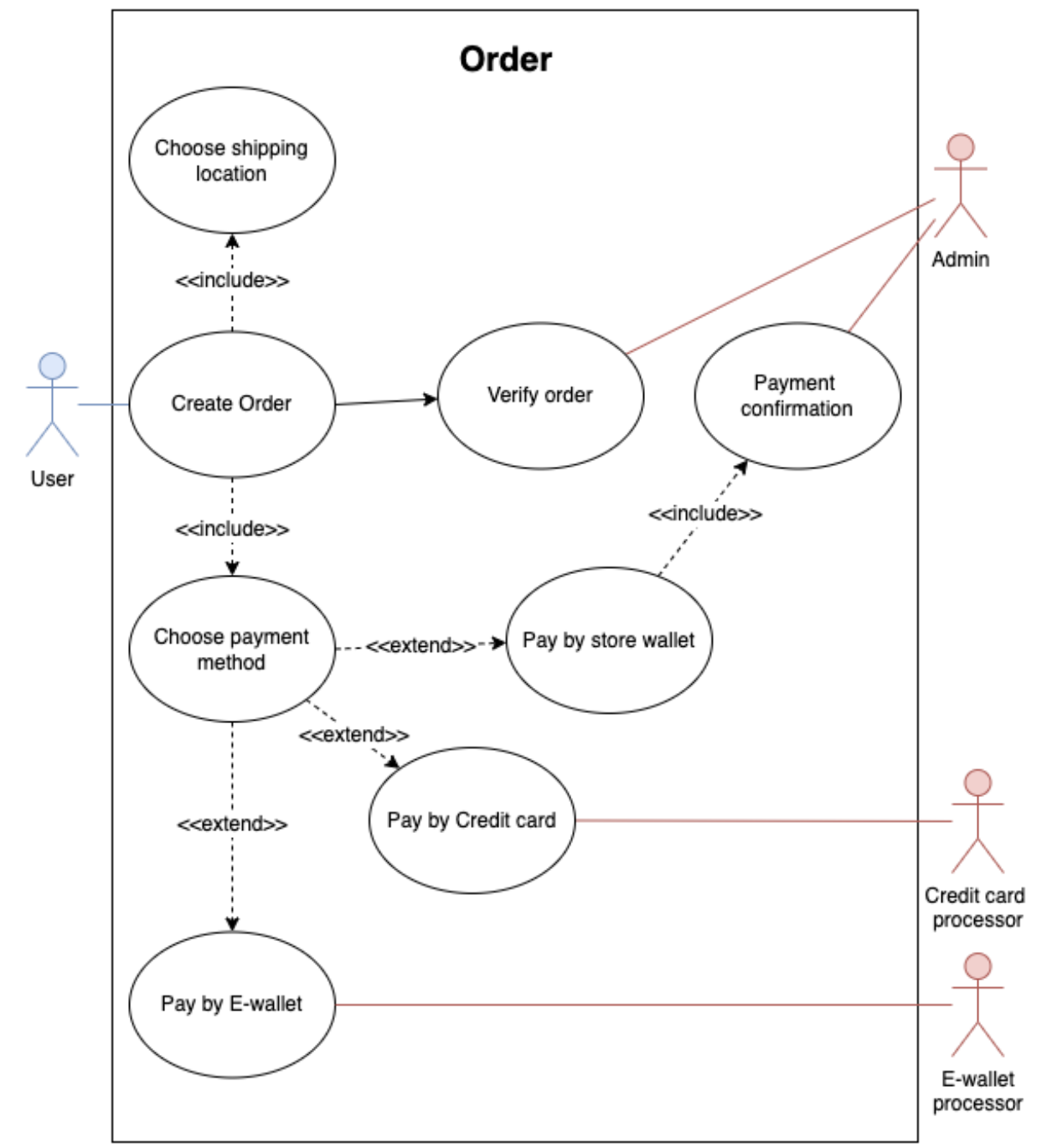
*Figure 4 Use case register*

*Table 4 Register use-case*

|  |  |
| --- | --- |
| **Use Case** | Register Account |
| **ID** | UC2 |
| **Description** | Users sign up to website |
| **Primary Actors** | Customers |
| **Secondary Actors** | None |
| **Preconditions** | None |
| **Main Flow** | 1. When a user selects the "Register" tab from the navigation bar, the use case begins. 2. The system shows the signup screen, where users can enter their password, username or email address, first and last name, and other information. 3. The user enters the information and selects "Sign Up." 4. The user's profile page is shown by the system. |
| **Post Conditions** | The user’s details are stored in the database. |
| **Alternative Flows** | Invalid field(s) |

*Table 5 Register use-case (case 1)*

|  |  |
| --- | --- |
| **Use Case** | Register Account (Invalid field(s)) |
| **ID** | UC2.1 |
| **Description** | User tries to sign up an account in the system with invalid details |
| **Primary Actors** | Customers |
| **Secondary Actors** | None |
| **Preconditions** | User has attempted to register for an account |
| **Main Flow** | 1. The alternative flows begin following step 3 of the primary flow. 2. The system cannot locate any matching information in the database. 3. The login page is re-displayed with an invalid login message 4. The user may re-enter the correct information. |
| **Post Conditions** | None |
| **Alternative Flows** | None |



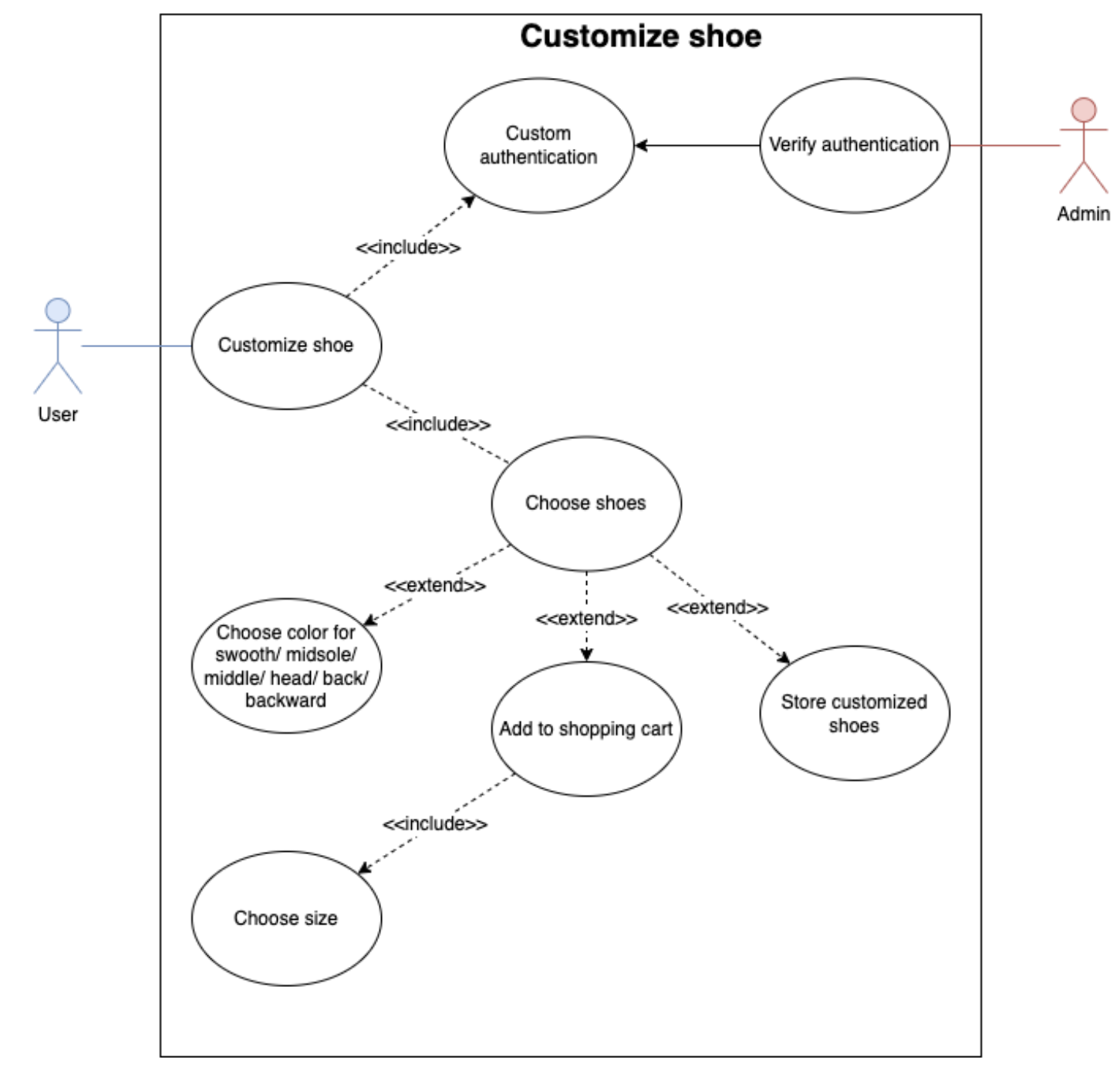
*Figure 5 Use case order*

*Table 6 Order use-case*

|  |  |
| --- | --- |
| **Use Case** | Order |
| **ID** | UC3 |
| **Description** | Users order products in the website |
| **Primary Actors** | Customers |
| **Secondary Actors** | None |
| **Preconditions** | User tries to order a shoes in the system |
| **Main Flow** | 1. When a user selects the "Cart" tab from the navigation bar, the use case begins. 2. The system shows the cart screen which the chosen shoes, total price and form to order. 3. The user enters the information and selects shipping location, payment method. 4. The notification is shown to notice. 5. The system saves the order |
| **Post Conditions** | The order is stored in the database. |
| **Alternative Flows** | Out of money |

*Table 7 Order use-case (case 1)*

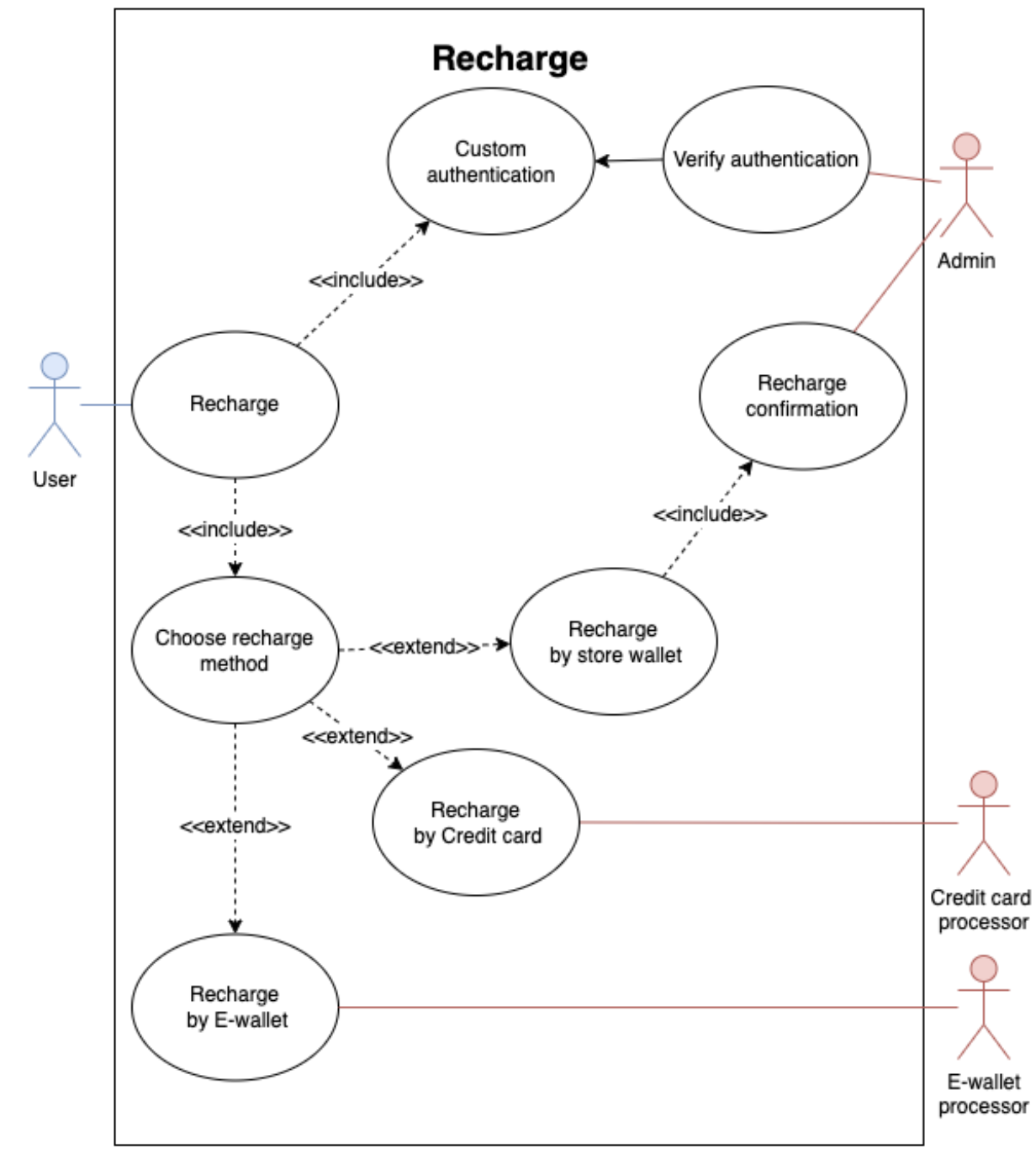
|  |  |
| --- | --- |
| **Use Case** | Order (Out of money) |
| **ID** | UC3.1 |
| **Description** | Users order products in the website without money |
| **Primary Actors** | Customers |
| **Secondary Actors** | None |
| **Preconditions** | None |
| **Main Flow** | 1. The alternative flows begin following step 4 of the primary flow. 2. The system determines whether the user has sufficient funds to place an order. 3. The system redisplays the cart page with an invalid message. 4. The user can recharge before placing an order |
| **Post Conditions** | None |
| **Alternative Flows** | None |



