VIETNAM NATIONAL UNIVERSITY OF HO CHI MINH CITY

INTERNATIONAL UNIVERSITY

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

WEB APPLICATION DEVELOPMENT

SELLING WEB

Lecturers

Dr. Nguyễn Văn Sinh

Mr. Phạm Quốc Sơn Lâm

By

Nguyễn Tố Quyên – ITITIU16050	front-end, back-end, report
Đàm Phi Hải – ITITIU16087	test, database
Nguyễn Đỗ Cường – ITITIU16076	back-end, database

Ho Chi Minh City, Vietnam

2021

TABLE OF CONTENTS

CHAPTER I: INTRODUCTION

1. Motivation

These days, businesses are conducted not just in physical stores but also online. The internet has become the most popular method of selling and buying. The market is growing regularly, especially during the Covid period, and internet shopping has become an integral feature of any business. Recognizing the appeal of online shopping, our company is working on creating an online website with some basic services to assist customers in displaying things.

We have created a more robust application. The actual initiatives, on the other hand, are difficult. The goal is not to teach us HTML/JavaScript, but to give us a real-world scenario and guide us through creating simple applications with the tools. This project is for students who have finished the Web Application Development course, such as us.

2. Problem statement

Perfume World is storage for perfumes. By access to our website, customers could find it easy to view various products and decide upon purchasing the right product to meet their requirements. The system allows customers to access, view products, buy products, and advise online. The major benefits are that the entire process is online, which means the customer can use it anywhere anytime.

The website is to be developed for the Windows Platform using HTML5, Servlet, JavaScript, and Geolocation. The site should work well in all leading browsers including Chrome, Internet Explorer, Firefox, Opera, etc.

3. Scope

Time and financial constraints. This project will be completed on June 20, 2021. This project will take three months to complete. Furthermore, the time limit will be established as the project progresses and after the lab sessions.

Our team will require a budget to do this project. This sum is divided into three categories: hardware, software, and human resources. The following is an estimate of the cost:

- Hardware budget: $2500
- Software budget: $500
- Human budget: $100
- Additional budget: $900
- Total cost estimated: $4000

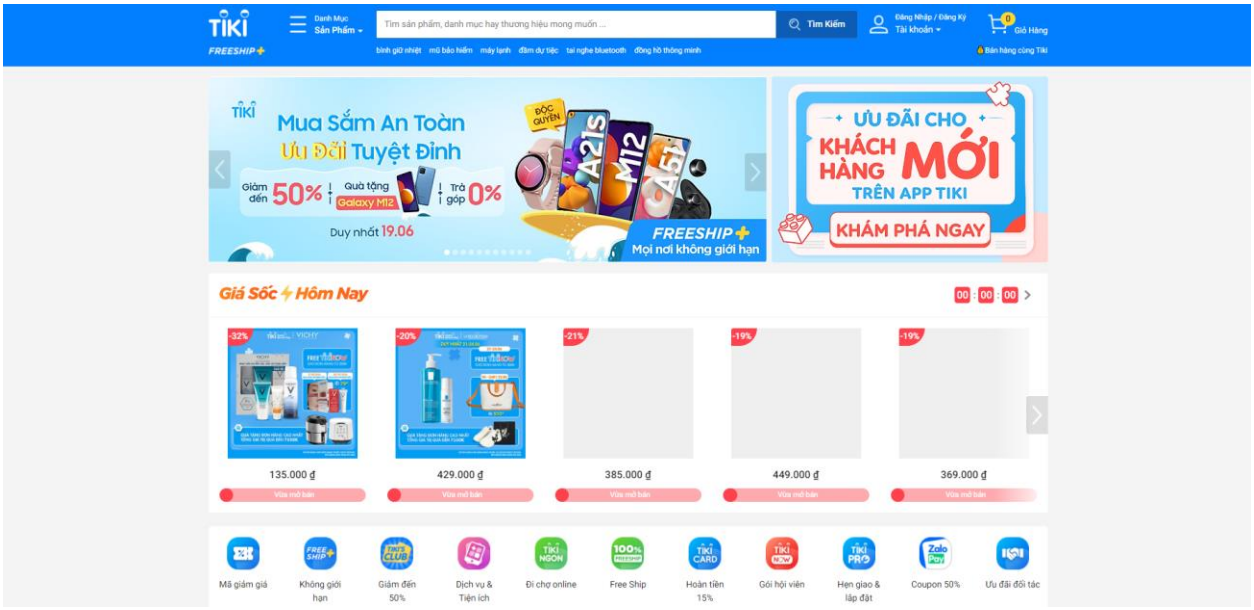
CHAPTER II: LITERATURE REVIEW

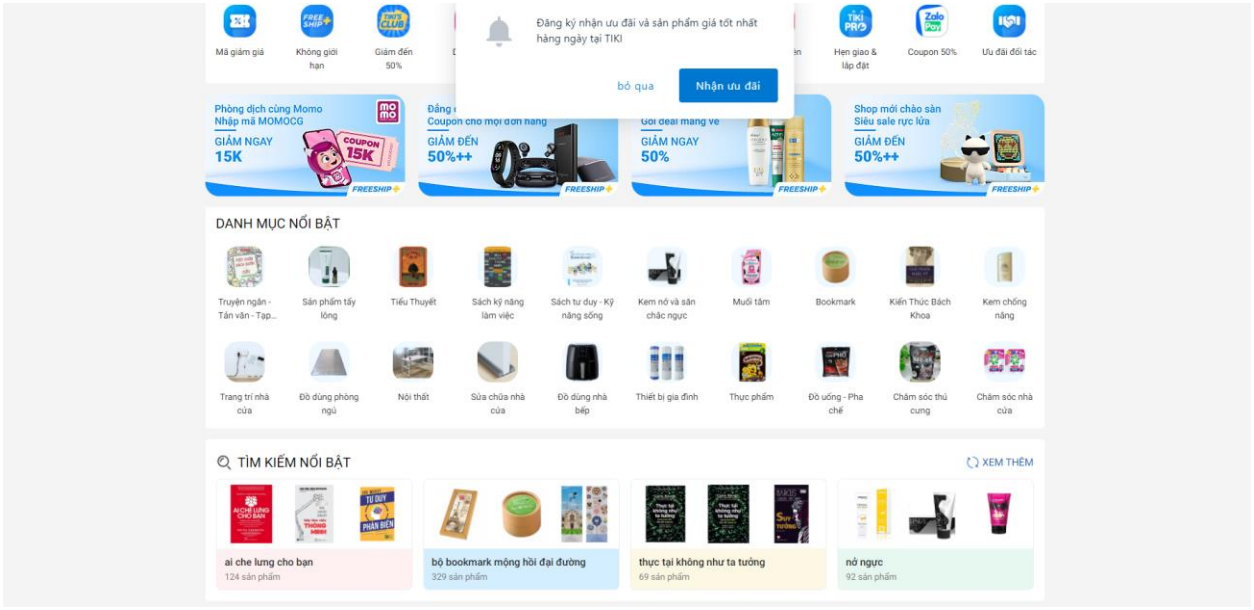
1. Similar applications/systems

Tiki, Shopee, and Lazada are just a few examples of such platforms. Before we go on to the next chapter, let us take a short look at their characteristics. Both websites serve as go-betweens for clients and distributors.

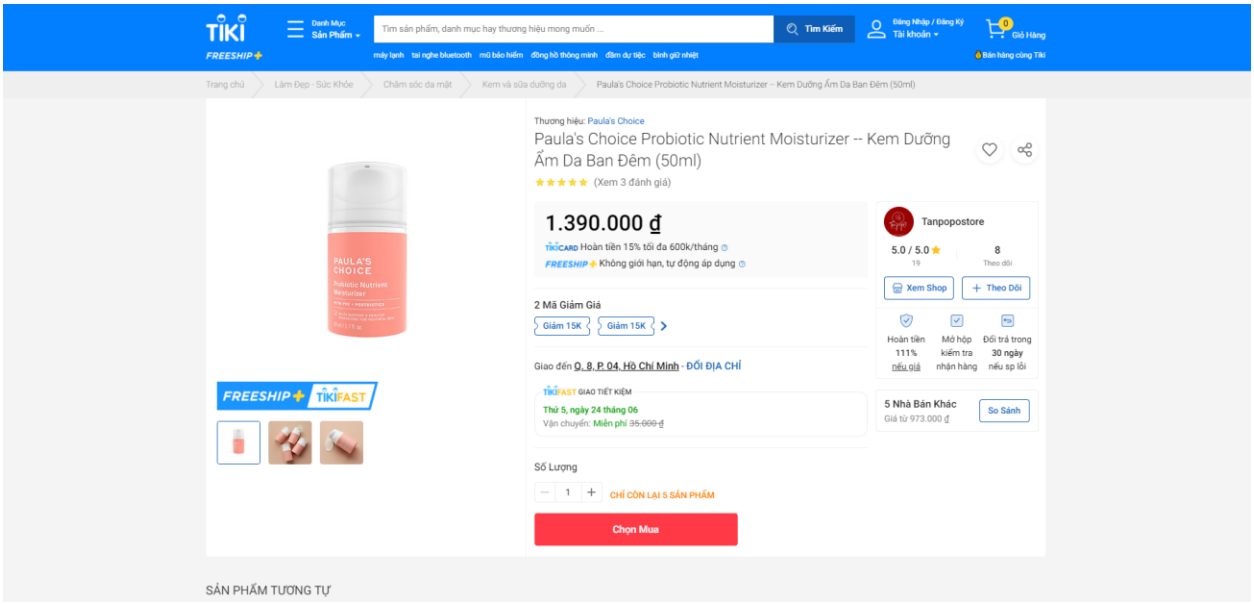
a. Tiki

The homepage.

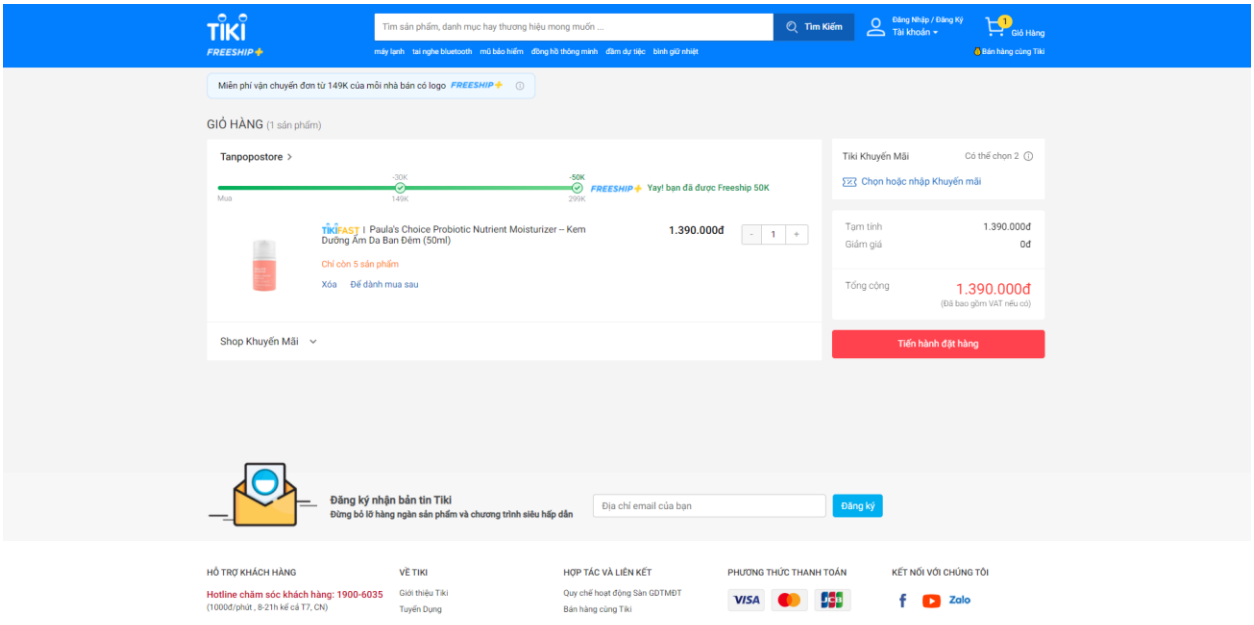




The product pages.