*Figure 6 Use case customize*

*Table 8 Customize use-case*

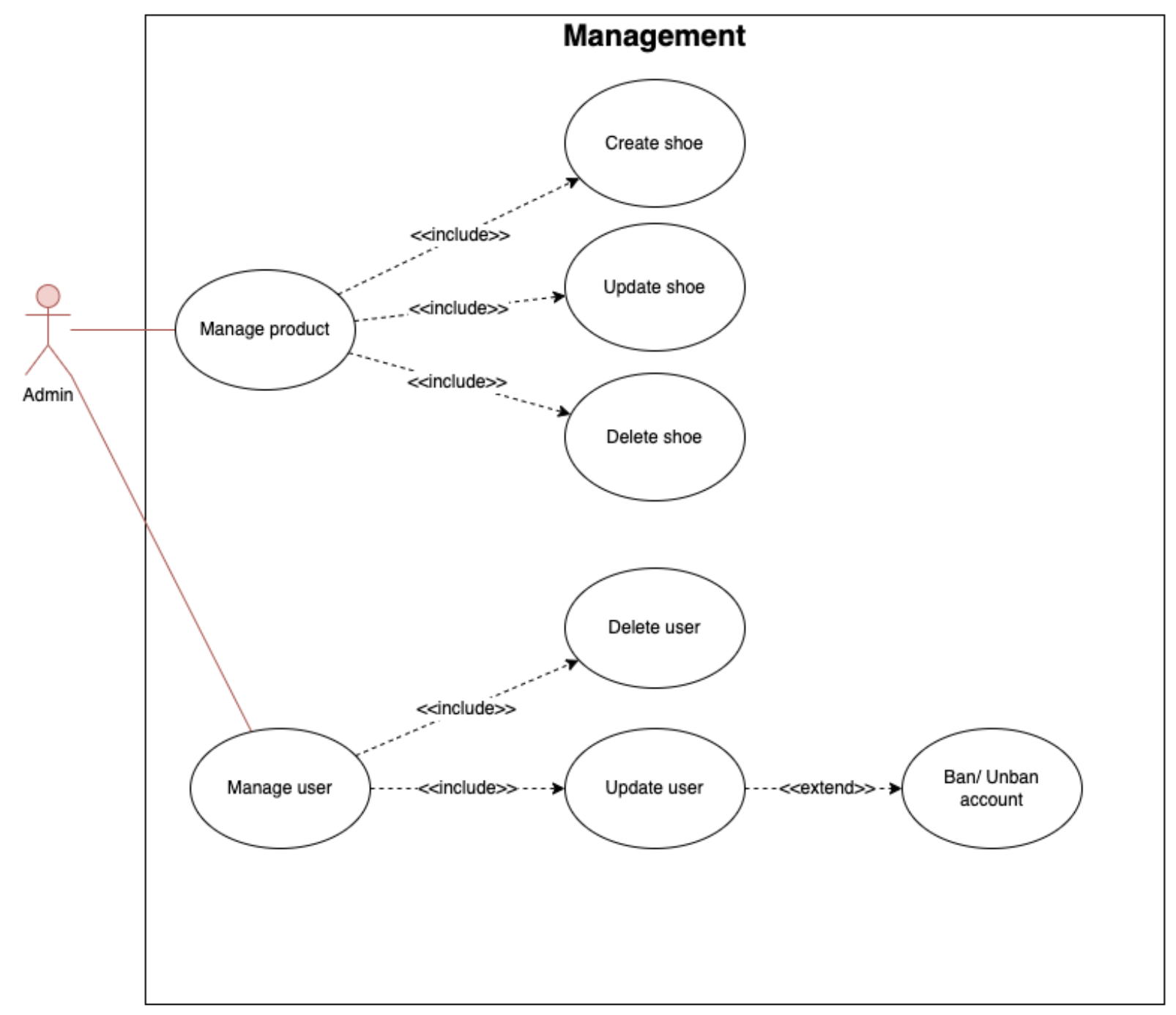
|  |  |
| --- | --- |
| **Use Case** | Customize shoe |
| **ID** | UC4 |
| **Description** | Users customize their shoes with color they want |
| **Primary Actors** | Customers |
| **Secondary Actors** | None |
| **Preconditions** | Users want to customize their own shoes |
| **Main Flow** | 1. When a user clicks the "Customize" tab in the navigation bar, the use case begins. 2. Users can choose their favorite shoes then custom 3. The users choose their favorite colors for each part of the shoes 4. The users choose button ‘Save’ so store it to the system 5. The system stores their new custom to database |
| **Post Conditions** | The user has access to the system, the shoes has been created |
| **Alternative Flows** | None |



*Figure 7 Use case recharge*

*Table 9 Recharge use-case*

|  |  |
| --- | --- |
| **Use Case** | Recharge |
| **ID** | UC5 |
| **Description** | Users customize their shoes with color they want |
| **Primary Actors** | Customers |
| **Secondary Actors** | None |
| **Preconditions** | Users want to customize their own shoes |
| **Main Flow** | 1. When a user clicks the "Customize" tab in the navigation bar, the use case begins. 2. Users can choose their favorite shoes then custom 3. The users choose their favorite colors for each part of the shoes 4. The users choose button ‘Save’ so store it to the system 5. The system stores their new custom to database |
| **Post Conditions** | The user has access to the system, the shoes has been created |
| **Alternative Flows** | None |

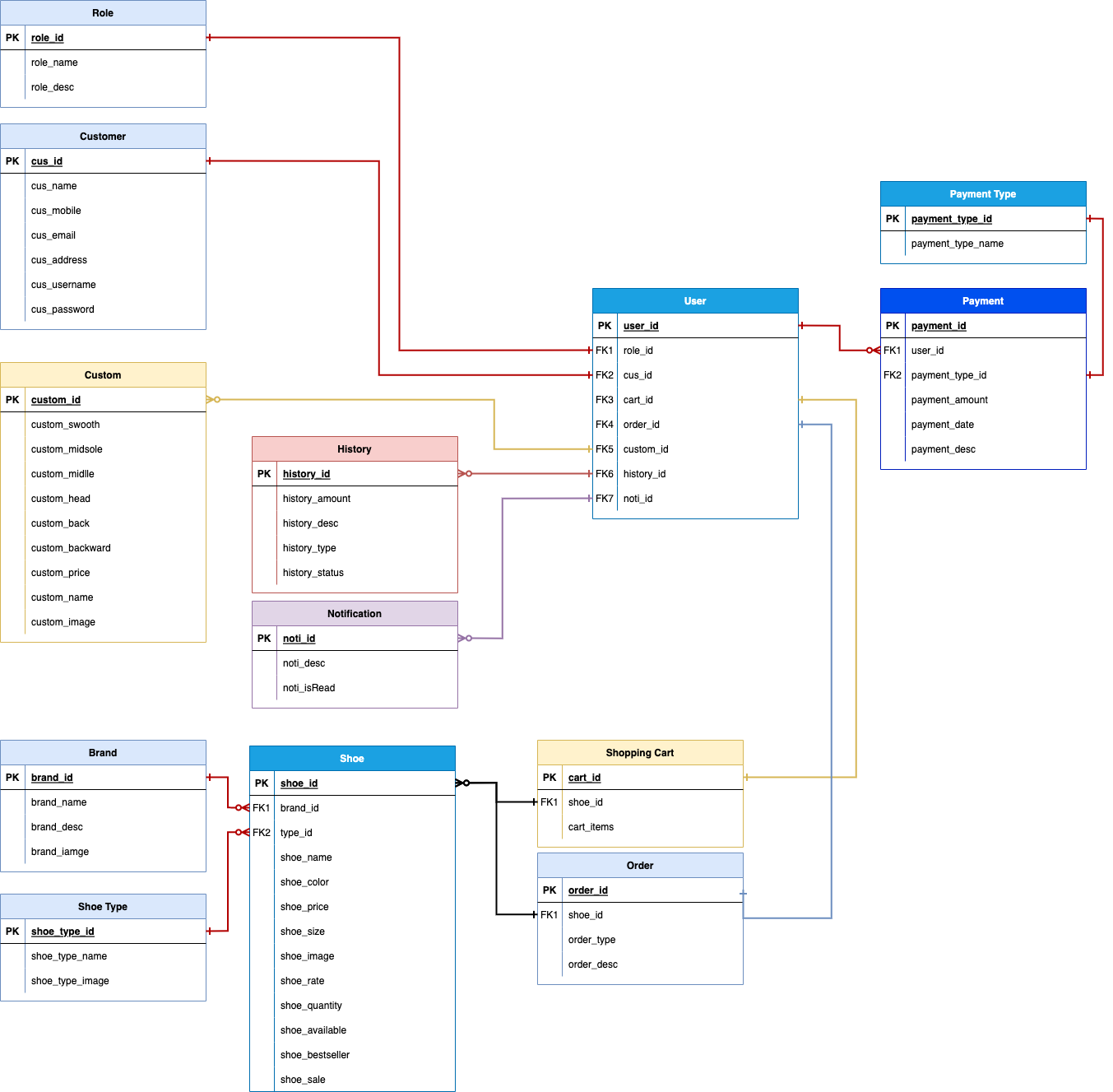


*Figure 8 Use case management*

*Table 10 Management use-case*

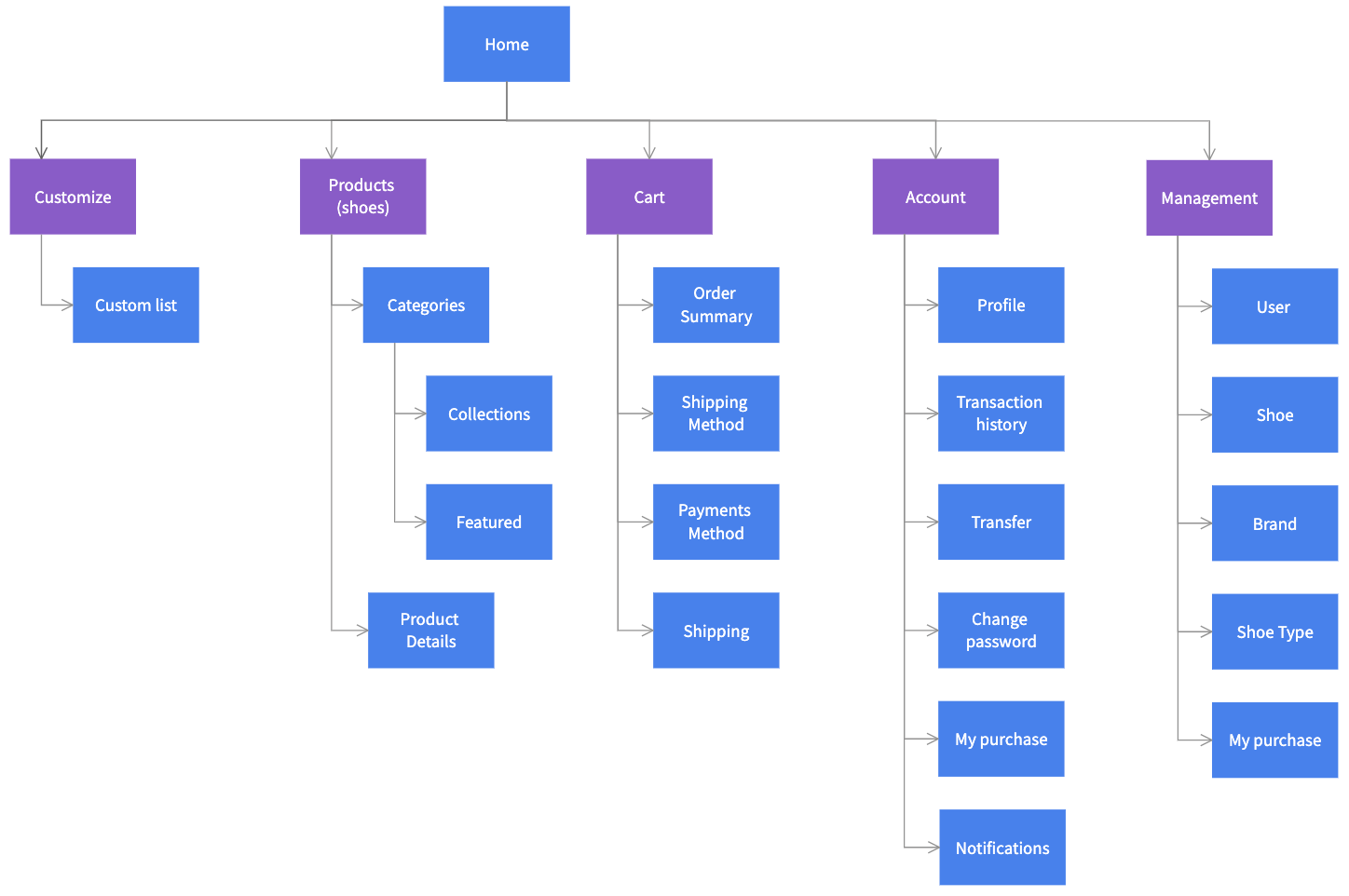
|  |  |
| --- | --- |
| **Use Case** | Management shoes |
| **ID** | UC6 |
| **Description** | Admin manages system to implement several main features in the system |
| **Primary Actors** | Admin |
| **Secondary Actors** | None |
| **Preconditions** | Admins want to create, update, delete shoes and users |
| **Main Flow** | 1. When the admin clicks the "Management" tab in the navigation bar, the use case begins. 2. Admin can choose model to manage its information 3. The system shows the table with the information of admin’s chosen 4. Admin choose the function needed to implement on the data 5. The system implements the chosen function then add, remove, or edit |
| **Post Conditions** | The data is changed than stored or updated in database |
| **Alternative Flows** | None |

## 4.4 ERD



*Figure 9 ERD*

## 4.5 Sitemap



*Figure 10 Sitemap*

# Chapter 5 Review of Software Development Methodologies

The software development life cycle (SDLC) is a framework that shows the gradual work flow of each step in the software development process, including requirement collecting, analysis, design, implementation, testing, and maintenance. By looking at cutting-edge approaches and methods to reduce the complexity of the development process, many businesses and software developers hope to enhance the field of software development. When selecting an acceptable software development technique, they are experiencing trouble. This is contributing to the software development process' failure. The Waterfall model, the Iterative model, the V-shaped model, the Agile technique, etc. are only a few examples of development models. These models have benefits and drawbacks. In this report, the key aspects of software development approaches are discussed, along with the benefits and drawbacks of each model.

## 5.1 Waterfall

One might present this approach as a conventional process for Software development. It is a sequential process of development, and the development's progress is estimated in accordance with its subsequent phases, each of which displays a downward increase like a waterfall. It places a strong emphasis on meticulous preparation and thorough documentation as results. Each and every initial stage of this process begins after the previous stage has been completed, and there is a linear sequential flow of actions. In order to get the ideal result at the end of each phase and up to that point, the Waterfall model is recursive. Each stage has unique deliverables that must be completed for that stage to be considered complete. This is a method or architecture for software development that makes it possible to anticipate the future using a software development plan. After the stages of development and testing are complete, the project owner may gather feedback. This methodology is appropriate for small-scale software development projects with uncomplicated needs and straightforward planning.