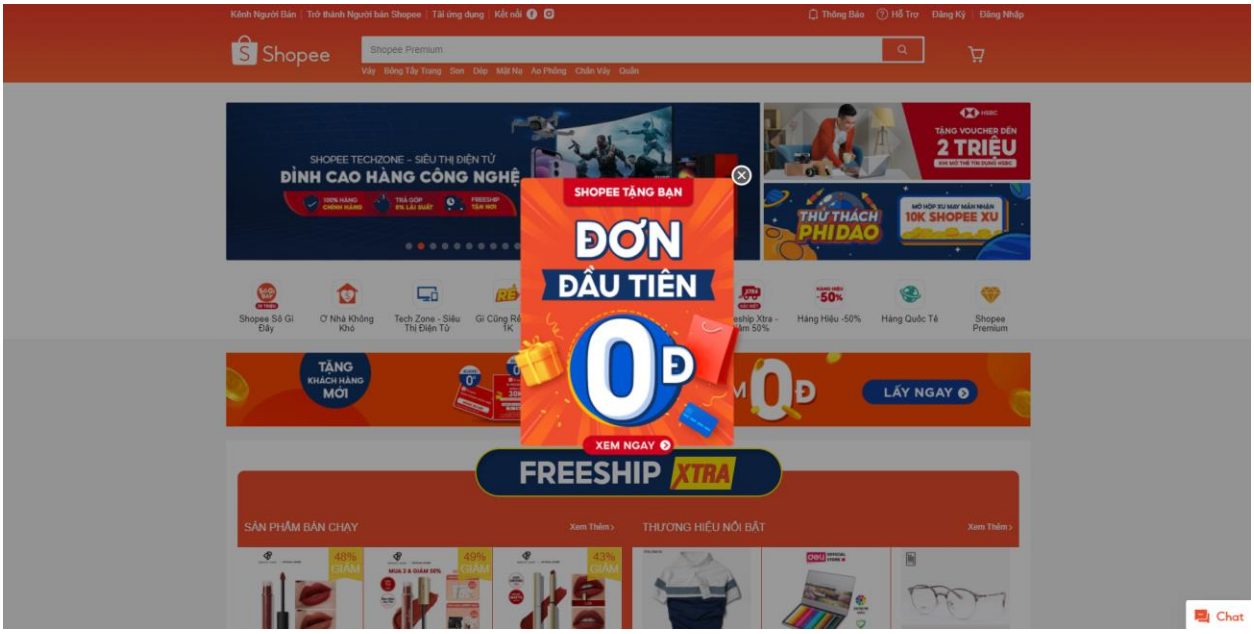


The purchase pages.

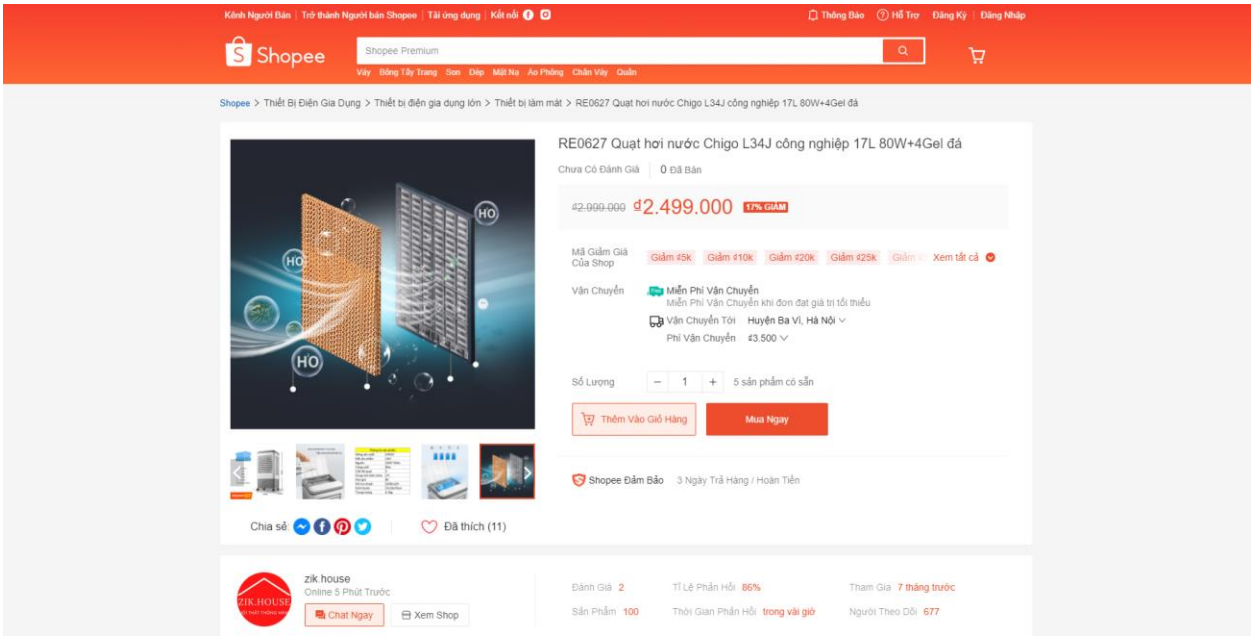


b. Shopee

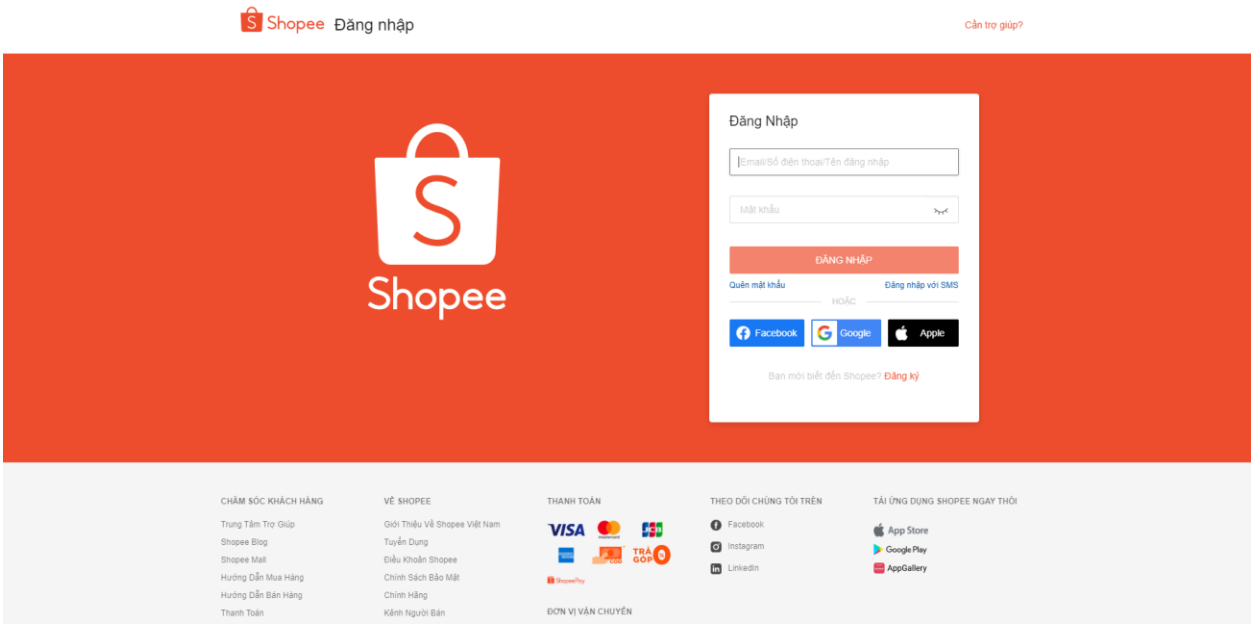
The homepage.



The product pages.



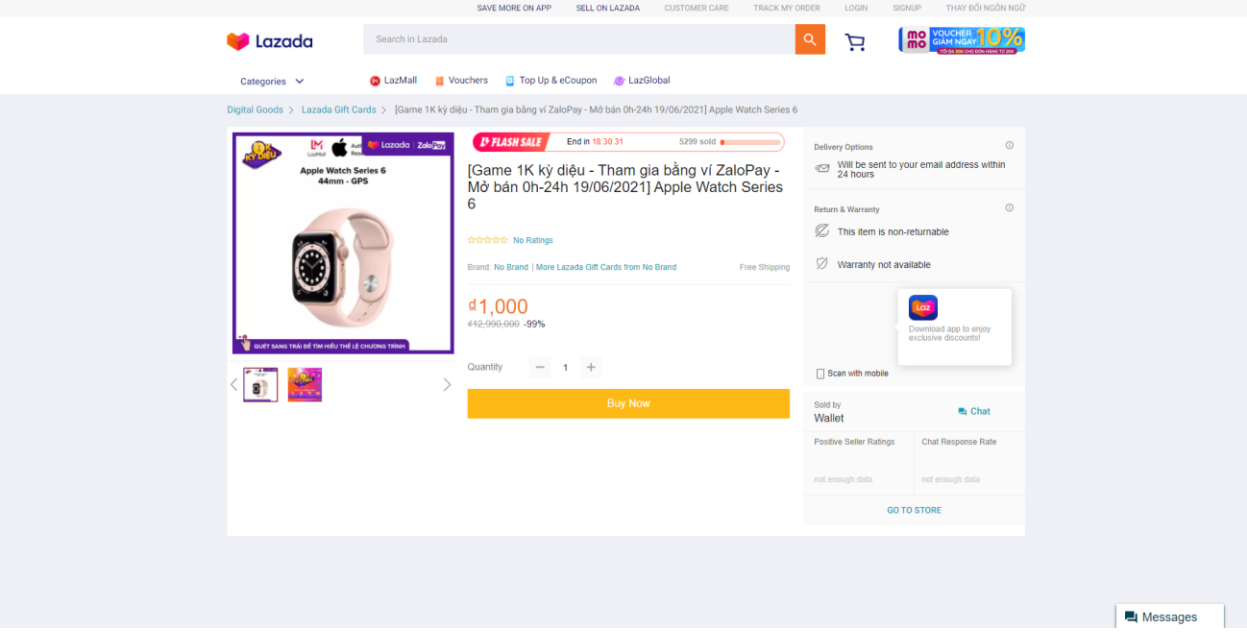
The account login and register.



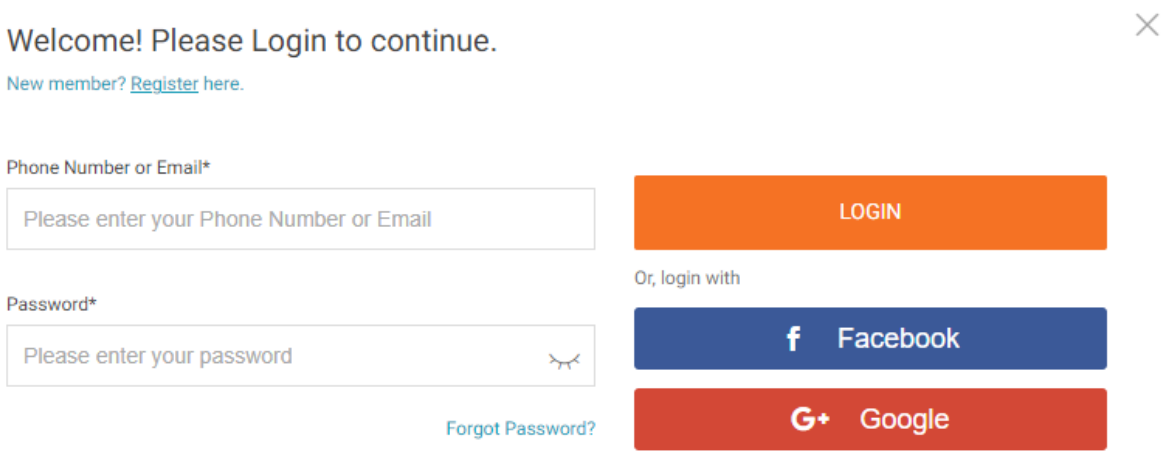
c. Lazada

The homepage.

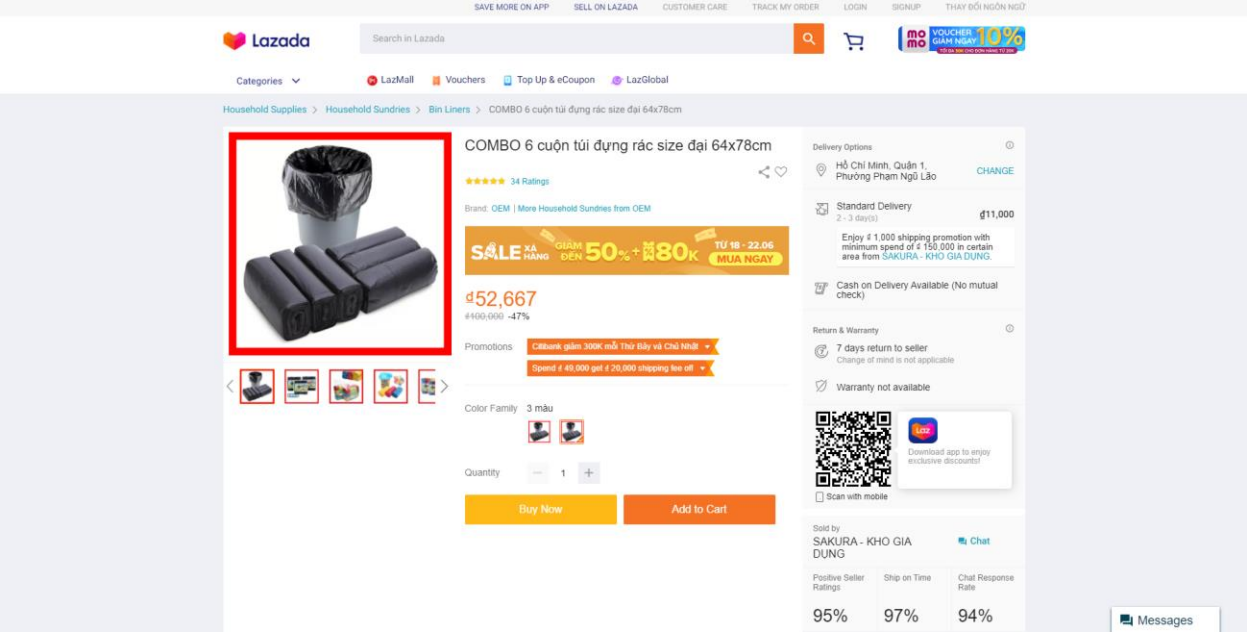
The product pages.



The account login and register.



The information of products.



2. Platform and tools review

Operating system Window 10.

Web browser Opera, Chrome, etc.

Hardware Laptop EliteBook HP.

Software

- Eclipse is a programming IDE. It comes with a default workspace and a plug-in framework for customizing the environment. It comes with a standard workspace and a plug-in framework for customizing the environment. Eclipse is primarily built in Java and is intended for the development of Java applications; however, it may also be used to construct programs in other programming languages via plug-ins, such as Ada, ABAP, C, C++, C#, Clojure, COBOL, D, Erlang, Fortran, Groovy, Haskell, JavaScript, Julia, Lasso, Lua, NATURAL, Perl, PHP, Prolog, Python, R, Ruby (including Ruby on Rails framework), Rust, Scala, and Scheme. It can also be used to create LaTeX papers (with the help of a TeXlipse plug-in) and Mathematica packages. Eclipse Java development tools (JDT) for Java and Scala, Eclipse CDT for C/C++, and Eclipse PDT for PHP are just a few examples of development environments.
- Microsoft SQL Server is a relational database management system that was created by the company Microsoft. It is a database server, which is a software product whose principal role is to store and retrieve data as required by other software applications, which may run on the same computer or a networked computer (including the Internet). Microsoft SQL Server is available in at least a dozen different editions, each geared at a

different audience and suited to varied workloads ranging from modest single-machine applications to huge Internet-facing systems with many concurrent users.

Language

- Hypertext Markup Language (HTML) allows us to generate and structure web page components, such as paragraphs, headings, and footers. Developers use HTML to create a user interface with a variety of fundamental tasks and features, such as displaying text, photos, and videos, creating a clickable button, and embedding hyperlinks.
- The cascading style sheet (CSS) is a language support developer that defines the style and format aspects of markup languages, such as HTML. CSS optimizes the coding process and saves time for developers by controlling the format of several web pages at the same time. The separation of HTML and CSS makes it easier to maintain websites, share style sheets across pages, and customize pages for diverse situations.
- JavaScript, or JS for short, is a basic, object-oriented programming language that is cross-platform. JavaScript is a compact, lightweight programming language. JS is compatible with connecting to other objects in a host environment and providing methods to handle those objects whenever it is implemented in that environment. JavaScript plays a vital function in assisting developers in creating animations and slideshows that make websites more user-friendly. Developers can also use JavaScript to create validation and verification routines.
- Servlet, formerly known as Java Servlet, is a Java software component that enhances a server's capabilities. Although servlets can reply to a wide range of requests, they are most typically used to host web applications on web servers, making them a server-side servlet web API. Other dynamic online content platforms, such as PHP and ASP.NET, have Java counterparts in the form of web servlets.
- Structured Query Language (SQL) is a computer language that is used to handle data in a relational database management system (RDBMS) or for stream processing in a relational data stream management system (RDSMS). It is especially beneficial for dealing with structured data, or data that has relationships between entities and variables.

CHAPTER III: SYSTEM DESIGN

1. System Requirement Specification

Non Functional requirements

No #1	**Environment**
1. Summary	The system is constructed on a solid foundation that must be both stable and upgradeable.
2. Rationale	When building an application, one of the most crucial aspects to consider is the environment. If the environment is unstable, it is risky for both the user and the developer, and it may have a significant impact on the application's revenues. The environment must also be upgradeable to accommodate future improvements and additional functions.
3. Requirements	The system is developed on the Eclipse framework. Using HTML5, Java (Servlet and JSP) language and database system: Microsoft SQL Server 2019.

No #2	**Performance**
1. Summary	The software has to be optimized for speedy loading.
2. Rationale	If a program takes too long to load, the user may become irritated and stop using it.
3. Requirements	The database is designed with BCNF. Hardware requires a 64-bit processor and operating system. Operating system: Windows 10 64-bit. Processor: Intel Core 2 Duo. Memory: 8GB RAM. Graphics: Intel HD Graphics. Network: Broadband Internet connection. Storage: 1 GB available space.

No #3	**Security**

1. Summary	External contacts between the data server and clients of the system must be encrypted.
2. Rationale	The system is only accessible to those who have been approved. The system must be secure enough to withstand outside attacks.
3. Requirements	Session.

No #4	**Understandability**
1. Summary	The user interface (UI) of a website should be welcoming, straightforward, and simple to use. The system's architecture is tidy, and the code is readable. It takes very little time to learn how to use it.
2. Rationale	As the number of users grows, the system must be straightforward to understand for a wide range of people. Because the software is used by many people on several platforms, it must have a user interface that is welcoming and responsive.
3. Requirements	HTML5, CSS3, JavaScript, Servlet, jQuery.

No #5	**Reliability**
1. Summary	Except when the input data is incorrect, the system shall not return any wrong results.
2. Rationale	When a user searches for programs, the results must be accurate and trustworthy, and all data must be permitted.
3. Requirements	On the website, you can find contact information. It is necessary to ensure that the search query result is accurate.

No #6	**Stability**
1. Summary	When the number of users grows, the system must remain stable.

2. Rationale	When a webpage becomes famous, the quantity and diversity of users increase. To offer a positive user experience, the webpage must be stable and maintain a fast loading speed.
3. Requirements	The database should be simple, yet adequate and upgradeable.

Functional requirements

No #1	**Strong password**
1. Summary	The function tool is to ask the user for a proper password.
2. Rationale	When register to the software, the user may be input a weak password. Then the accounts are not safe. So, we need to make sure that the user accounts are secure.
3. Requirements	There is a service to alert the register form validation.

No #2	**Sales report should be generated automatically**
1. Summary	The sales report will be automatically prepared.
2. Rationale	The auto report generator helps the manager save time while the monthly report helps the admin follow the store better.
3. Requirements	There is a service that can detect the end of the month, collect data automatically, and send the email. The manager can configure the information type, email, and send time in this section.

No #3	**Logging session**
1. Summary	While the user is still on the website, each logging session will be active.
2. Rationale	The logging session aids in the tracking of working time as well as the accountability for the order if something goes wrong.
3. Requirements	After a user logs into the system, all date and time information, as well as user data, is saved in the database.

Requirement analysis

UC1: Login

To login, show a window that asks for a username and password.

Validate user and password inputs against the database.

After logging in fails, redirect the user to the login page.

After logging in successfully, redirect the user to the category page.

Remember to log in to your account.

UC2: Register

Show a page that asks for the user information.