Requirements

Design

Implementation

Verifacation

Maintenanace

*Figure 11 Waterfall Methodology*

### Advantages of Waterfall methodology

One benefit of employing this system is its simplicity and ease of use and understanding. Convenient technique management aids in producing accurate and detailed outputs at each stage and enables process evaluation after completion. When used for small projects with appropriate and clear needs, this methodology is more effective. Before going on to the next phase, each phase must be finished within a predetermined time frame, showing a demonstrable advancement of the development project. Another advantage of this system is its easy implementation due to its linear flow. The very minimum of resources would be required for implementation. A thorough and appropriate documentation process helps to improve both the final product's quality and the development process.

### Disadvantages of Waterfall methodology

Before all phases of the life cycle are finished, no functional software could be obtained. It is important to keep in mind that there is more danger and unpredictability available. Large, intricate, and object-oriented projects should not be undertaken using this paradigm. If this methodology is used on projects that involve high-risk moderations, it may have an impact on project failure. After the first phase of development is over, the second begins.

## 5.2 Spiral

The Spiral Methodology should be used with a large development that may be introduced incrementally and iteratively. It should be built on the idea of a development project. A risk-driven process model generator that aids in guiding multi-stakeholder software systems is what the spiral development model might be characterized as. Spiral methodology focuses on identifying the goals and evaluating potential alternatives while following the correct, documented project procedures. The four main zones of the Spiral approach are as follows:

Planning

Analyzing risks

Evolution

Analysis

*Figure 12 Spiral methodology*

As long as the final product or generated software program is not suitable for the market environment, the project will again go through the same zone in the manner stated above. Alternate solutions are proposed and put into action if any risks are found.

### Advantages of Spiral methodology

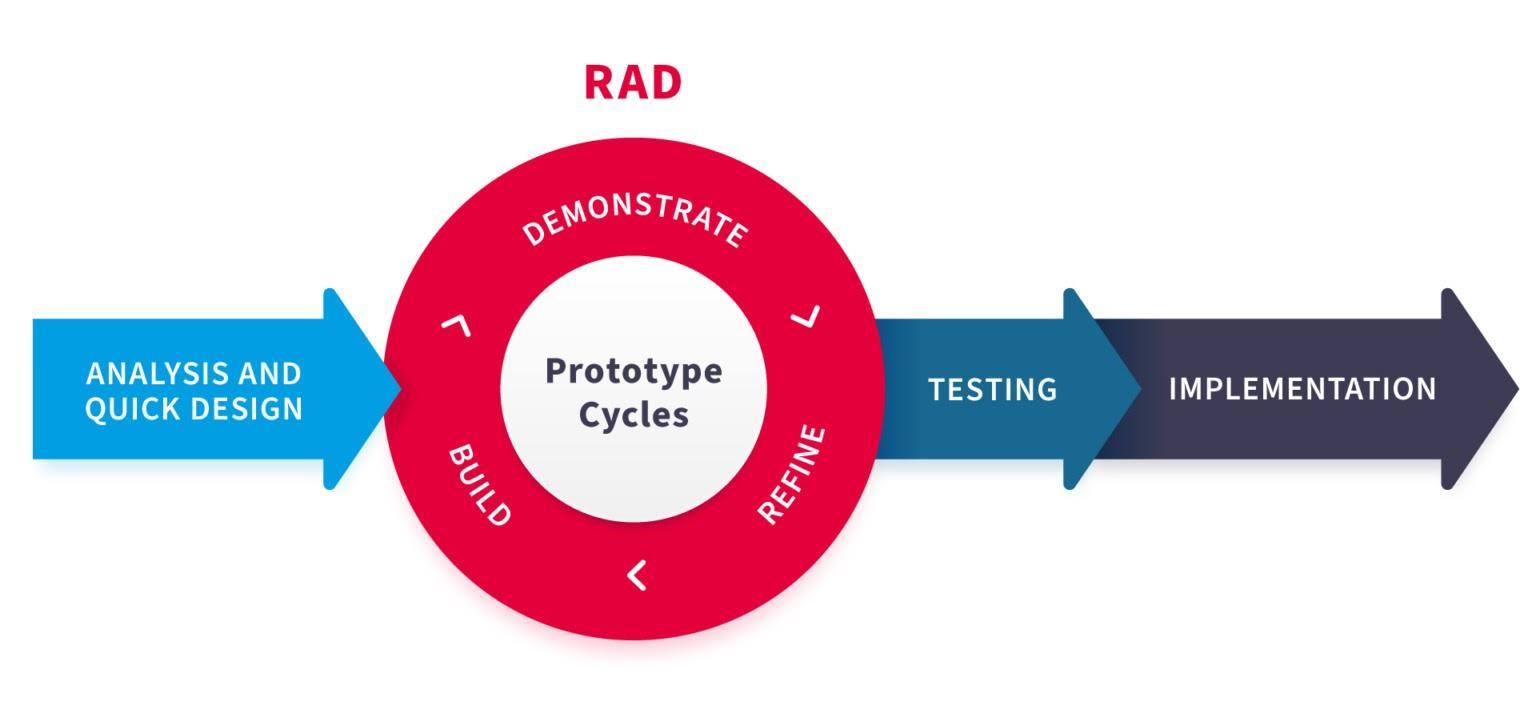
It is preferable to apply for large-scale projects with dangerous missions. The requirements could change in response to feedback from clients or stakeholders. control consistency with the aid of documentation. The development process allows for the addition of additional features and capabilities. more adaptability and the ability to use different development methodologies. ease of adaptation that aids in meeting project requirements.

### Disadvantages of Spiral methodology

High cost to employ the methodology In-depth specialists are required to evaluate the project's hazards. Success of the project as determined by the risk analysis session. Not appropriate for applications for small-scale projects. To handle the project's need modifications and who is properly knowledgeable about modifying deadlines and adding milestones, you need an experienced project manager.

## 5.3 RAD

The goal of **rapid application development (RAD)** is to produce an end product quickly and of higher quality than is possible using conventional techniques. It was created to fully profit from development projects. Compared to planning, which is the most crucial component of the development project, this method is more specialized and significant to the development. Structured approaches and prototyping are particularly employed in rapid application development to establish user requirements and create the ultimate solution. The process of development starts with the structural techniques. Preliminary and business data models, which provide assistance, are intertwined with this process. The next stage involves prototyping to validate the requirements that have been gathered, ultimately clearing out the data and the process models. These stages are repeated repeatedly until the project satisfies the technical and business needs of the project owner, which are the fundamental two aspects completely affecting the outcome. The project owner can offer input when each phase is finished. This methodology can be used for any large, medium, or small-scale project.



*Figure 13 RAD methodology*

### Advantages of RAD methodology

Enhanced capability to reuse the components. Occurrence of fast fundamental reviews which is a remarkable benefit. Proper and easy changes of requirements by collecting customer and project owner feedback. Better support and good capacity for solving integration problems from the start.

### Disadvantages of RAD methodology

One fundamental drawback is inadequate documentation. High costs for new projects and potential code issues during integrations. For the development phase, more experienced designers and developers are required. depending on modeling abilities that are most useful for project and development process dependability. Due to the significant cost of modeling and the automated code production, low budget projects are not appropriate.

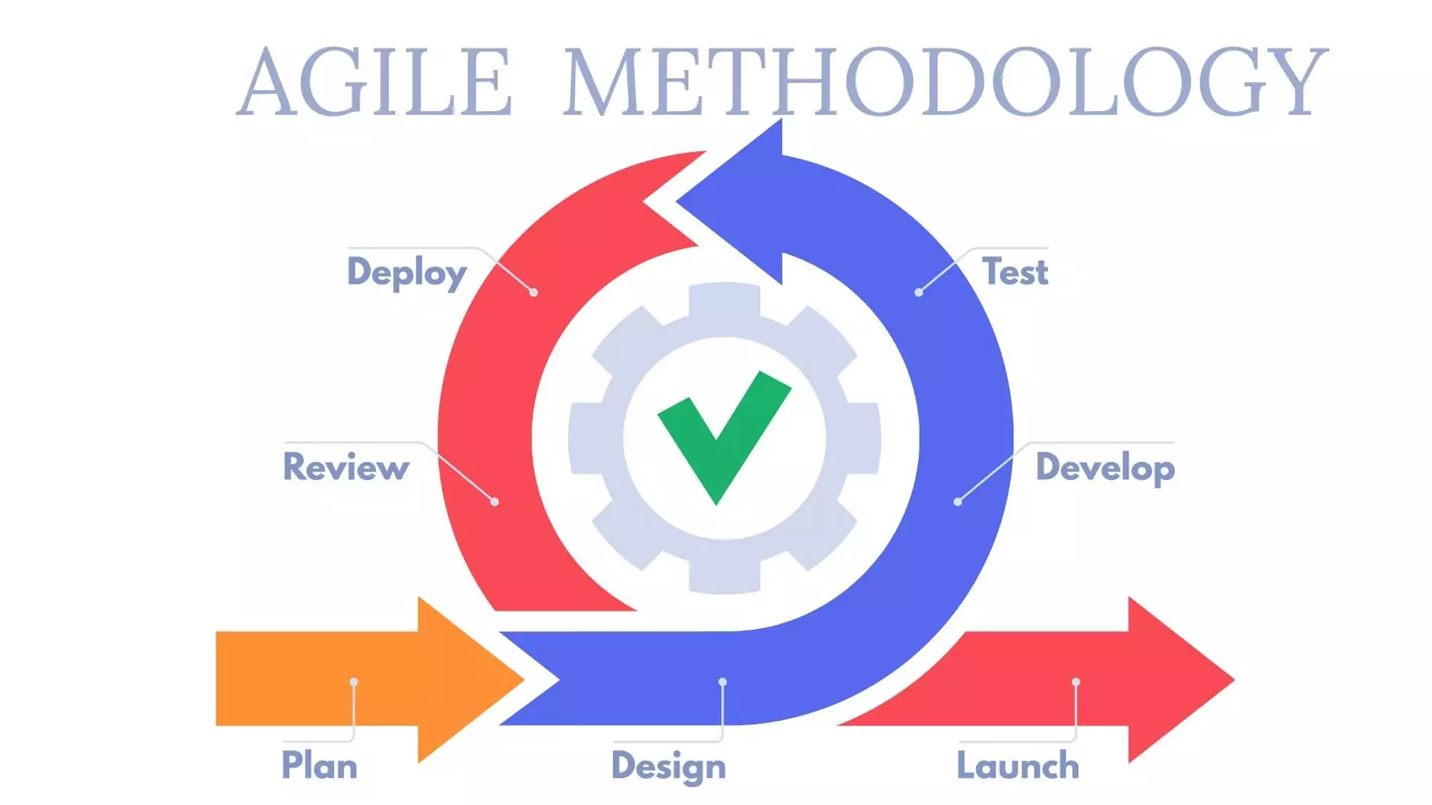
## 5.4 Agile

A software development approach can be characterized as the appropriate direction to continue ongoing progress. Additionally, it provides structure for all development operations and is a collection of organized tasks involved with software development process functions. By providing appropriate rules and guidelines, a software development methodology facilitates the planning, analysis, design, development, implementation, testing, and maintenance of the software development process. These methods aid in project management for software development (Rathnayaka and Kumara, 2022).

There isn't a single, universal approach that works in every circumstance. The reason for this is because software projects differ from one another in terms of their traits, the resources they require, and the development tools they employ. When choosing a software development approach, it's vital to take the project's characteristics and nature into account. The choice of the proper software development approach determines the outcome and success of the project. Additionally, it facilitates meeting deadlines, controlling costs, improving client satisfaction, decreasing expenses, etc. It's crucial to select the best and most appropriate software development technique for the project after taking these factors into account. This article discusses some of the fundamental agile software development approaches.

### Agile process

Simply said, agility refers to removing the burden typically associated with traditional software development processes in order to speed up the ability to respond quickly to changing surroundings, adjustments in user requirements, and work ahead of project deadlines. Agile techniques place greater emphasis on software development than documentation. This methodology tends to produce more work during its iterations and leverage consumer feedback to improve the program. This development methodology is far more effective at enhancing the ability to adjust user requirements until the user or customer is satisfied with the finished product, as well as quick and efficient development throughout the entire process. The majority of project teams with less than ten members can use this strategy. The team members must frequently engage in face-to-face interviews or conversations, which is the reason for this. When employing these approaches, picking a team spokesperson is preferable. Agile methodologies have the drawback of not working well in large teams.



*Figure 14 Agile methodology*

The various agile methodologies are listed below (Rathnayaka and Kumara, 2022):

* Extreme Programming
* SCRUM
* Crystal family methodologies
* Feature-Driven Development
* Dynamic Systems Development Method
* Adaptive Software Development
* Open-Source Software development
* Agile Modeling
* Pragmatic Programming

### Advantages of Agile methodology

Rapid, continuous improvement can lead to highly delighted clients. Contrary to sustaining connections with processes and tools, the development team's open lines of communication and consistent contacts with consumers have a positive impact on future outcomes. One benefit of these approaches is the quick delivery of software in a short amount of time. Face-to-face communication between the customer and a project team member makes it simpler to understand the needs and comments of the consumer. A better development process is aided by ongoing, current cooperation and engagement of project operations. continuous attention to technological development and good design, both of which are important for producing quality software. It is much simpler to systematically adjust to changing circumstances, demands, and project aspects. Permit changes to the project's needs even beyond the deadline and while the development process is still underway. The methodology's daily work logs assist in keeping current day activities throughout the project development process.

### Disadvantages of Agile methodology

The first is that because there won't be any system documentation, agile approaches are unsuitable for new-build engineering and unsuitable for maintenance. It might be challenging to initially estimate the amount of work required for a large-scale project's whole software development process. During this process, there was more documenting and designing. If the customer does not hold clear requirements, the project development process may veer off course and result in worse results than the customer anticipated. Agile methodology places a greater emphasis on user involvement, and ineffective user cooperation and communication can lead to project failure. It is not the correct methodology to create reusable software because it concentrates on offering fixes for specific problems rather than general ones.

## 5.5 Selected software development methodologies with justification

Our team wasn't sure which software development approach to adopt at the start of the project to meet the requirements and make the project execution easier. We have chosen 2 techniques that may be used effectively in this project after learning about software development methodologies like Agila, Spiral, RAD, etc. because there are others we must adopt when implementing the project.

To start with, this is a little project with open-source code. Compared to other projects, there is less work involved in this one. The team members can support one another while working alone or collaboratively to build the functionality included in the project at the same time as the tasks outlined in the criteria. judgment.

Second, there are just 2 team members on this small, unique project are Dong (the front-end developer) and I (the back-end developer), both of whom are capable of being divided into separate tasks.