Verify the username, email address, phone number, and password.

After successful registration, create a new account in the database.

UC3: Buy Product

Show a page with all of the products that are offered.

Next to the products, place the "Add to Cart" button.

Next to the products, place the "Check out" button.

UC4: Check cart

The page where you may see your shopping cart.

Display product information that has been added.

Orders can be viewed by the administrator.

UC5: Give feedback

Show the feedback page.

Provide feedback via email.

UC6: See locations

A page for displaying locations.

For each location link, display a separate map.

UC7: Control user

Display a page containing user information.

Users can be added, deleted, viewed, or blocked by the administrator.

UC8: Control product

Display a page containing product details.

Admins can add, delete, and view products.

UC9: Make a report

Admins can generate reports whenever they wish.

Every month, send an email to the admin with a report.

Name	**UC-1: Login**
Summary	The user interfaces for logging in to the software. And the user's information at the time of the session.
Rationale	This is the initial step in determining if the user is a customer or an administrator. Furthermore, an anonymous user cannot access the database through the security gate.
Users	All users.
Preconditions	The old login session is expired.
Basic Course of Events	1. The user enters the username and password. 2. The user clicks the button.
Alternative Paths	When the user forgot the username or password. 1. Loading the log-in page.

Postconditions	The dashboard will show for the user.

Name	**UC-5: Give feedback**
Summary	Send the email to a single employee or a group of employees.
Rationale	For official notification, an email is a suitable option. Employees or managers will receive feedback via email on their work performance and business potential.
Users	All users except admin.
Preconditions	Open the website and click the “Send feedback” button.
Basic Course of Events	1. The user enters the subject and contents. 3. The user clicks the “Send feedback” button.
Alternative Paths	When the user does not input information but only clicks the button. 1. Click the “Send feedback” button and edit the email.
Postconditions	The email page will show for the user.

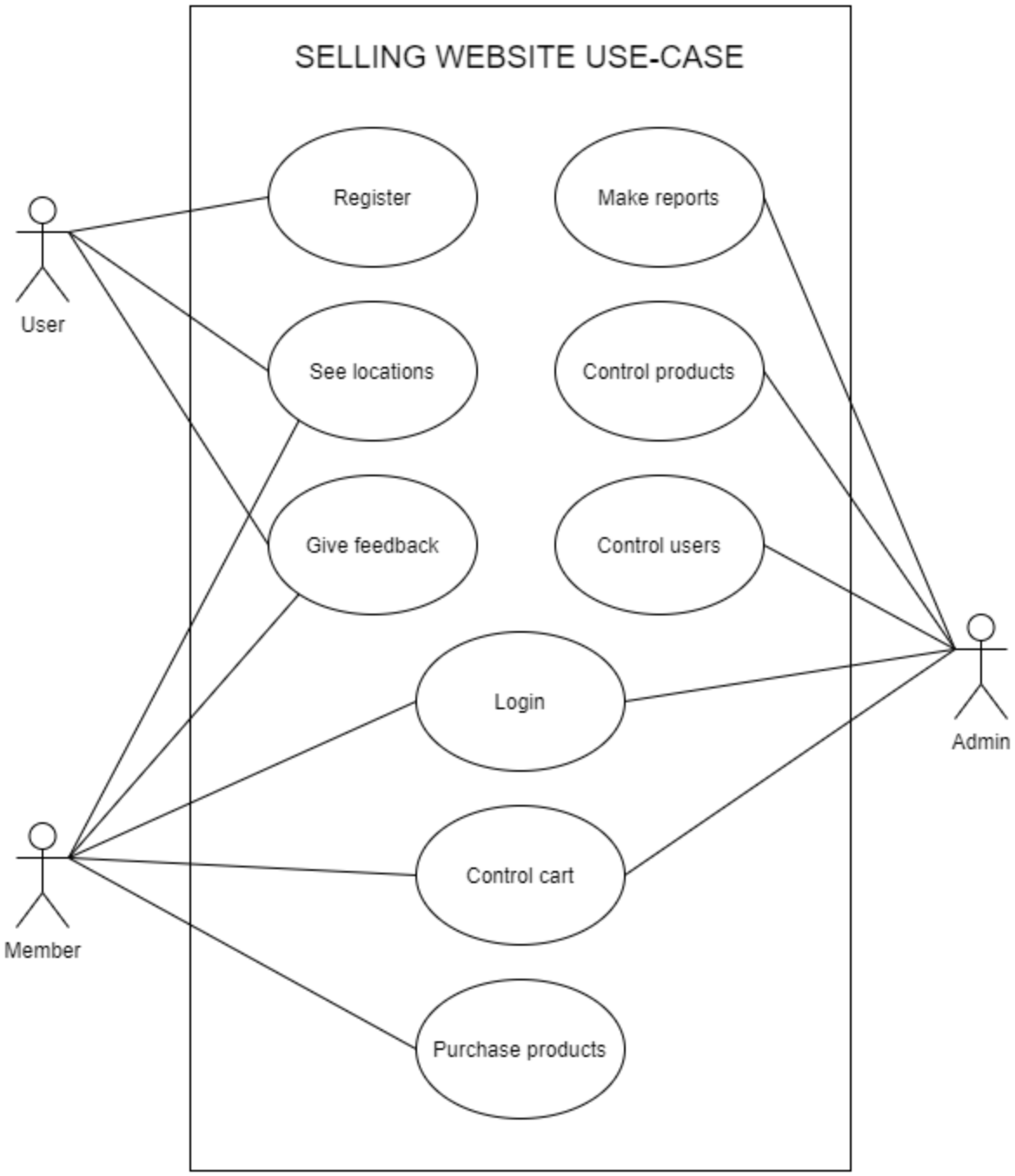
Name	**UC-9: Make a report**
Summary	Prepare a monthly report on the history of working hours and the state of the firm.
Rationale	The report is crucial for reviewing previous profits. Making the report will make it easier for the manager to keep track of the business, labor conditions, and make appropriate improvements to the set of goods or beverages.
Users	Admin.
Preconditions	Log in as Admin. The software is in the report interface.
Basic Course of Events	1. The administrator specifies the start and end dates for data collection, which can be monthly, annually, or weekly. 2. The software retrieves data for a specified period, calculates a summary, and plots a graph. 3. The item filter condition can be entered by the administrator. 4. The software re-fetches the data that has been filtered and re-summarizes it.
Alternative Paths	The date is not precisely defined by the administrator.

	1. If the admin selects monthly, annually, or weekly, the software will determine the end date and start date automatically.
Postconditions	The report is created, and the option to transmit it through email is displayed.

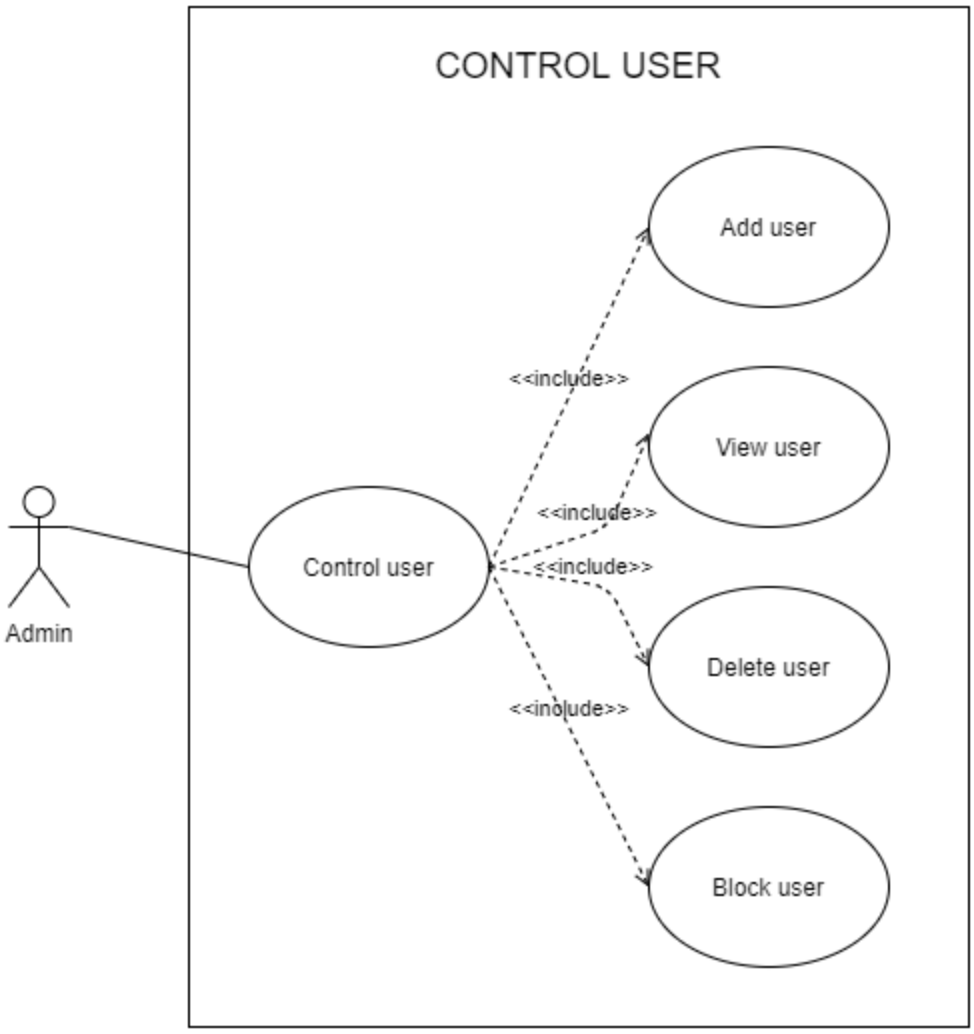
2. System Design Specification

Use-cases diagram

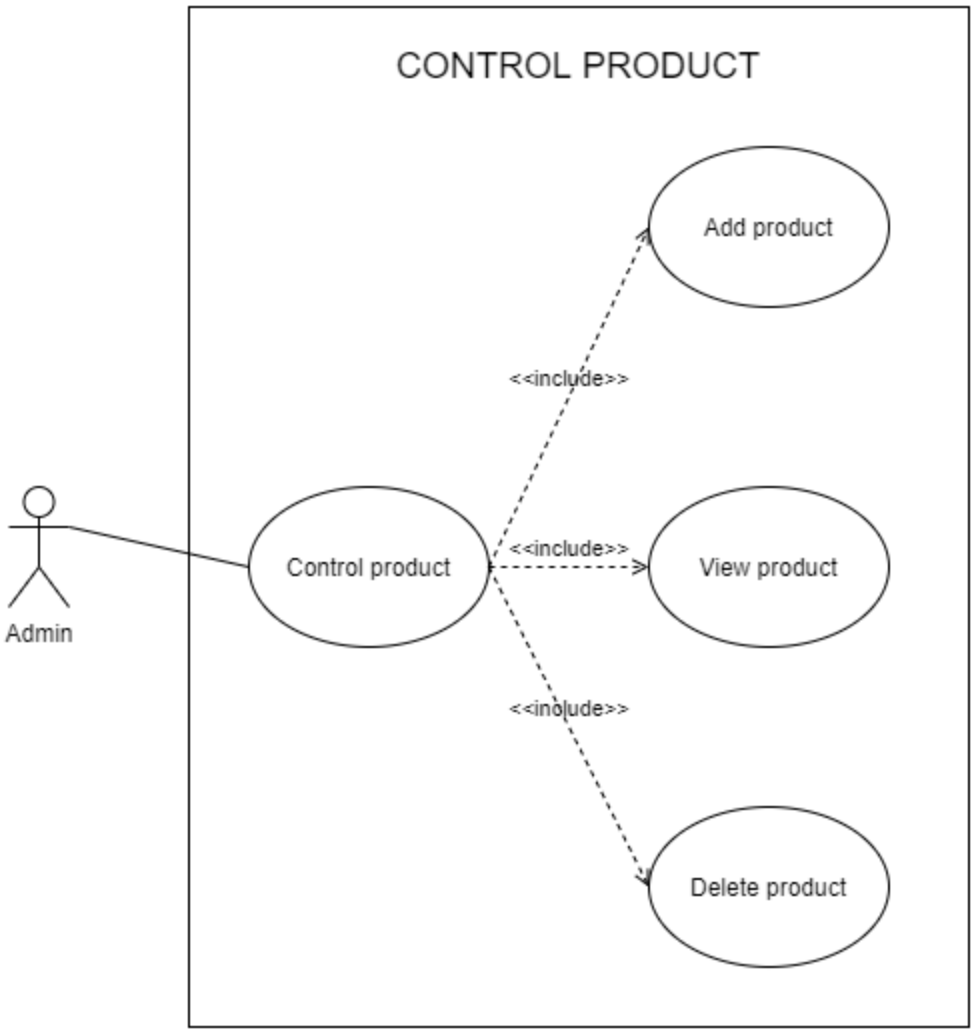
- Summary goal use case



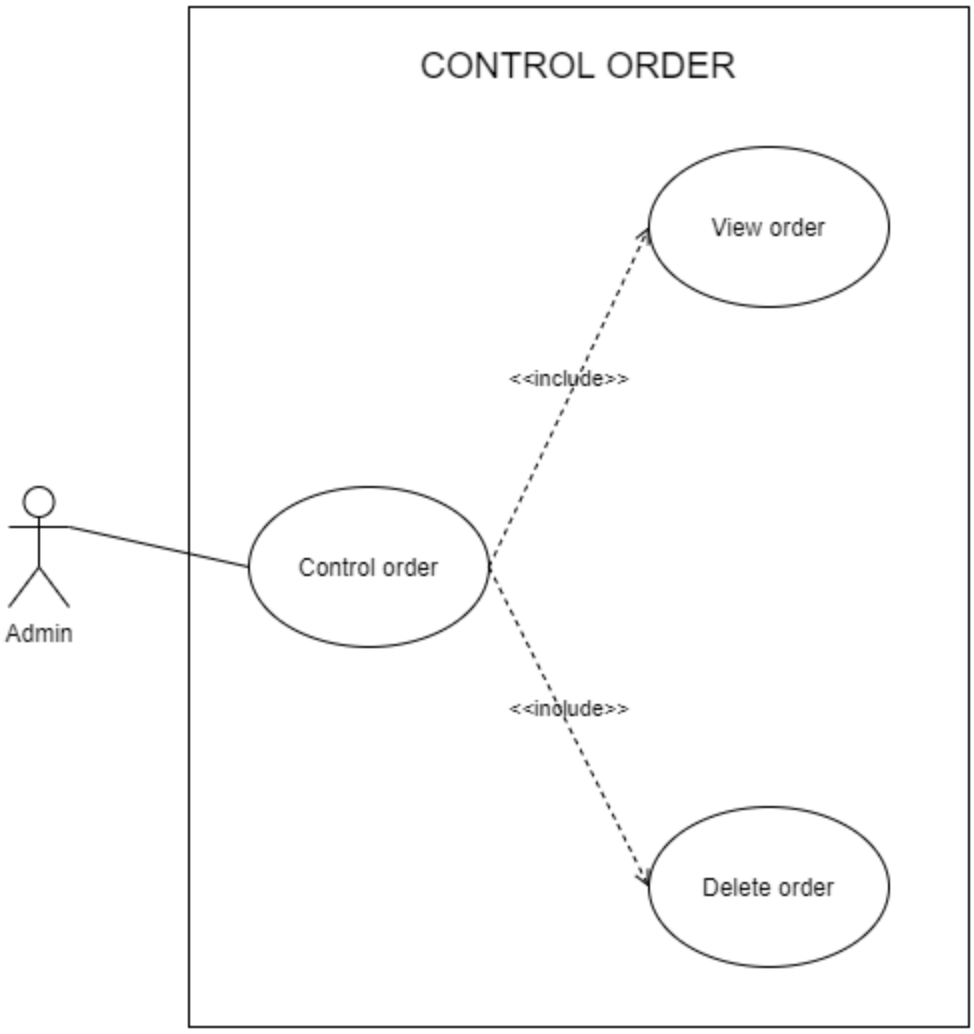
- Use case-control user



- Use case-control product

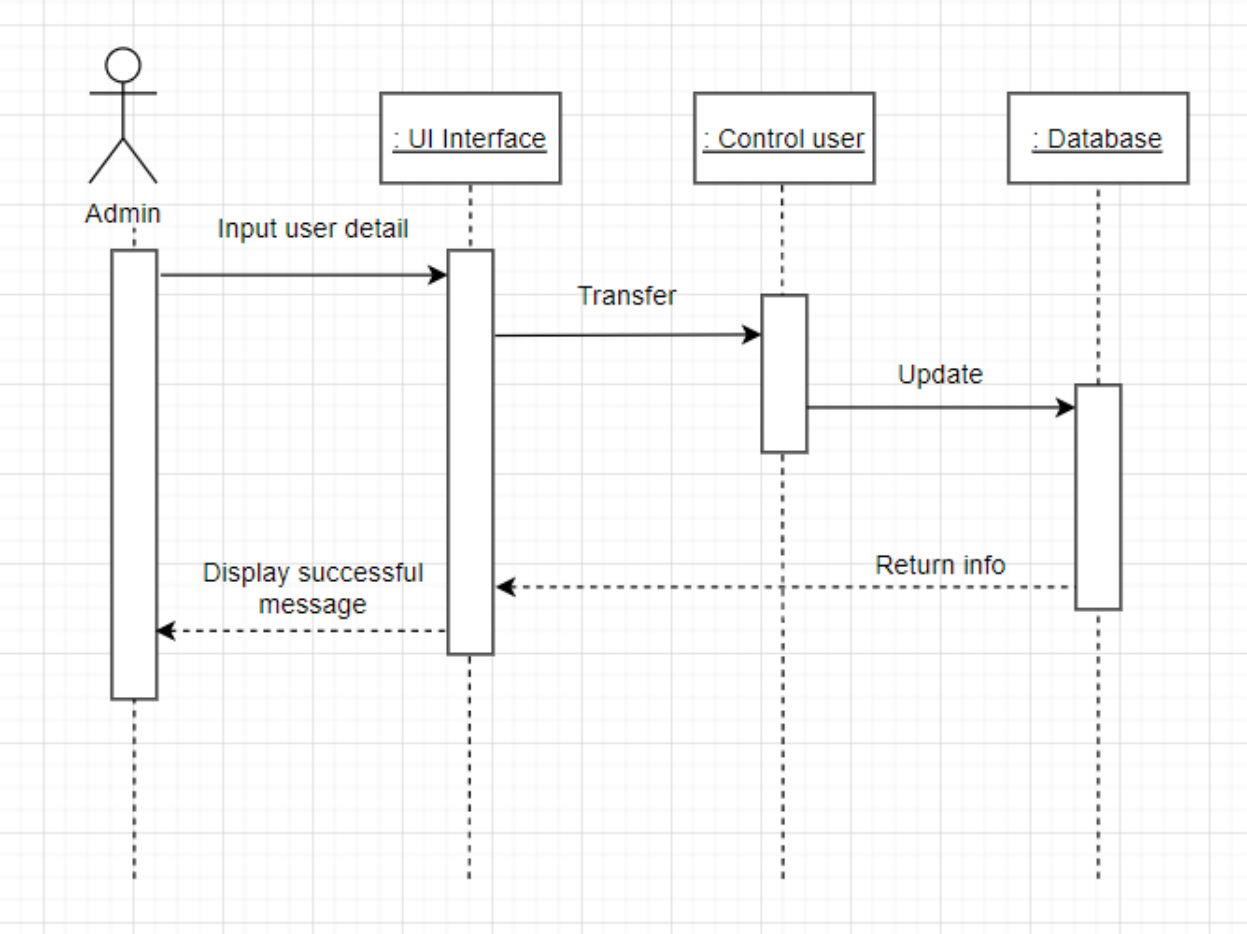