But after carefully examining the benefits and drawbacks of the two approaches, Spiral and Agila, we made the decision to conduct further research to identify the methodology that would be most effective for our Dustin e-commerce project.

| **Agile Model** | **Spiral Model** |
| --- | --- |
| The objective of the Agile approach is to enhance agility by eliminating time- and labor-intensive processes. | Principal principle of the Spiral approach is risk management. |
| The Agile model's focus on providing an increment to the customer after each Time-box increases client contact. | The spiral strategy focuses largely on a number of unanticipated risks, but there is minimal customer participation. |
| Large projects that are easily decomposable into manageable chunks that may be generated progressively during each iteration are suitable for the agile methodology. | The Spiral model is suitable for projects that are susceptible to several hazards that are difficult to anticipate at the beginning. |
| In the agile methodology, documentation is not required. | The Spiral model requires appropriate documentation. |

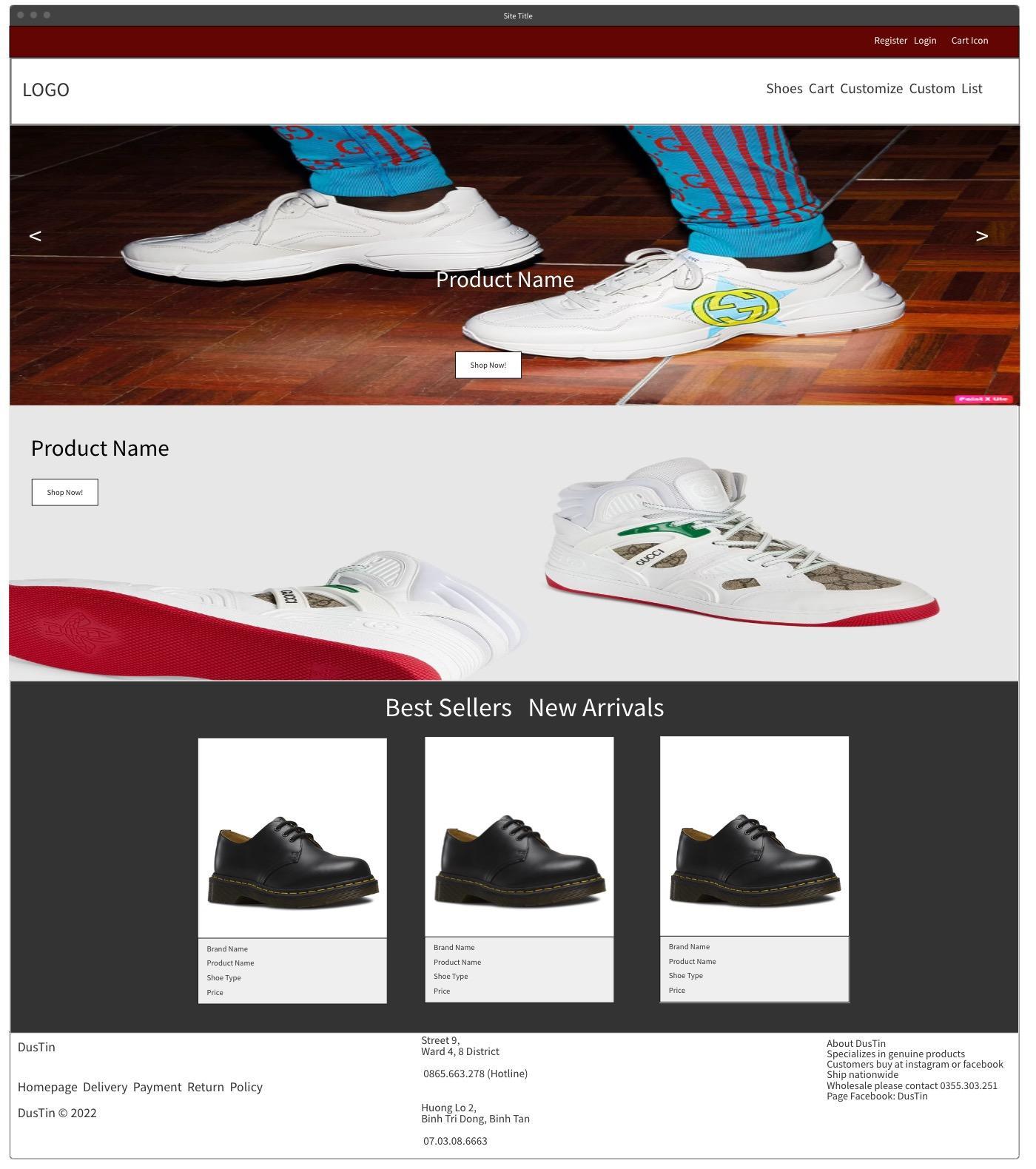
This project is modest, thus there is a lot of risk in the planning and implementation phase. Yes, these are the hazards we wanted to prevent but were unable to do so:

* **Not timing the tasks correctly**, leading to the fact that when team members perform sometimes too much time or finish too quickly.
* **Difficult 3rd party systems** such as Momo or Heroku. With Momo: When creating the transaction function with e-wallet payment, the majority of people will target Paypal, Apple Pay, etc. - Difficult third-party systems like Momo or Heroku. However, Momo is the most popular choice in Vietnam because it is practical and was created with Vietnamese users in mind, making it simpler to use and register. Momo is targeted for document design and use for Momo-supported or available stores, which makes it challenging to set up Momo in the project. The payment function was supposed to be designed by me as a back-end developer in 7 days, however when I worked with Momo, it took me more than 1 week. About Heroku, most people are aware of Heroku as a platform for supporting website deployment that is free to use, but Heroku will soon stop offering this service to programmers working on smaller projects like us. We will either have to pay for Heroku or find another platform to deploy our website on.

As a result, we made the decision to adopt the Spiral model because its core idea is to support the team in managing risk throughout execution. Agile focuses mostly on finding ways to eliminate pointless tasks so that the project can be finished more quickly. However, the application of Spiral will be very helpful as we execute in order to resolve unforeseen issues prior to the project's deployment. Agile emphasizes sending increments to the client after every Time Box, making customer interaction more regular. In our project, however, this is not necessary. In particular, with Spiral, we can manage risks while executing rather than having to plan for them beforehand.

# Chapter 6 Design and Implementation of your demo product

## 6.1 Product Analysis and Design



*Figure 15 Home page GUI*

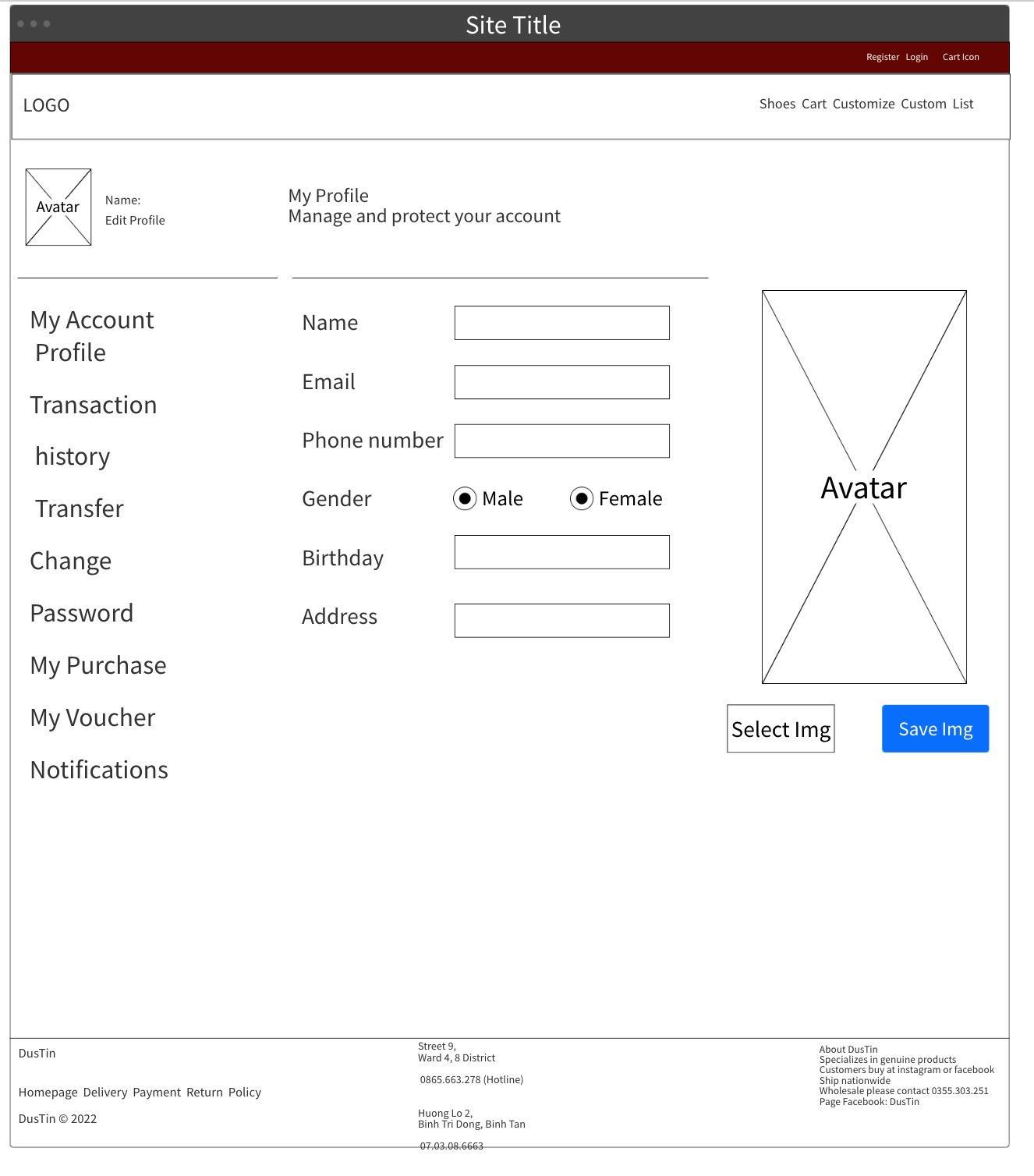
As per our team's plan, we decided to design a sub nav instead of designing a nav with no sub nav, so we design the sub nav so it's easy for customers to take actions when you register and log in. In addition, this makes it possible for customers to use the website more easily to use and act more comfortably. The reason we decided to leave the logo on the left and the items on the right is because it is more convenient for mobile users and small devices, because for small devices, we use the sidebar for manipulation. Collapsible with navbar. And users can more easily operate with one hand when the sidebar is on the right-hand side. Not only that, the navbar is also designed with a fixed position for scrolling down with the client view, making it easier for people to manipulate instead of scrolling up.

Next is the carousel, we choose the carousel to display featured images to promote to customers about new products and easily catch the information instead of having to enter specific shoes. Along with the button in the middle of the image can help users redirect to the detail page instead of having to search for the shoe name. And the next part, we put pictures of new and most typical products to stimulate users to show users the best products of our shop so that users can easily choose the best products. suitable for individual use needs and, in this image, we also leave the product name and a button so that users can easily manipulate more conveniently.

In the Horizontal slide bar section, we also leave two buttons: best seller and new arrivals to display the products corresponding to the two buttons above and to display more clearly and in a more beautiful way we have limited to only three products and in Each product, user can click to see details of the above products to create a pleasant comfort when using the website in the most comfortable way for users.

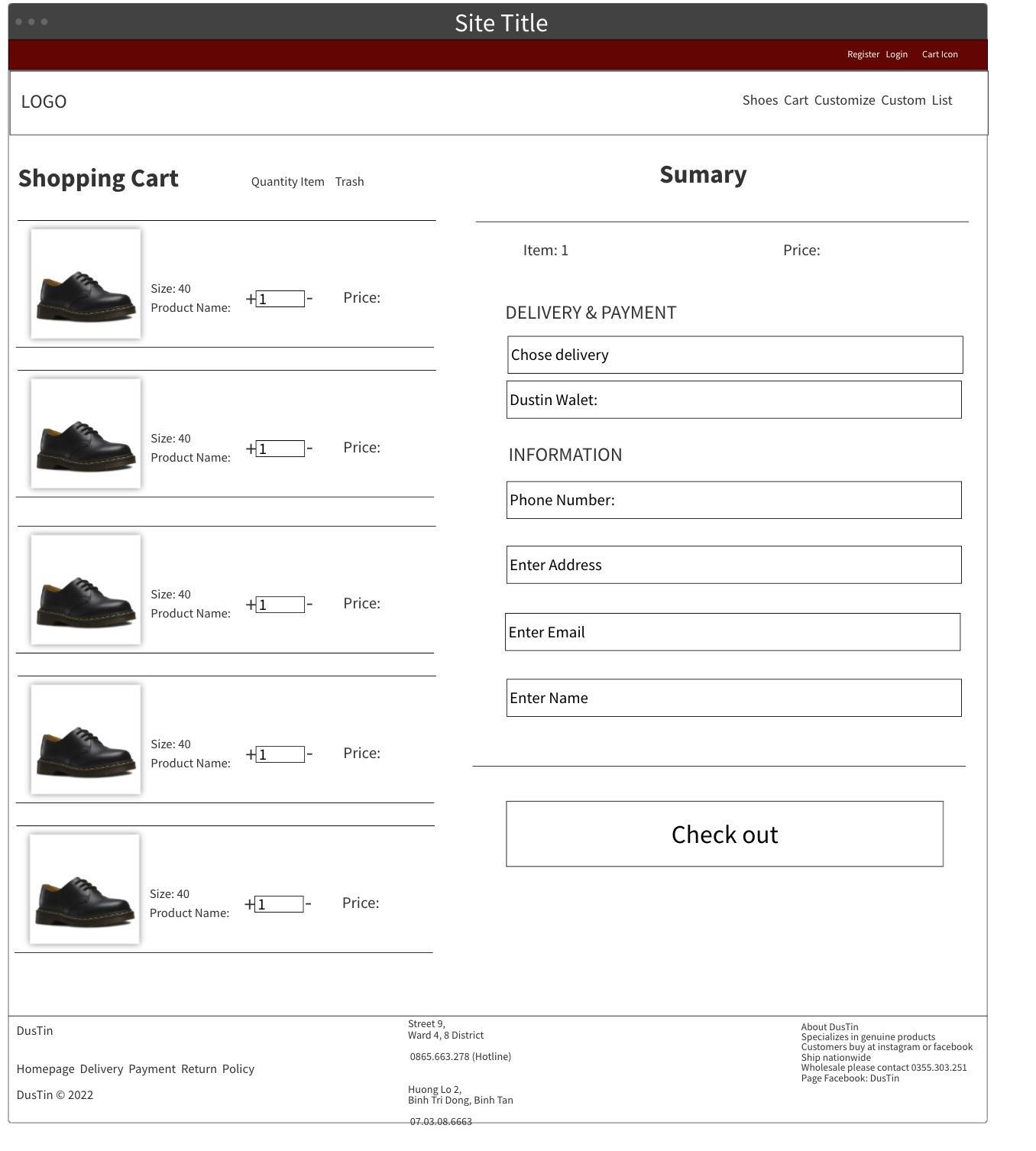
And finally, the footer section, we divide it into three columns to display the necessary information for the user

* The first column we when the information of the site
* The second column we will show the user information such as his address and phone number for the user to contact
* The third column we will put general information about our shop



*Figure 16 Profile GUI*

Like the homepage, the page above is the user's page (profile) with the same header and footer. By design, we divide this page into 2 columns, the first column contains buttons to help navigate the user, column 2 is the information of the button from column 1. For example, My Account>Profile will display input boxes containing user information. used to change. The top of column 1 contains the user's personal avatar. And column 2 is again divided into 2 columns for users to manipulate, otherwise the input box will be too long to enter short information that is not necessary. So, dividing by 2 makes the website look clearer with more information to edit. Such as username, address, etc. in column 1 and change the avatar in column 2 as designed.