- Use case-control order or control cart

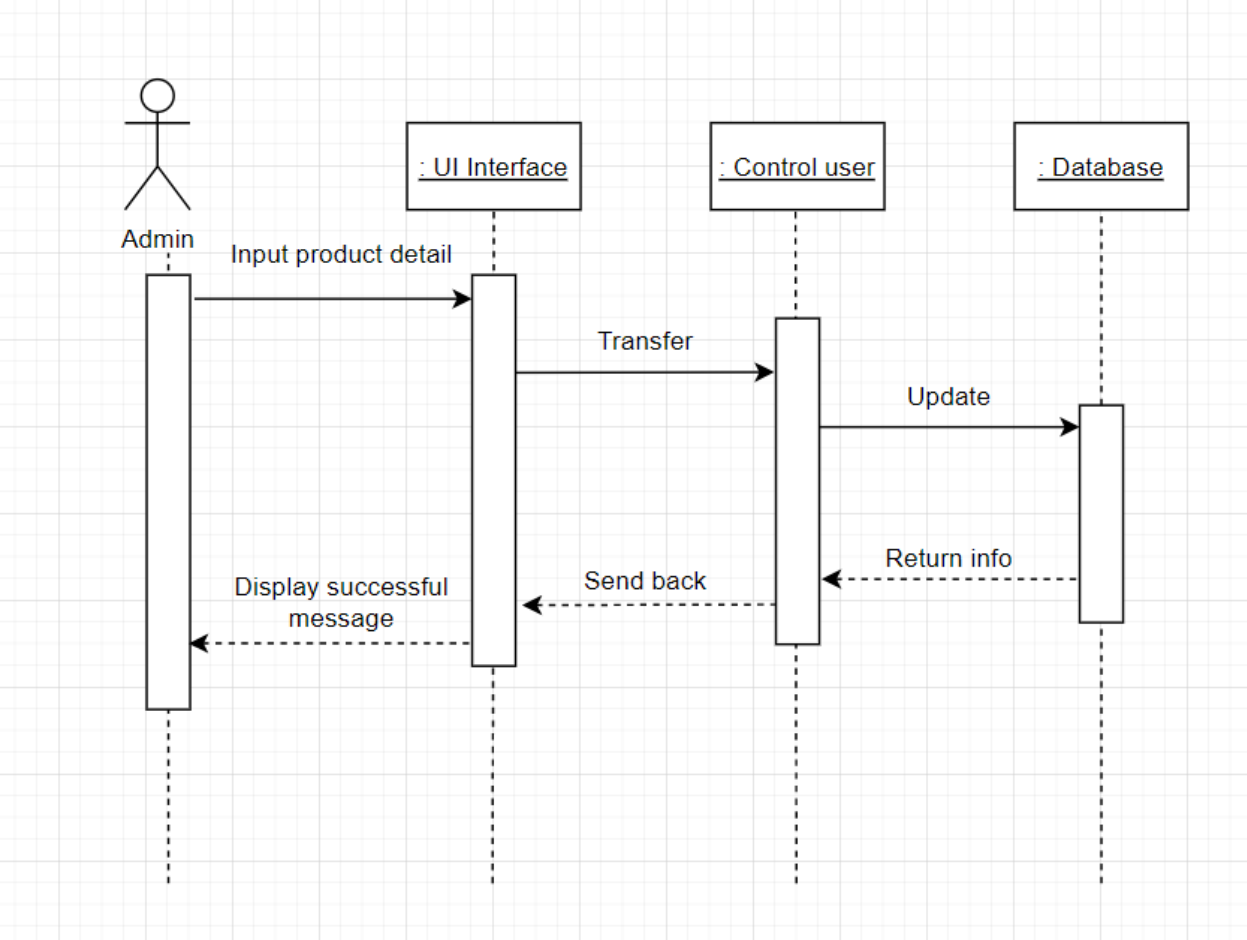


Sequence diagram

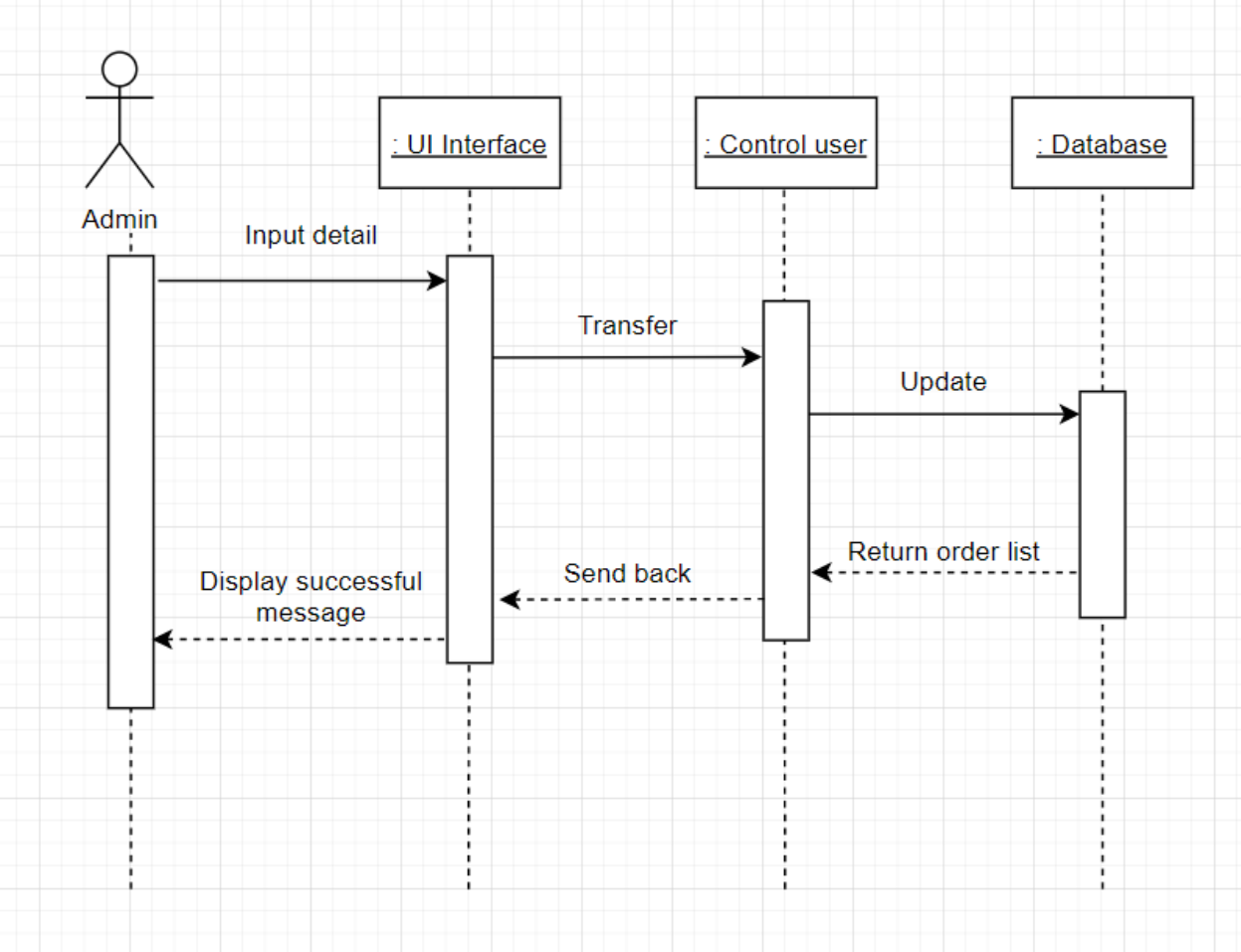
- Control user



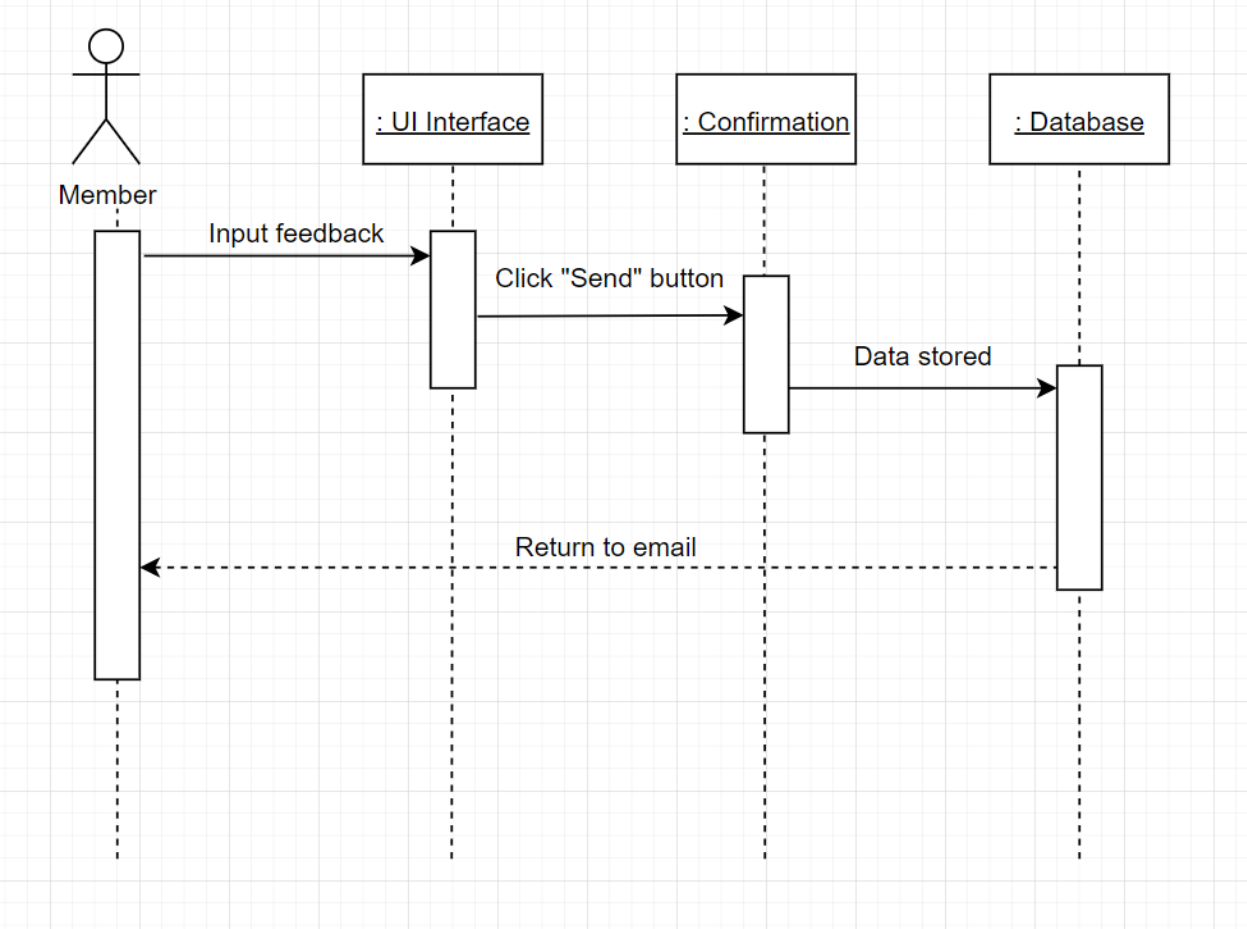
- Control product



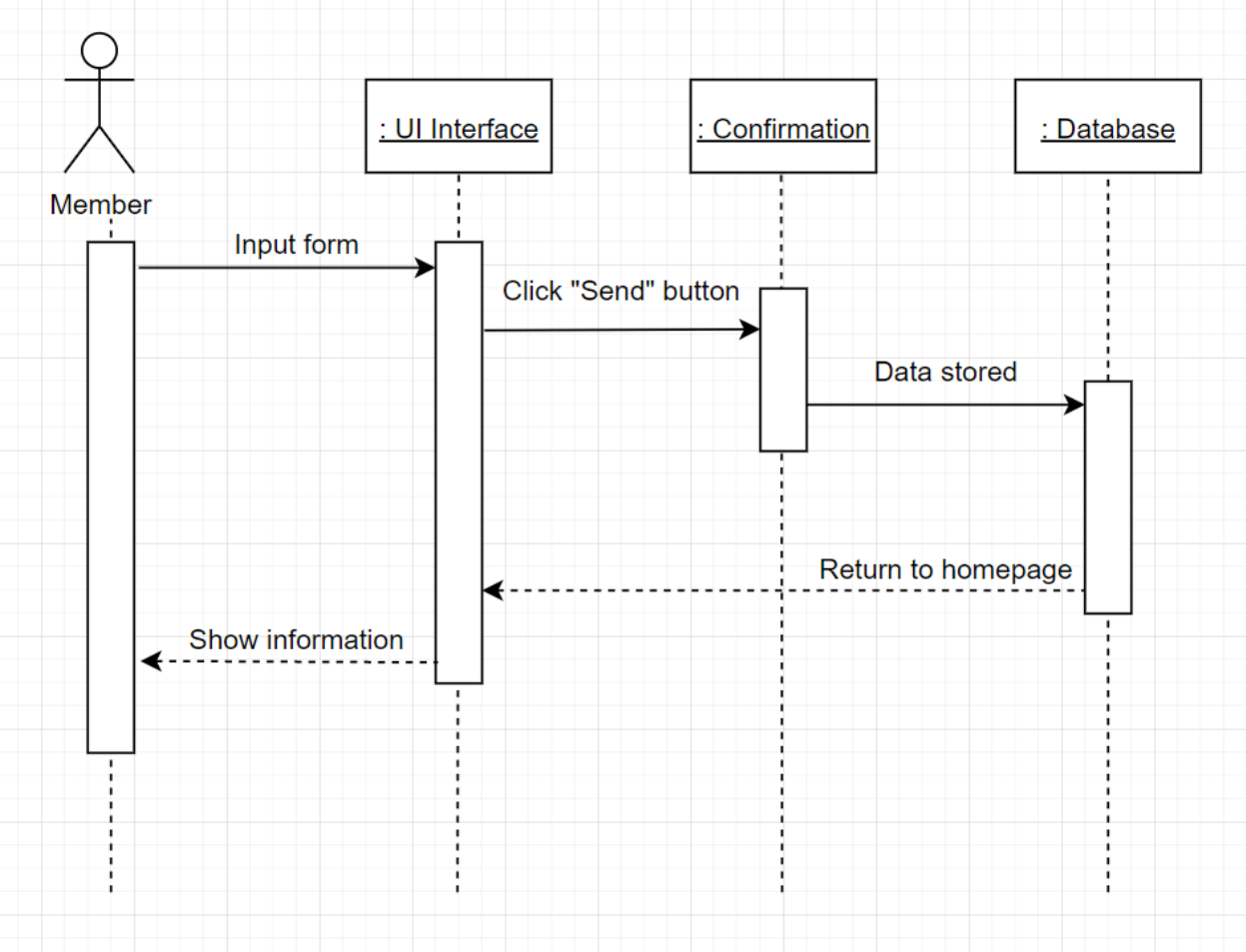
- Control cart



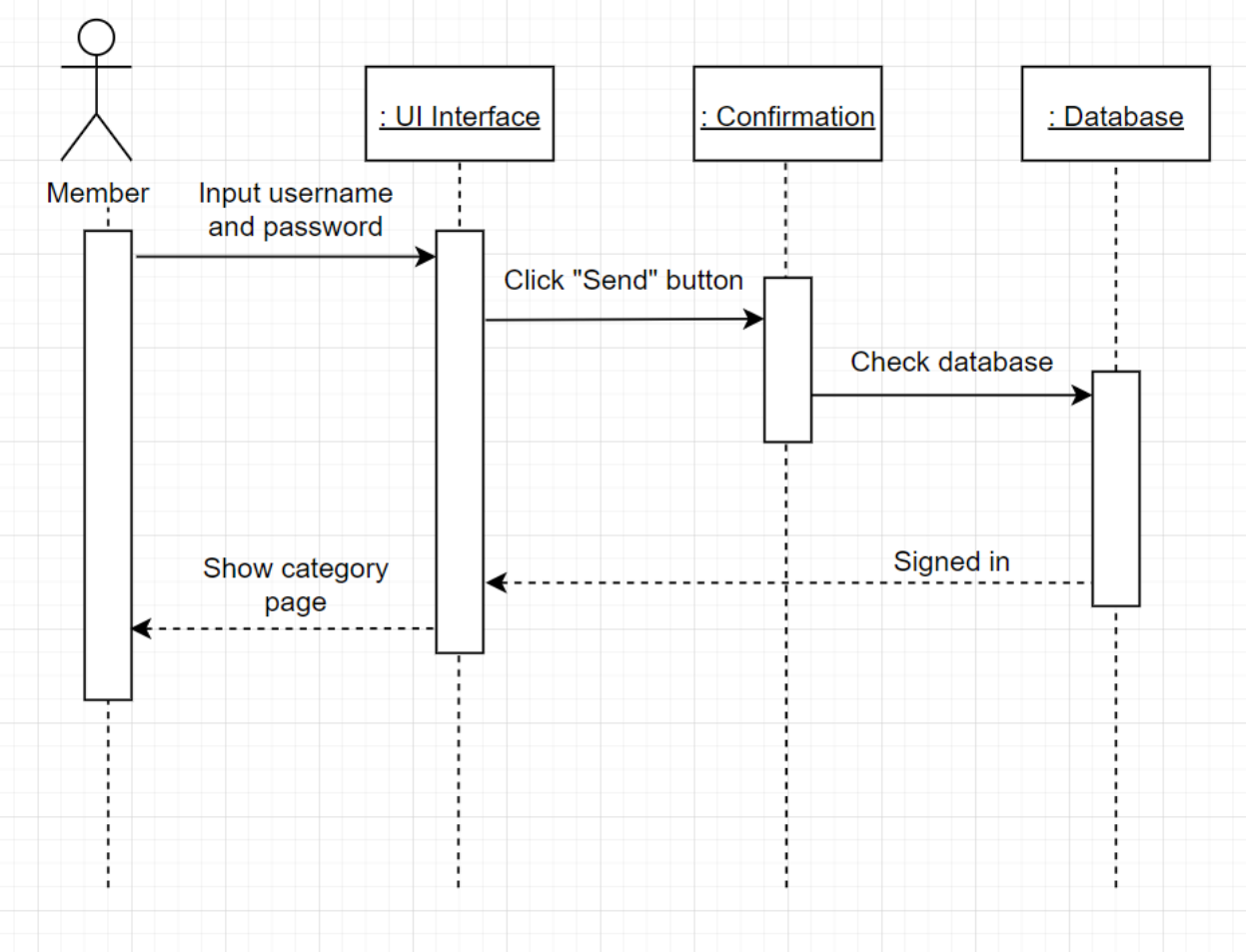
- Send feedback


Member
: UI Interface
: Confirmation
: Database
Input feedback
Click "Send" button
Data stored
Return to email

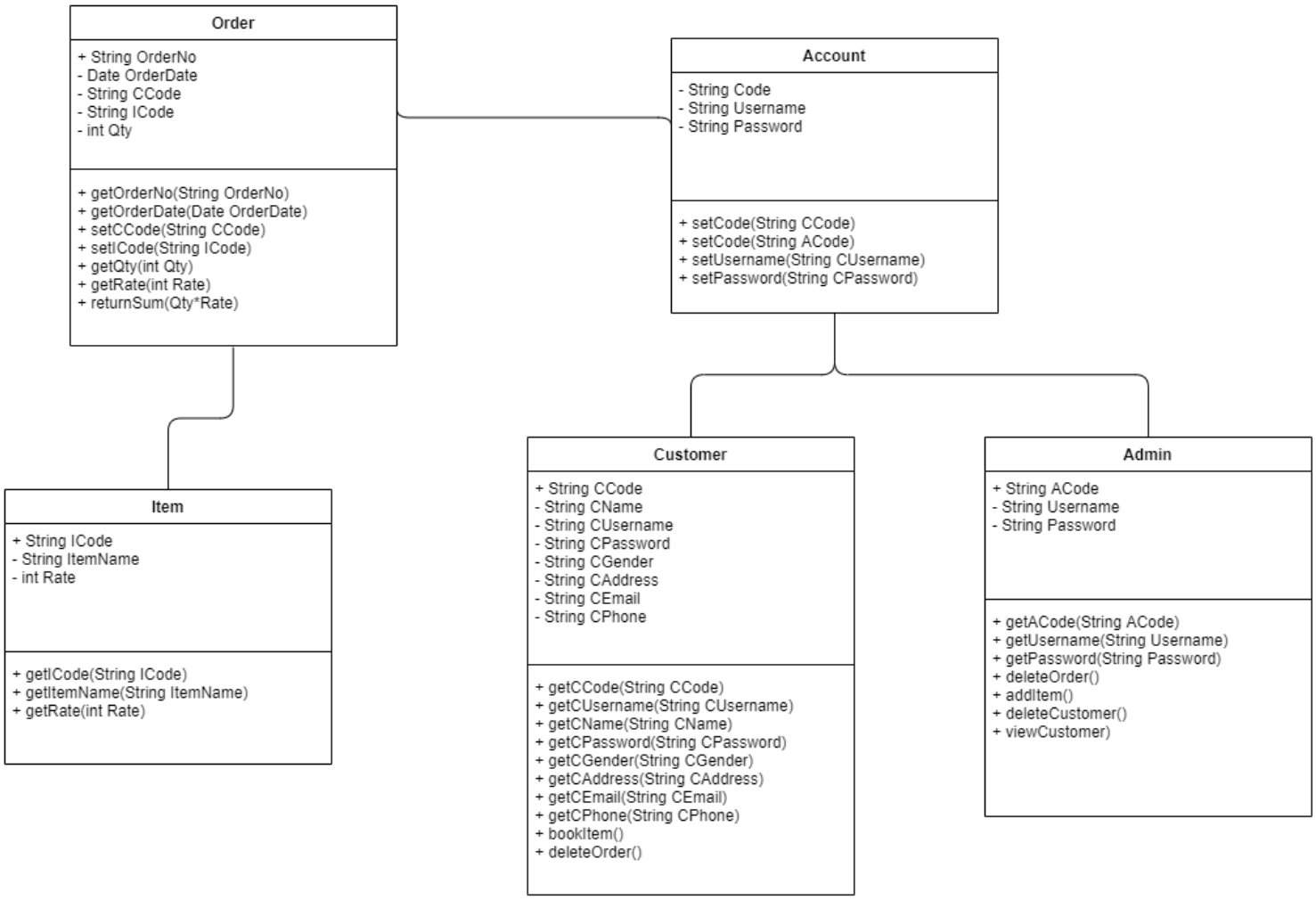
- Register


Member
: UI Interface
: Confirmation
: Database
Input form
Click "Send" button
Data stored
Return to homepage
Show information