*Figure 17 Shopping Cart GUI*

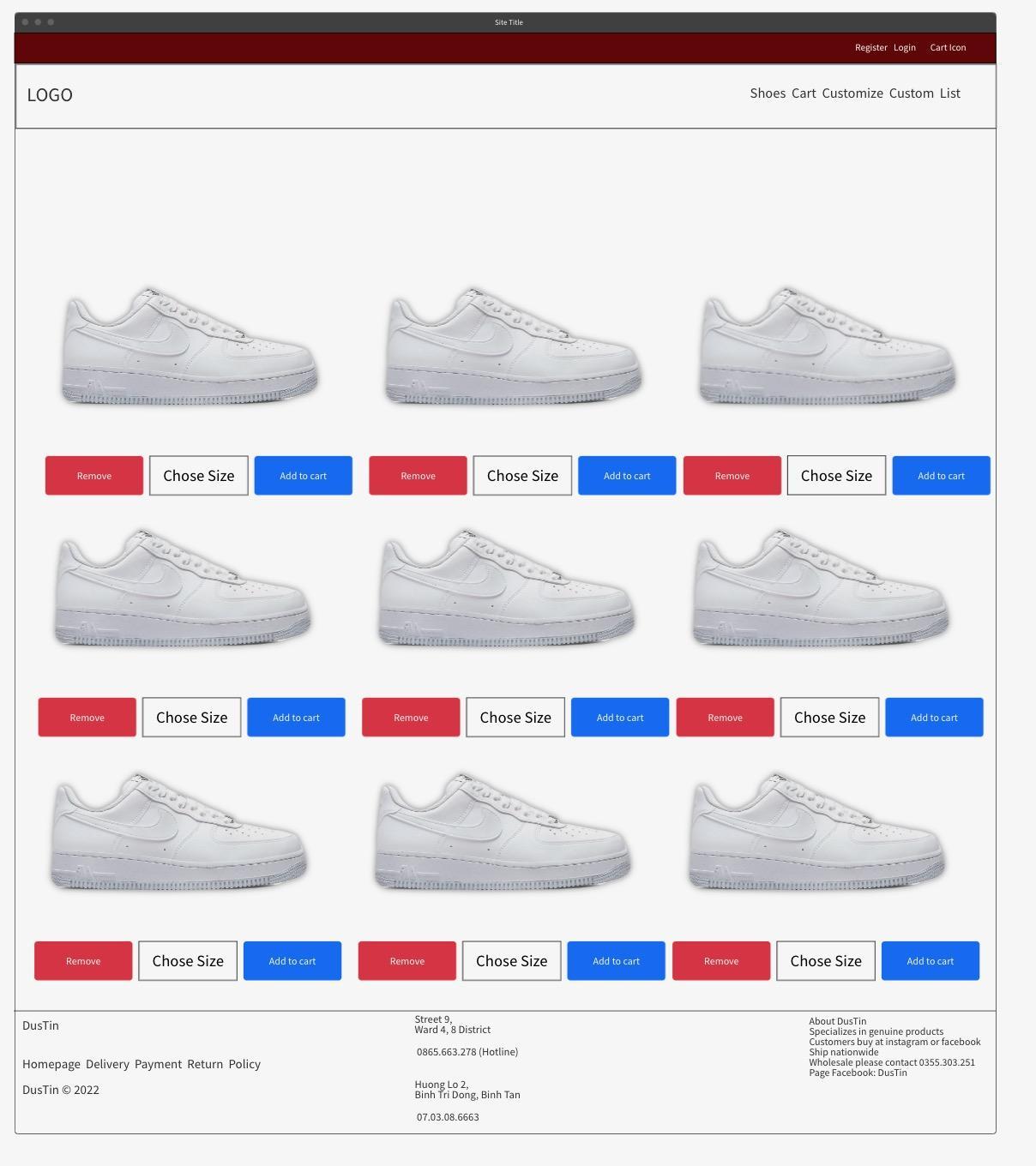
Next is the Dustin store shopping cart page. To display shopping cart information and as well as personal information for checkout, we decided to split this page into 2 so that all information is displayed in the user's view, avoiding scrolling down to checkout. However, with smaller devices, users still need to scroll down to see the full information. On the left column, we display the images and information of each product in the cart sorted by the time it was added. A main image of the row, followed by the product name, size, and quantity of the available product. There are icons representing each function on the cart such as cancel, decrease, increase and remove from the cart.

On the right side is the information for checkout, we build a form data to submit the information. There will be 2 select options for users to choose more convenient shipping methods and payment methods. Regarding payment methods as mentioned above, we have 3 main payment methods: Momo, Paypal, Credit card and Dustin wallet. Below is the personal information and the address to receive the goods after checking out. To increase the UX for the user, we render the personal information of the account into the input boxes so that the user does not need to enter it, but if the user wants to buy for a relative or friend, they can remove the information and edit it. according to your will. Finally, the button to checkout, if the cart is empty the button will be disabled.



*Figure 18 Customize GUI*

The next page is the special feature of our website - Customize. This page is created for customers to customize the color of the shoes they want, with a simple interface design with white as the main color along with header and footer like other pages. This feature can be used on any device to make it more comfortable for users to use. Currently with limited time, we only design the feature in 2D with 6 main parts of the shoe, the color is customized in RGBA's 16 million color palette. Some of the added features are Undo, Save and Cancle. We originally planned to position the buttons according to the UI design, but according to the user UX, the save button is usually on the right and cancel on the left, so the actual design has changed. The shoe parts to change on the shoes are a bit hard to see for the user, so we plan to use hover so that when the user touches it, it knows which part of the shoe it is.



*Figure 19 Custom shoe list GUI*

The finished custom shoes can be saved to their personal list for review or order. So, the custom list page is designed so that users can review their own designs. So, we designed them as a list, with 2 buttons so the user can remove or add to the cart and an input with a data list so the user can choose the size before ordering. Depending on the device, it will display more or less columns for the user to add or remove from the list.



*Figure 20 Product list GUI*

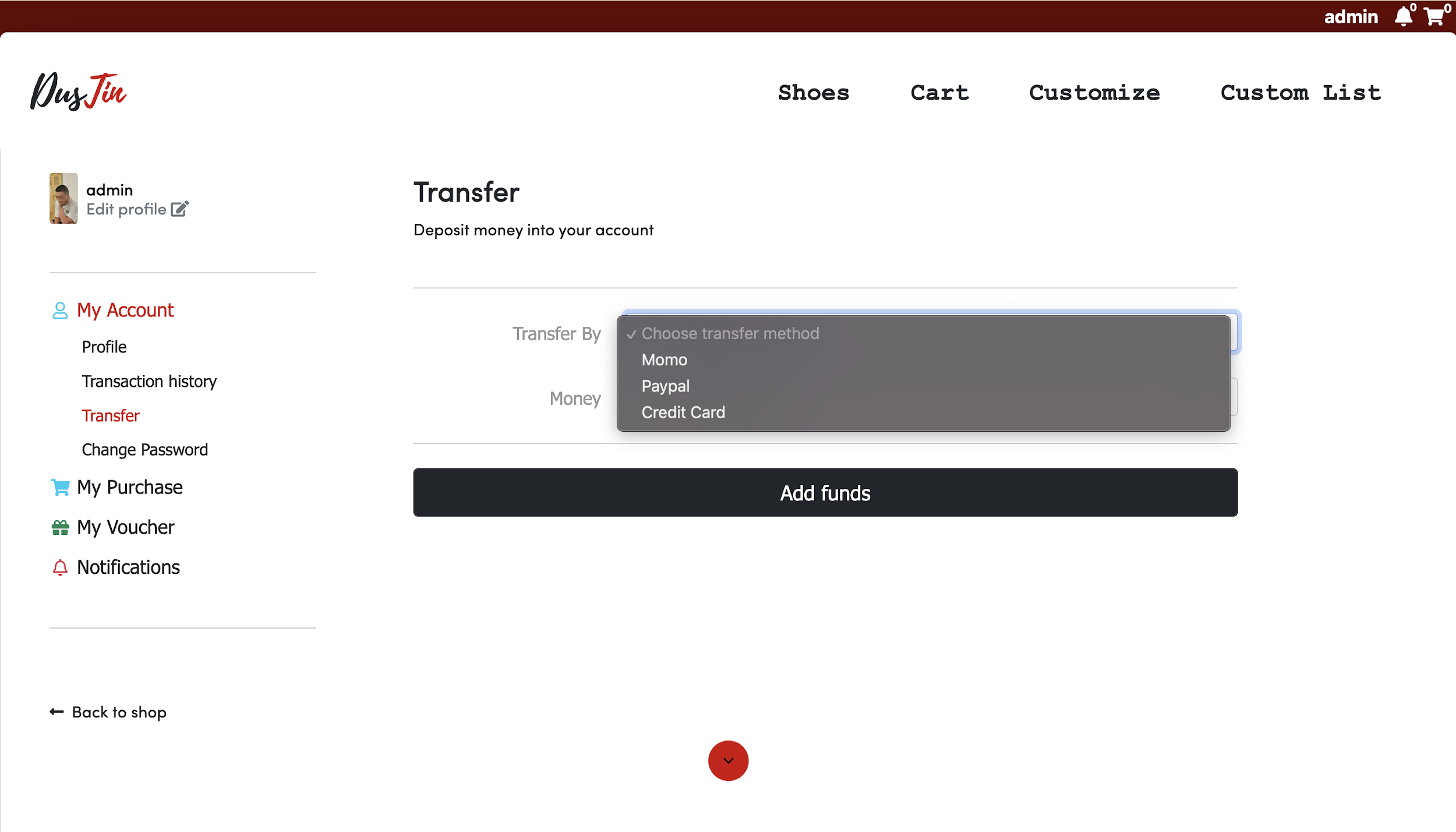
Finally, the product list page of the system, with a simple design for users to easily select products in the system, by dividing the column into 2 so that the left side is the shoe category and the filters for users' convenience. in finding. The right side will show the shoes available in the system, the number of columns will be based on the length of the system screen. For small devices like mobile and tablet, the search and filter list on the left will be displayed as a topbar and ul li so that users can click to open the search filters.

## 6.2 Features include with screenshots



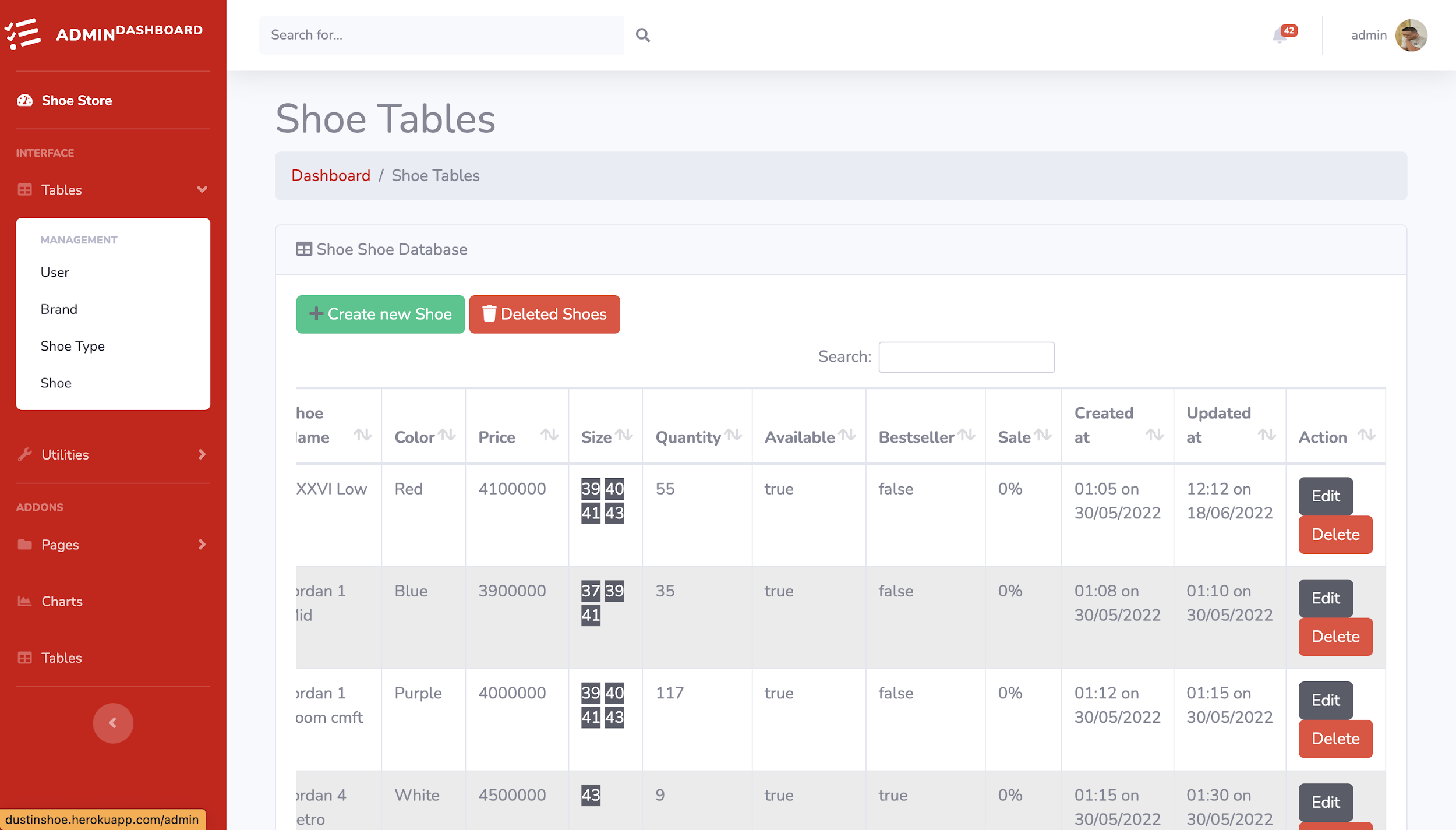
*Figure 21 Customize interface*

This screen we created for the Customize function, creates a custom shoe by changing the color of the 6 sections below the screen, then the user can undo his actions or save the customizations. complete. All changes are saved as an object in the database which can then be rendered in the user's custom listing page, from which they can Delete or add to cart after choosing the size of the shoes. that custom.



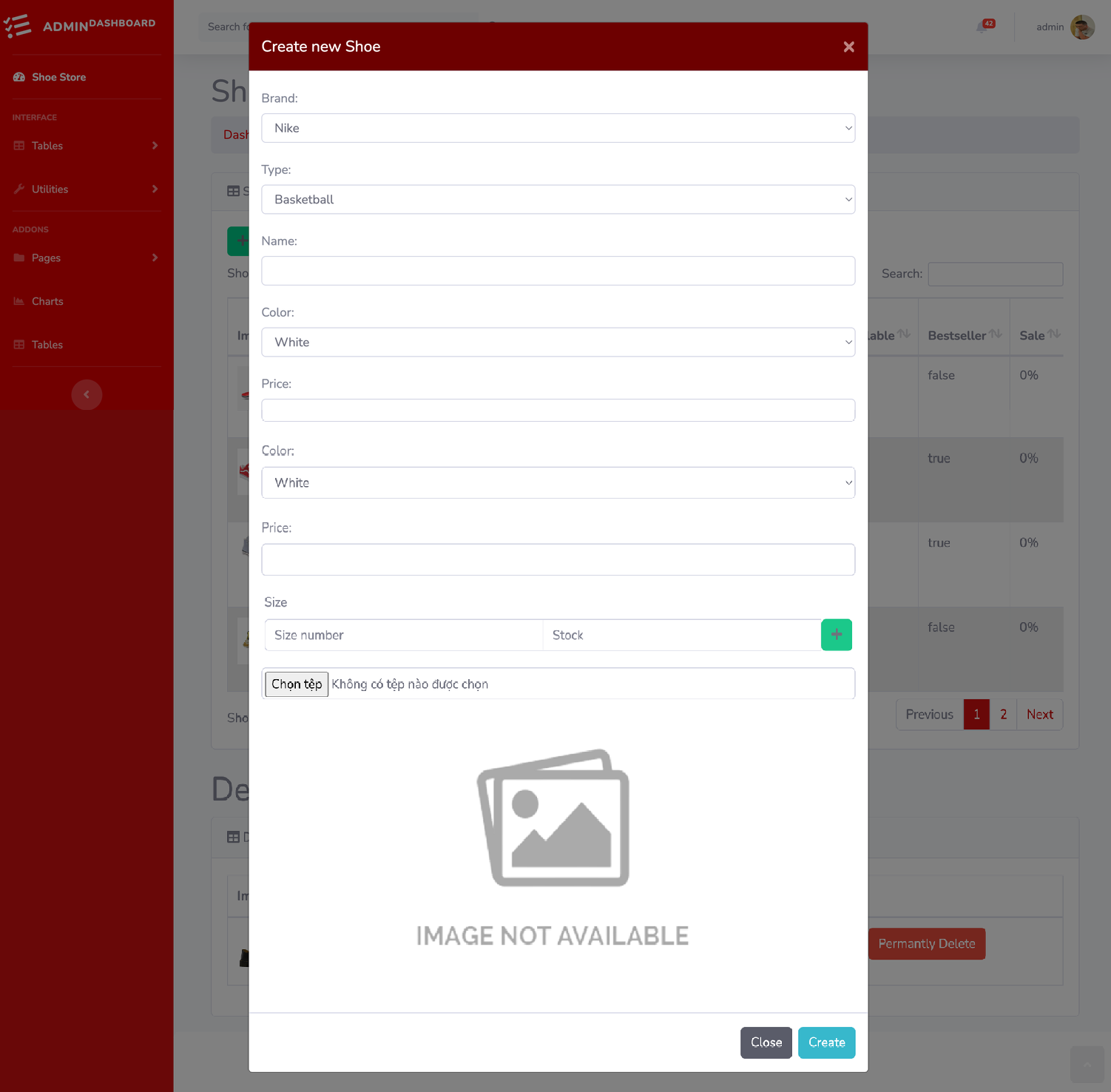
*Figure 22 Transfer UI*

The image above is the page to top up the user's account. As shown in the picture, there are 2 main ways to deposit money: E-wallet (Momo, Paypal) and Banking (Credit card). All deposits and deposit information are encrypted and to protect user information, and when depositing with Credit card, the website will open 3 more input boxes to enter card information to top up. If Paypal or Momo will redirect to the payment website of each E-wallet. After the payment there will be a notification if successful or failed, and the notification will be saved to the user side or history if successful.



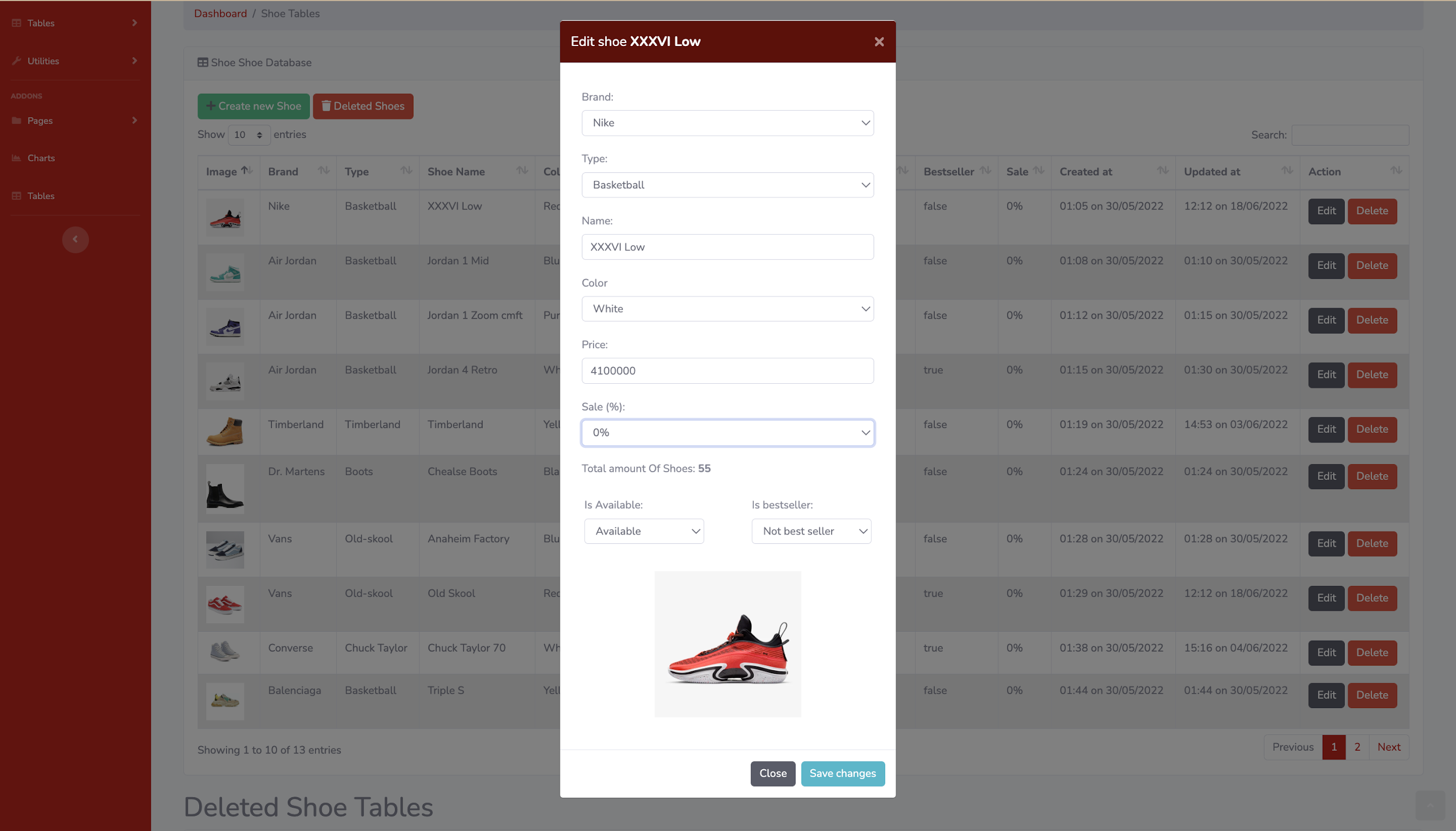
*Figure 23 Management UI*

The page above the picture is the admin's management dashboard page, here only the admin can access it and if users of other roles access it, it will be pushed to the Error page. This page is used to manage lists in the database such as Shoe, Brand, Type and User. Above will display the main functions such as Create, Edit, Update and Delete.



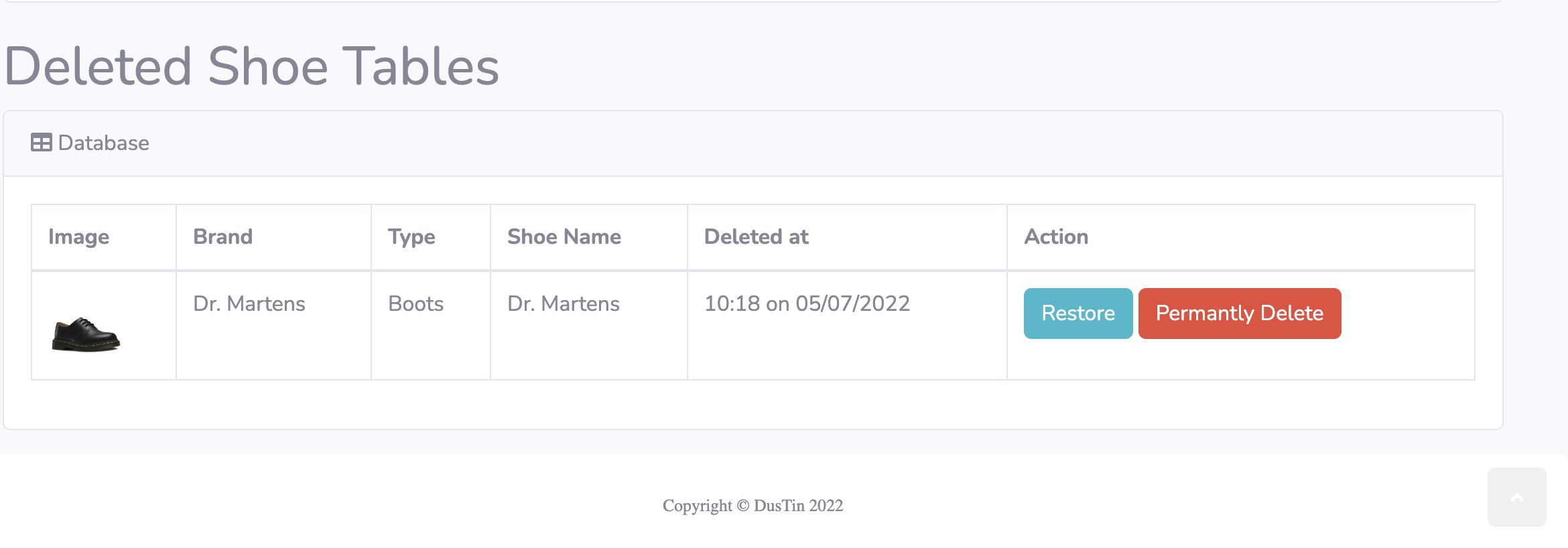
*Figure 24 Create shoe UI*

Next is the create function in the admin dashboard, depending on the data, it will display Create differently. Create is created from the modal via the Create button at the top left of the table. Modal appears for the user to enter new product or brand information, etc. Images can be uploaded from the user's machine side and stored in the upload folder of the code. On successful creation of new data will be added on the database side and rendered to the table.



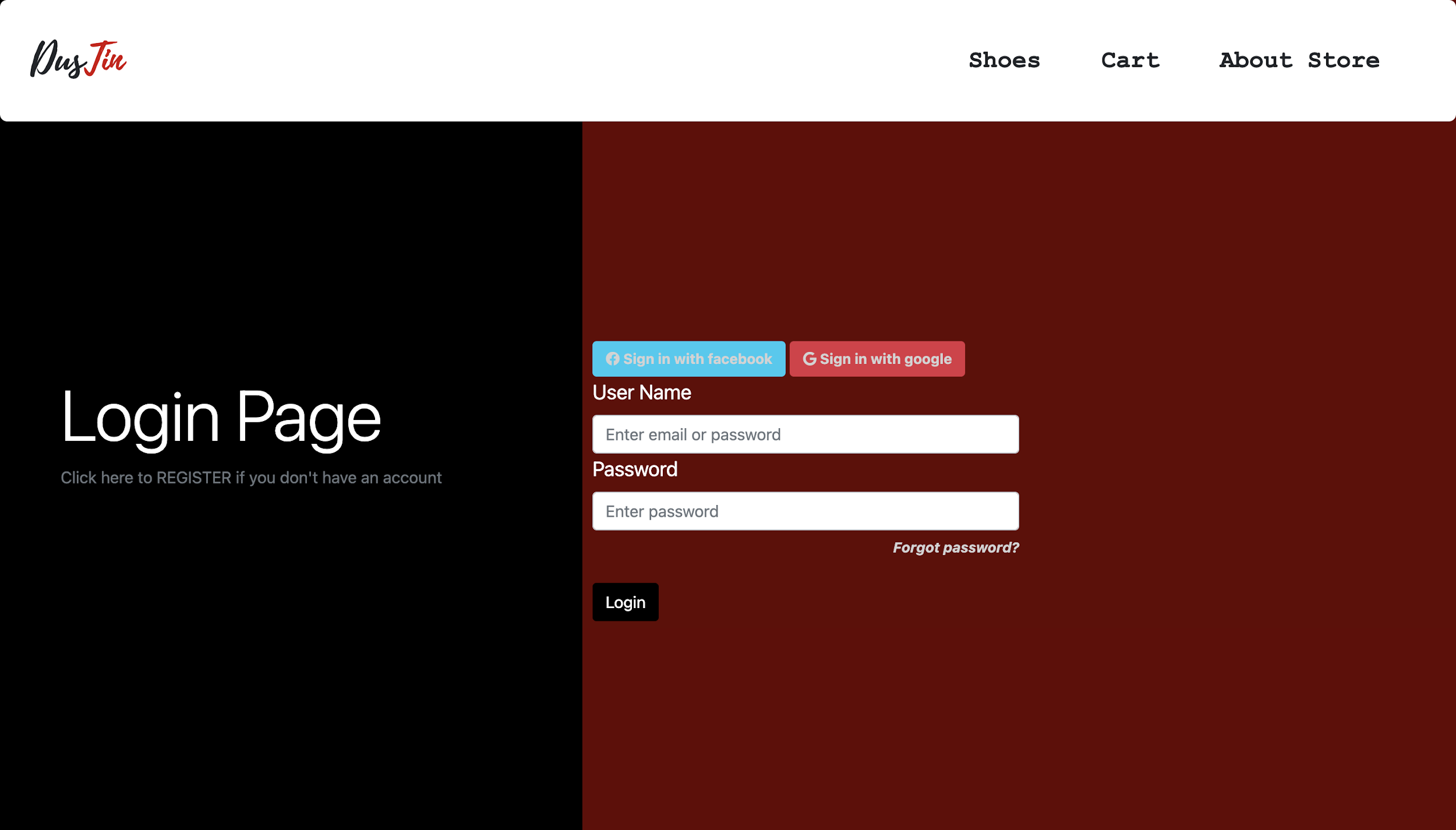
*Figure 25 Update shoe information UI*

Next is the edit function in the admin dashboard, depending on the data, it will display different Edit. The edit function is created from the modal via the Edit button to the right of each data row of the table. Modal appears for users to change the information of new products or brands, etc. When the new change is successful, the data will be changed on the database side and rendered to the table.



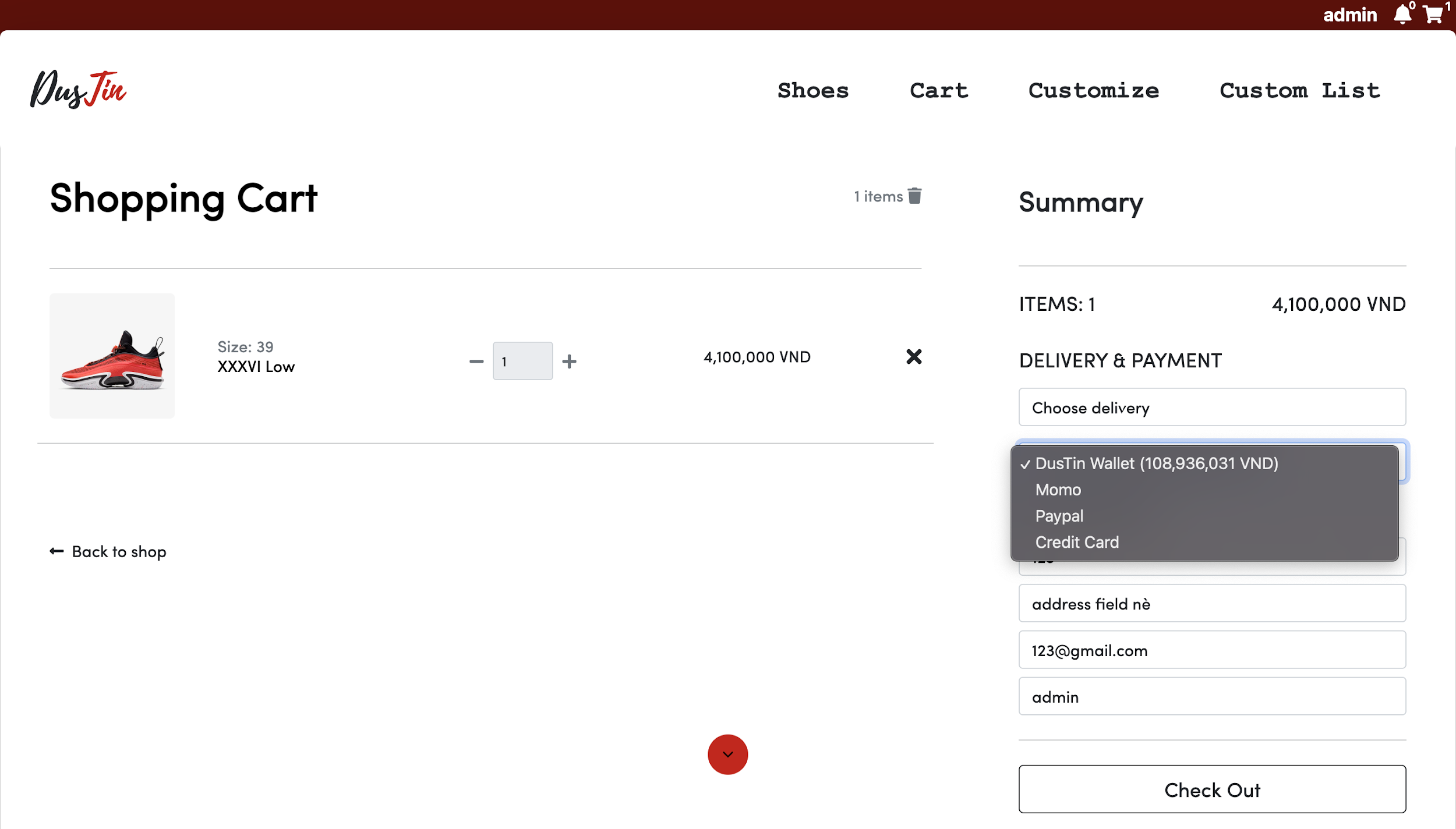
*Figure 26 Deleted shoe table*

This is the delete function like many e-commerce websites do, but I decided to use Soft Delete to add the delete function of this system. Because if force delete will cause the data to be deleted from the database and cannot be restored, soft delete will help the data to be deleted temporarily and can restore the website. And the 2 buttons are Restore and permanently delete



*Figure 27 Login UI*

This is Dustin's login page with 3 ways to log in: Dustin, Google and Facebook accounts. When logging in with the above social media, the system will push through a 3rd party to log in to the system. And when you log in incorrectly 3 times, the system will lock the account for 1 minute, 6 times will be permanently locked and can only be unbanned by the Admin. An important function is resetting the password if the user forgets his password.



*Figure 28 Shopping cart UI*

Shopping cart is an indispensable page in an e-commerce website with information about shoes when the user adds to the cart. Besides, there will be fields to display the price, quantity of each product as well as the total of the cart. Besides, there is a form for users to checkout if they have completed their shopping cart. The form will have the user's information available after logging into the system such as: delivery method, payment method, recipient's name, etc. And the checkout button will be locked if the cart is empty. Otherwise, it will be redirected to a 3rd party to pay if the user uses the e-wallet, if using the dustin wallet, the payment will be made immediately and if it is a credit card, the user needs to input the card's information. After payment, there will be a success or failure message.

## 6.3 Product Implementation

store(*req*,*res*,*next*) {

User.findOne({

$or: [

{email: *req*.body.email},

{phone: *req*.body.phone}

]

}).then(*data* => {

if(*data* != null){

return *res*.render('register', {

title: 'Register',

layout: 'loginLayout',

msgReg: 'Email or phone is already registered',

success: false

})

}

else{

var temp = *req*.body.password

bcrypt.hash(temp, 10, function(*err*, *hash*) {

const user = **new** User({

role: 'Customer',

password: *hash*,

email: *req*.body.email,

phone: *req*.body.phone,

name: *req*.body.name,

birthday: *req*.body.birthday,

address: *req*.body.address,

avatar: 'sample-avatar.jpg'

})

user.save(*err*, *result* =>{

if(*err*) {

console.log('err: ' + *err*)

return *res*.render('register', {

title: 'Login',

layout: 'loginLayout',

msgReg: 'Failed to register',

success: false

})

}

return *res*.render('register', {

title: 'Login',

layout: 'loginLayout',

msgReg: 'Register Successfully',

success: true

})

})

})

}

})

.catch(*err* => {

console.log(*err*)

*res*.redirect('partials/error')

})

}

**Description**: The above code is written to perform the function of creating a Customer account. The above function is executed after the user enters the register page, enters all the information and clicks the Register button. Then the information will be sent to the system and processed through steps such as checking the existence, encrypting the password and saving it to the system, then redirected to the register page to notify the creation of success or failure.

**The usage**: To implement the above feature, users need to be able to access the register page, input full information such as name, date of birth, address, etc. Then click the Register button, submit the form data with the above information to the ./store route of the system. The system will check if the username (phone or email) already exists and if not, it will return a message, if it does, it will encrypt the password and create a new variable containing the above information under the role of a Customer and save it to the database.

**Evaluation:** Although the code does not have problems with system errors, but because it depends on MongoDB and has not handled system overloads, if there are too many people accessing and using it, the website may be stopped. active or default runs into error. Regarding user information, the above code has not created logic to check the authenticity of information such as incorrect date of birth, incorrect address, etc.

bcrypt.compare(password, *user*.password, function (*err*, *result*) {

if (*result*) {

var token = jwt.sign({ \_id: *user*.\_id }, 'secretpasstoken', {})

User.updateOne({ username: username }, { $set: { countFailed: 0 } }, (*err*, *status*) => {

if (*err*) {

console.log(*err*)

}

})

*res*.cookie('token',token, { maxAge: 2147483647, httpOnly: true });

return *res*.redirect('/')

}

**Description:** The above code is written to generate a token containing the information of the user who has logged into the system. This feature is based on npm's jwt (jsonwebtoken) library.

**The usage**: When users log into the system with Dustin's account, the system will check 2 information: username and password. It will first check if it exists and then check the password. However, the password has been encrypted with bcrypt, so to check, the system needs to decrypt the password through the password entered from the user. Along with the check, if the person enters the wrong information more than 3 times, they will be locked for 1 minute, if more than 5 times will be permanently locked and can only be unbanned by the admin.

**Evaluation**: The code works fine on all passwords entered from the user, error reports and account key information is updated via the notification on the login page. However, error messages from the server are not displayed directly, if the user encounters a server error, the website may be down.

*this*.add = function(*item*, *id*, *size*){

var storedItem = *this*.items[*id*];

if(!storedItem){

storedItem = *this*.items[*id*] = {item: *item*, qty:0, price: 0, size: *size*};

}

if(storedItem.size.indexOf(parseInt(*size*)) == -1){

storedItem = *this*.items[*id*] = {item: *item*, qty:0, price: 0, size: *size*};

}

storedItem.qty++;

storedItem.price = storedItem.item.price \* storedItem.qty;

*this*.totalQty++;

*this*.totalPrice += storedItem.item.price;

}

**Description**: The above code is written to implement the add to cart feature. As a website selling shoes, the shopping cart we designed will contain information such as: cart id, quantity and size of each pair of shoes.

**The usage**: When executing, the user will select a pair of shoes on the website and choose the size then click add to cart. The front-end will send the system information of the product such as size, quantity and id of the shoes. From there, the system will check if the shoes have been added to the cart or not, if not, it will create a new product in the cart with the information sent back with the default quantity of 1. If already created, 1 will be added to the number. quantity of that product and add 1 to the current cart, update the total amount. If the same pair of shoes but different size will also create a new pair of shoes in the cart.

**Evaluation**: Although based on the requirements and plans, the feature has been completed and tested thoroughly, but according to the logic of today's e-commerce sites, this feature has some unreasonable points such as: if user want to add 100 pairs to cart, then user need to click add button 100 times. This causes a bad user experience. Or for custom shoes, the user cannot see the difference of the custom shoes and added to the cart because there is no image displayed.

paypal.*payment*.create(create\_payment\_json, function (*error*, *payment*) {

if (*error*) {

throw *error*;

} else {

for(let i = 0; i < *payment*.links.length; i++) {

if(*payment*.links[i].rel === 'approval\_url'){

*res*.redirect(*payment*.links[i].href);

}

}

}

});

**Description**: The above code is used to generate a payment request by Paypal, using the paypal-rest-sdk library to use the payment functions. After the client side enters the payment information into the form data and sends it to the back-end, the system will generate a request url for the PayPal side. Since the data sent will contain payment information, after successful or failed payment, will return a previously described url like /success or /error and the system will notify the client and take action on that payment.

**The usage:** With variable create\_payment\_json containing payment information such as name, product, price and quantity. The library provides information to the PayPal side about the current transaction, thereby creating a form data to send. If there is an error in the sending process, an error will be displayed to notify the client, if not, it will be redirected to PayPal's payment page and PayPal will provide information for the user to pay after logging in. system.

**Evaluation**: The site is able to deposit and pay with PayPal in a stable way, the functions created also ensure to debug many cases of risks that are caused by the redirection process.

const *https* = require('https');

const options = {

hostname: 'test-payment.momo.vn',

port: 443,

path: '/v2/gateway/api/create',

method: 'POST',

headers: {

'Content-Type': 'application/json',

'Content-Length': Buffer.byteLength(requestBody)

}

}

//Send the request and get the response

const request = *https*.request(options, *response* => {

*response*.setEncoding('utf8');

*response*.on('data', (*body*) => {

var paymentUrl = JSON.parse(*body*).payUrl;

*res*.redirect(paymentUrl);

});

})

request.on('error', (*e*) => {

console.log(`Problem with payment Momo: ${*e*.message}`);

});

request.write(requestBody);

**Description**: Momo is a widely used E-wallet in Vietnam, so in our website we have created the feature to deposit and pay with Momo. On the Momo side, they request to receive the request with an https request contained in an object named options containing information such as hostname, port, etc. After sending, Momo side will process and return a JSON as an object, and the system will redirect to the web page after parsing.

**The usage:** This function is designed because most users in Vietnam often use Momo more than international payments because of intermediary fees. The feature is fully used on the Momo test provided by the developer, and payment information such as product price is also updated and changed depending on the data sent. Notifications of successful or failed payment are recorded by the server and returned to the client side.

**Evaluation**: The page has a stable ability to deposit and pay with Momo, the functions created also ensure to debug many cases of risk caused by the redirection process.

var transporter = nodemailer.createTransport({

service: 'gmail',

auth: {

user: process.env.NODEMAILER\_USERNAME,

pass: process.env.NODEMAILER\_PASS

}

});

var mailOptions = {

from: { name: "DUSTIN-SHOP", address: process.env.GMAIL },

to: *req*.body.email,

title: 'Activation Account',

subject: 'DUSTIN STORE - Reset Password for Account',

text: `Information about this:

New password: ${temp}

`

};

transporter.sendMail(mailOptions, function (*error*, *info*) {

if (*error*) {

*req*.flash('failMessage', 'Can not send email. Please try again')

console.log(*error*);

} else {

var noti = **new** Notification({

user: *user*.\_id,

desc: 'Your password is changed.'

})

noti.save()

*req*.flash('successMessage', 'New password has sent to' + *req*.body.email)

console.log('Email sent: ' + *info*.response);

*res*.redirect('back')

}

});

**Description:** Next is a password reset feature for customers if they forget their password. This function uses email to reset their password, to send mail we choose to use node mailer and send Gmail through Google account.

**The usage:** Although this function can reset the password, the user can only reset it by email, and if an attacker knows the user's email, he can reset all passwords to destroy the accounts in the system. If the user can access the core email, the password can be reset easily.

**Evaluation:** The function can be used well, but the logic of resetting the password is not accurate and has many limitations. Low security leads to accounts that can be easily sabotaged.

router.use(session({

secret: 'dustinsecret',

resave: true,

saveUninitialized: true

}));

router.use(passport.initialize());

router.use(passport.session());

router.use(cookieParser());

passport.use(**new** FacebookStrategy({

clientID: CLIENT\_ID\_FB,

clientSecret: CILENT\_SECRET\_FB,

callbackURL: "http://localhost:5000/auth/facebook/callback",

profileFields: ["id", "displayName", "name", "gender", "email", "photos"]

},

function(*token*, *refreshToken*, *profile*, *done*) {

process.nextTick(function() {

User.findOne({ 'facebookId' : *profile*.id }, function(*err*, *user*) {

if (*err*)

return done(*err*);

if (*user*) {

return done(null, *user*);

} else {

var newUser = **new** User();

newUser.email = *profile*.emails[0].value

newUser.facebookId = *profile*.id; id

newUser.name = *profile*.displayName

newUser.money = 0

newUser.gender = *profile*.gender

newUser.avatar = *profile*.photos[0].value

newUser.save(function(*err*) {

if (*err*)

throw *err*;

return done(null, newUser);

});

}

});

})

}

));

passport.serializeUser(function(*user*, *done*) {

done(null, *user*.id);

});

**Description**: As an e-commerce site, the login function is essential so that the system can know the needs and preferences of the user to easily advertise the appropriate products to the user. This functionality is created using libraries like passport-facebook and passport to create a new account based on the JSON returned by facebook. Then parse and save the data into the user's database with information such as: name, email, age, avatar, etc. After successful creation, the id will be generated into a token and saved in the passport to be able to access the account into the system.

**The usage**: This function is used to give customers more options when accessing the website. Like Facebook, Google is also designed the same and has 2 options for users to autoize on Dustin.

**Evaluation:** This function has been completed but can only be used as localhost even though the website has been deployed on heroku. Because the deployed website does not meet the security criteria, facebook does not accept login as a legitimate website.

## 6.4 Evaluation of the product

Following Dalhousie University, there are 6 main criteria to evaluate whether a website is good or not. The six criteria are (Dalhousie, 2022): AUTHORITY, PURPOSE, COVERAGE, CURRENCY, OBJECTIVITY, ACCURACY. So, before I judge the website by myself, I will evaluate the Dustin website based on the above 6 criteria to evaluate the website.

Starting with **Authority**, our website needs to identify the person, organization or authority responsible for each function and system of the website. Regarding the person responsible for the website's features, I - the backend developer will be responsible for the website's features as well as handling user information, encrypting personal information such as passwords, addresses, etc. And Dong - frontend developer will be responsible for the design of UI, user interaction. On the 3rd party side, the 3rd parties working with the website will be responsible for their part such as: Database will be responsible by MongoDB, as well as E-wallet payment by Momo, Paypal. To ensure this, we must accept the legal and use policies they have set forth to ensure the Authority of the Website.

Next is **Coverage**, it is difficult to assess the extent of coverage since depth in a site, through the use of links, can be infinite. In order to properly evaluate Coverage, I will answer the questions asked that the topics covered by the requirement are thoroughly addressed, the arguments are clearly presented, and are fully supported for presentation. Not only that, the updated information from previous projects ensures not to repeat the errors that previous projects have encountered and is always updated with new trends and modern technologies. From there, it is possible to determine the target audience that matches the requirements of the system.

The **Purpose** of the information presented in the site should be clear. So, we have planned and carefully analyzed the requirements from users to make a list of features and user requirements before implementing the project. This helps the content proposed to be relevant and helpful for the project and at the same time targeting specific customers such as sneaker enthusiasts, customers looking to buy shoes, etc.

There are 2 main pieces of information that website **Currency** refers to: 1) how current the information is presented and 2) how often the website is updated or maintained. To answer the above question in my website, information about the site is presented in the site information section or at the bottom of each page. The information will mention the creation date, development date as well as the last update date of the website. The website will be checked and upgraded every week if there are no unexpected errors.

Regarding **Objectivity**, the website is created on the basis of an e-commerce website, so the website will be used to sell items online - Sneaker. However, the website information will not be used to sway, persuade customers or explain to customers whether to buy products at the door or for the customers' sake. The website only aims to bring products to people in the easiest way to experience. However, as an e-commerce website, the website will try to explain and convince users to buy products in the store, and the website's advertising will sometimes conflict with the content because other products each other from various sources.

Next is **Accuracy**, there are few standards to verify the accuracy of information on the web. It is the responsibility of the reader to assess the information presented. So, if I were a client, I would rate the site by its reliability, references, spelling of words, and accuracy of metrics. In terms of reliability, because the website is only running on a free domain - Heroku, it is difficult to guarantee the security of an https website, which creates a domain that is not reliable for customers and 3rd parties like Google, Facebook. And the website is set up with the main language of English, so the language and spelling may only be 90% accurate, sometimes some sentences will be a bit confusing and difficult to understand. This is the word point for Dustin.

Finally, through the evaluation based on 6 criteria of Dalhousie University, I realized that although the website has been completed, there are still some difficulties that have not been overcome in some criteria such as Objectivity and Accuracy. But the website is still in the process of development and implementation, so on the development side, we will continue to update and improve the features and weaknesses from the above criteria, from which the website will become more and more complete. improve and ensure the criteria set out at the beginning. However, the remaining criteria are closely followed and completed in a clear and guaranteed way, which makes the website still able to ensure the basic criteria and features that can improve the user experience. This is a good thing for one of our e-commerce sites.

# Chapter 7 Conclusions

## 7.1 Learned from the project

After completing the product, I have accumulated a lot of experience not only in coding, but also gained more knowledge about the process of building and executing projects as well as learning how to work as a team with tools. This is our first complete project of an e-commerce website full of basic features and we also designed a special function to ensure that we can attract customers as well. as customers. create a breakthrough compared to current e-commerce.

Starting with what we learned about coding, about the technologies used in the project, we choose to learn and use HTML, CSS and JavaScript to implement the user interface in the project, not just with Handlebars (Hbs) help helped us to render in CLI app, not Text-HTML - plain content, server-side rendering. Not only that with the APIs passed up from the back-end of the system, Hbs helps us to render more easily. With some support from Hbs we can create functions more concisely and easily. Easy to use in HTML files. Next is SCSS, which empowers front end developers to enhance their user interface skills with unique features that extend CSS capabilities. Although people often choose to use SASS more than SCSS, the two technologies are aimed at making it easier for developers to manipulate CSS as well as easier to maintain, organize and improve in the future. Finally, JavaScript, a website without Js will be quite dry and not attractive to customers with a simple interface and not many effects. Js is a vital technology for every project, but utilizing pure Js would be rather difficult and time-consuming, thus we choose to utilize jQuery, a Js library capable of doing several jobs. Common tasks that need numerous lines of JavaScript code to perform are encapsulated into methods that may be called with a single line of code.

Next, we also learned about the process of building and implementing the project. Regarding construction, we apply the knowledge gained from the Spiral model to implement the project because of the benefits that this methodology brings such as: Risk Handling, Flexibility in Requirements and Customer Satisfaction. Because this is our first project, it is certain that the risks that we research in advance will not be complete and comprehensive, leading to a lot of unexpected risks when implementing. handle. Therefore, the Spiral Model is the best development model to follow due to risk analysis and risk treatment at every stage. Not only that, but this model is flexible in all requirements because requirements that change in Requests at a later stage can be precisely matched using this model.

And finally, the experience and knowledge gained from teamwork. When working in a team, we need to make a complete plan with clear division of work and the proposed time needs to be researched carefully to avoid the case that some tasks need more time to complete in the day. while others don't need too much time. Another experience is that when working as a team, we can refine the features and direction of the project through discussions and meetings to ensure the project is completed according to the requirements from the system as well. such as ensuring errors need to be handled accurately and easier than working alone. More specifically, we can divide complex tasks into parts and steps so that we can work together faster and more securely.

## 7.2 Result of the project

Regarding the output of the project, I will divide into 2 main parts of the results, the results from the Back-end and the results from the Front-end.

Regarding the Back-end, we use NodeJS technology to program along with MongoDB for the database. The first is an e-commerce website project to sell shoes, I have to ensure basic features in CRUD and some encryption features to protect user information. Especially the function of customizing shoes to create a pair of shoes with color codes that customers like and can save in the cart as well as order after completion. Regarding registration and login, we also created a login feature using social media such as Facebook and Google to help users easily log in to the system instead of having to register cumbersome and manually. However, the traditional registration function is still designed so that users who do not have Google and Facebook accounts or do not want to use those accounts can register for an account in this system. And an e-commerce website, the payment function is extremely important, we have many difficulties to design and choose 3rd parties to implement this feature. We decided to choose Momo and PayPal for e-wallet payments and Stripe to simulate the ability to pay by credit card. For the user, I have designed the models to contain information about notifications and histories so that the user can know whether the payment was successful or not or any notifications from the system. Regarding user personal information, we use bcrypt for encryption to protect user information. We also integrate a forgot password feature for users to use when they forget their password, and when selected, the system will send an email via the Gmail registered by the user. Another feature is the Dustin wallet, a wallet used in the Dustin store to pay for products. Users can top up via E-wallet or Credit card. However, the website is programmed to execute requests from users, so there is no guarantee of security. Security technologies have not been applied to the system much, the website is prone to information leakage. Every model in Dustin can be created with an uploaded image then saved in project folder, this is only a temporary solution for storage

For the Front-end, we choose to use Hbs(HTML), SCSS(CSS) and JavaScript to design the interface and retrieve information from the database for the e-commerce website. In order for the website to be usable on many devices, we decided to design it as a Website Application with full Responsive for all pages so that users can access on multiple devices. We also spent 2 weeks researching and coming up with a feature called custom shoes, this is a feature that helps users choose the color for each part of the shoe that the system has given in advance. With limited time, we only planned 2D shoe custom with 1 pair of shoes and 6 main parts. This custom color will be chosen by the customer in 1 of 16 million colors of RGBA. Information about the custom shoe is saved as a color code from RGBA, not the actual image seen by the user, so this feature is not guaranteed. However, it can be said to be successful in terms of implementation and results.

## 7.3 Further development of this project

Regarding future developments, for the Back-end, we will focus on applying security technologies to protect user information and also avoid information leakage and theft. The second is to increase the performance and speed of the website, we will re-implement the code to make it shorter and easier to understand. As well as applying some technologies to help store user data and sessions like Firebase to reduce the amount of data in the code for faster execution and retrieval. From there, it is possible to improve the user experience for the product. Second, we will apply the use of API instead of Restful API to make the division of work easier and clearer, avoiding dependence on each other. Language-wise, TypeScript will be used instead of JavaScript because of its more maintainability and programming capabilities, but we also consider this option because of the need to rebuild the entire project.

For the Front-end, currently using Bootstrap and jQuery to execute the code, but this makes the code dependent on the above libraries, making maintenance and upgrading worse and more difficult when working and help if someone doesn't use the above libraries. So, we are working on applying frameworks to the website to improve the user experience as well as make maintenance and upgrades easier. For example, ReactJS or VueJS for programming on Single page instead of Multi pages as it is now. Although the switch to using Single Page websites caused Front-end developers to rebuild the website with more complex features, it nevertheless resulted in a better user experience.

Finally, about the Customize function, as mentioned above with limited time we can only do 2D Customize, however with our original plan and goal is to Customize 3D shoes with most models. Rich and customizable nowadays. There were some issues we had when we tried to implement this feature. We didn't have a 3D design team for the original shoe, which made programming more difficult. Secondly, because of the lack of initial 3D design, testing this feature on html-powered canvas bodies is not feasible and as well as lack of experience in working with 3D design, we are still working on the feature. this.

# REFERENCE

C. Wangmo, “E-commerce picks up, as Covid-19 discourages outing," Kuensel, 7 April 2020.

S. E. Ullah, T. Alauddin and H. U. Zaman, "Developing an E-Commerce Website," in 2016 International Conference on Microelectronics, Computing and Communications (MicroCom), Durgapur, 2016.

M. S. Kandhari, F. Zulkernine and H. Isah, "A Voice Controlled E-Commerce Web Application," in 2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), Vancouver, 2018.

D. N. Naidu, P. Adarsh, S. Reddy, G. Raju, U. S. Kiran and V. Sharma, "E-Commerce web Application by using MERN," International Journal for Modern Trends in Science and Technology, vol. 7, no. 05, pp. 1-5, 2021.

P. Umberto, "Developing a price-sensitive recommender system to improve accuracy and business performance of ecommerce applications," International Journal of Electronic Commerce Studies, vol. 6, no. 1, pp. 1-18, 2015.

M. B. Subba, "Home delivery and online shopping to stay," Kuensel, 18 August 2020.

Aritra, B. and Rajasi, D. (2020) Role of social media and e-commerce for business entrepreneurship. Available at: <https://www.researchgate.net/publication/348460175_Role_of_Social_Media_and_E-Commerce_for_Business_Entrepreneurship> (Accessed: August 25, 2022).

Dalhousie University (2022) 6 criteria for websites - dalhousie university. Available at: <https://cdn.dal.ca/content/dam/dalhousie/pdf/library/CoreSkills/6_Criteria_for_Websites.pdf> (Accessed: September 11, 2022).

Mailchimp (2021) E-commerce websites: What they are and how to build one, Mailchimp. Available at: <https://mailchimp.com/marketing-glossary/ecommerce-website/#:~:text=A%20website%20that%20allows%20people,logistics%2C%20and%20provide%20customer%20service> (Accessed: August 22, 2022).

Nagothu, D. et al. (2021) E-commerce web application by using Mern Technology. Available at: <https://www.researchgate.net/profile/Vikas-Sharma-35/publication/353481574_E-Commerce_web_Application_by_using_MERN_Technology/links/60ffd3550c2bfa282a035023/E-Commerce-web-Application-by-using-MERN-Technology.pdf?_sg%5B0%5D=started_experiment_milestone&amp;origin=journalDetail> (Accessed: August 22, 2022).

Rathnayaka and Kumara (2022) A review of software development methodologies in software ... - IJARIIE. Available at: <https://ijariie.com/AdminUploadPdf/A_Review_of_Software_Development_Methodologies_in_Software_Engineering_ijariie12553.pdf> (Accessed: September 6, 2022).

Sangay, T., Tshering, L. and Tsheten, D. (2022) Design and development of e-commerce web application for Cooperative Store. Available at: <https://www.researchgate.net/publication/358927961_Design_and_Development_of_E-Commerce_Web_Application_for_Cooperative_Store> it(Accessed: August 22, 2022).