- Log in



Class diagram

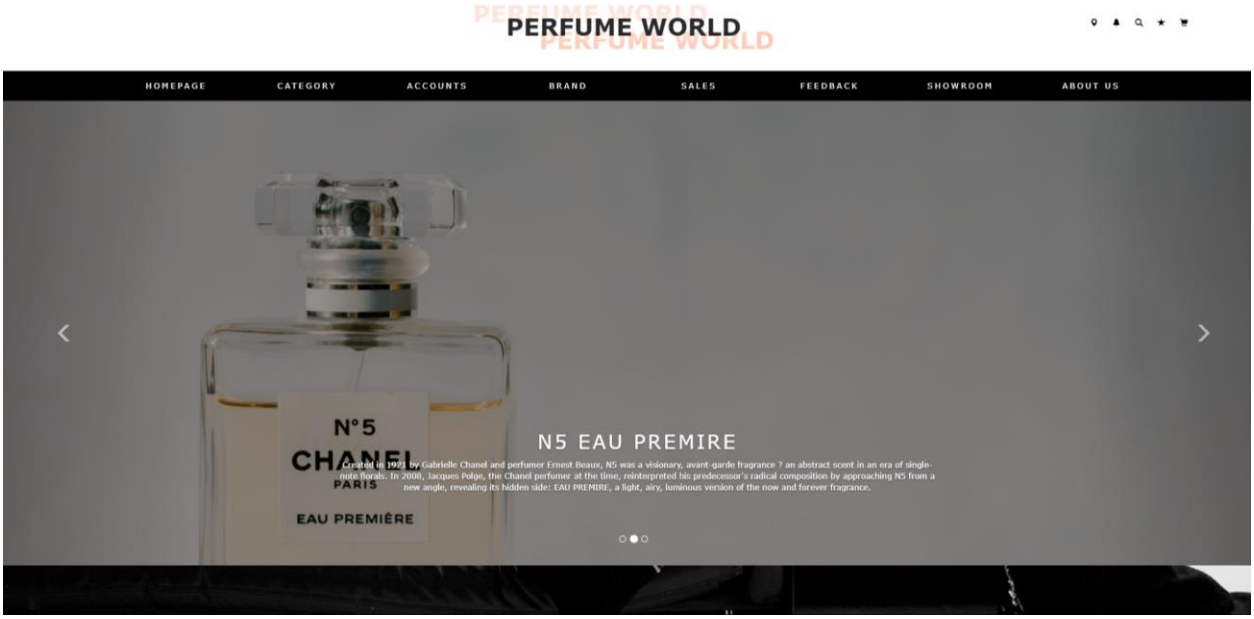

Order
+ String OrderNo
- Date OrderDate
- String CCode
- String ICode
- int Qty
+ getOrderNo(String OrderNo)
+ getOrderDate(Date OrderDate)
+ setCCode(String CCode)
+ setICode(String ICode)
+ getQty(int Qty)
+ getRate(int Rate)
+ returnSum(Qty*Rate)
Account
- String Code
- String Username
- String Password
+ setCode(String CCode)
+ setCode(String ACode)
+ setUsername(String CUsername)
+ setPassword(String CPassword)
Item
+ String ICode
- String ItemName
- int Rate
+ getICode(String ICode)
+ getItemName(String ItemName)
+ getRate(int Rate)
Customer
+ String CCode
- String CName
- String CUsername
- String CPassword
- String CGender
- String CAddress
- String CEmail
- String CPhone
+ getCCode(String CCode)
+ getCUsername(String CUsername)
+ getCName(String CName)
+ getCPassword(String CPassword)
+ getCGender(String CGender)
+ getCAddress(String CAddress)
+ getCEmail(String CEmail)
+ getCPhone(String CPhone)
+ bookItem()
+ deleteOrder()
Admin
+ String ACode
- String Username
- String Password
+ getACode(String ACode)
+ getUsername(String Username)
+ getPassword(String Password)
+ deleteOrder()
+ addItem()
+ deleteCustomer()
+ viewCustomer)

CHAPTER IV: SYSTEM IMPLEMENTATION

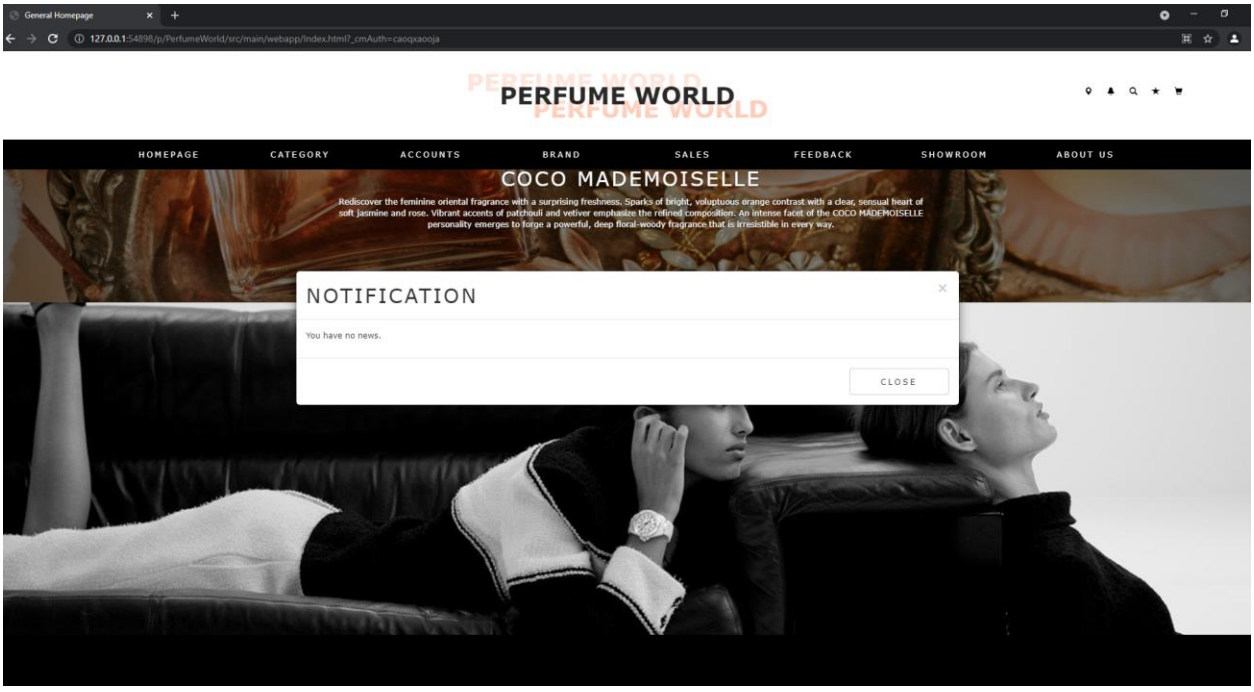
User manual

- General functions

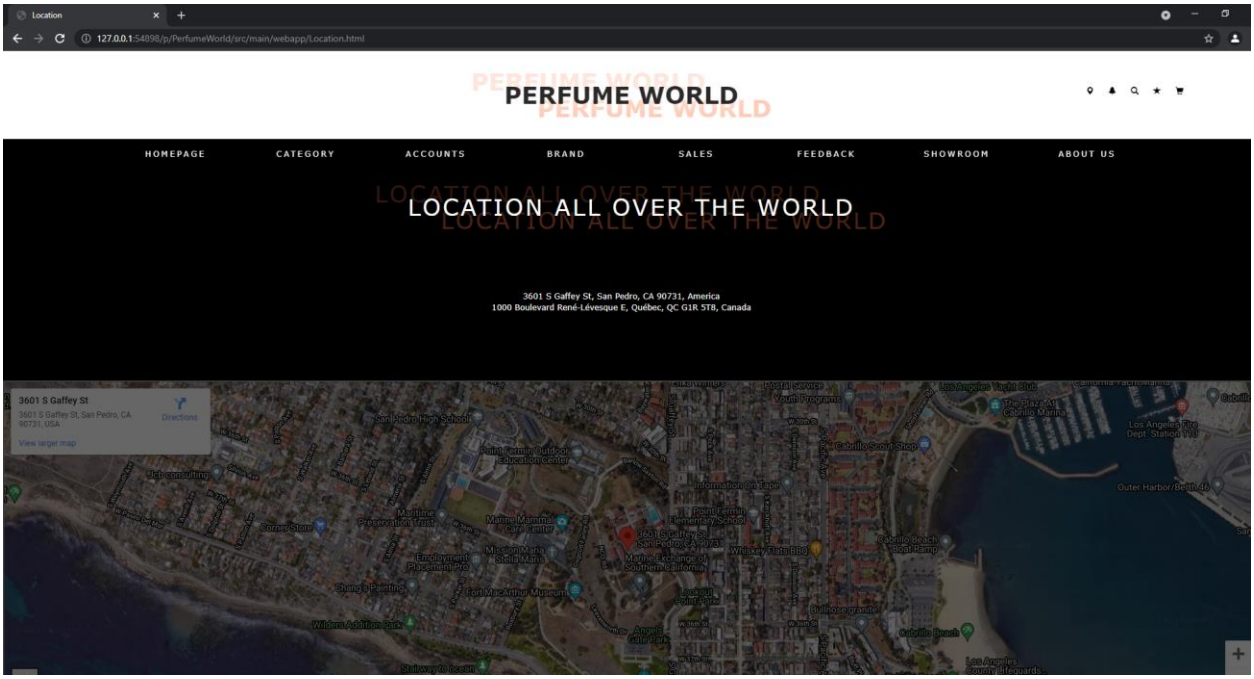
The homepage.



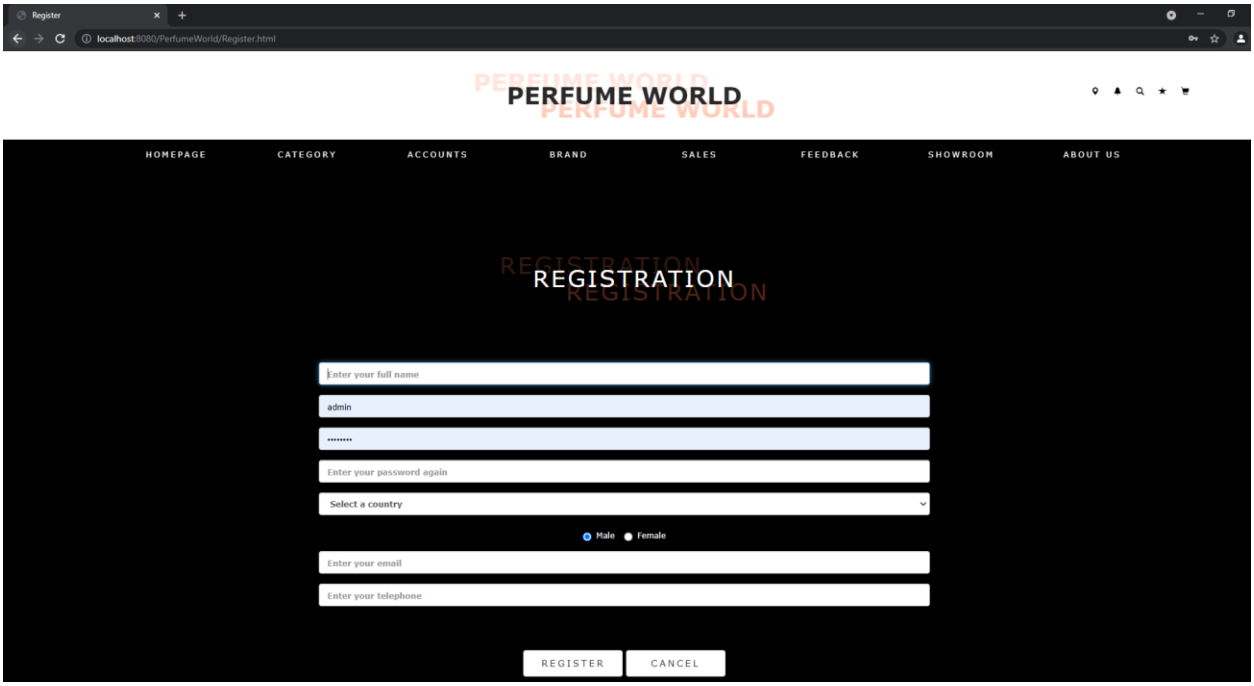
The notification.

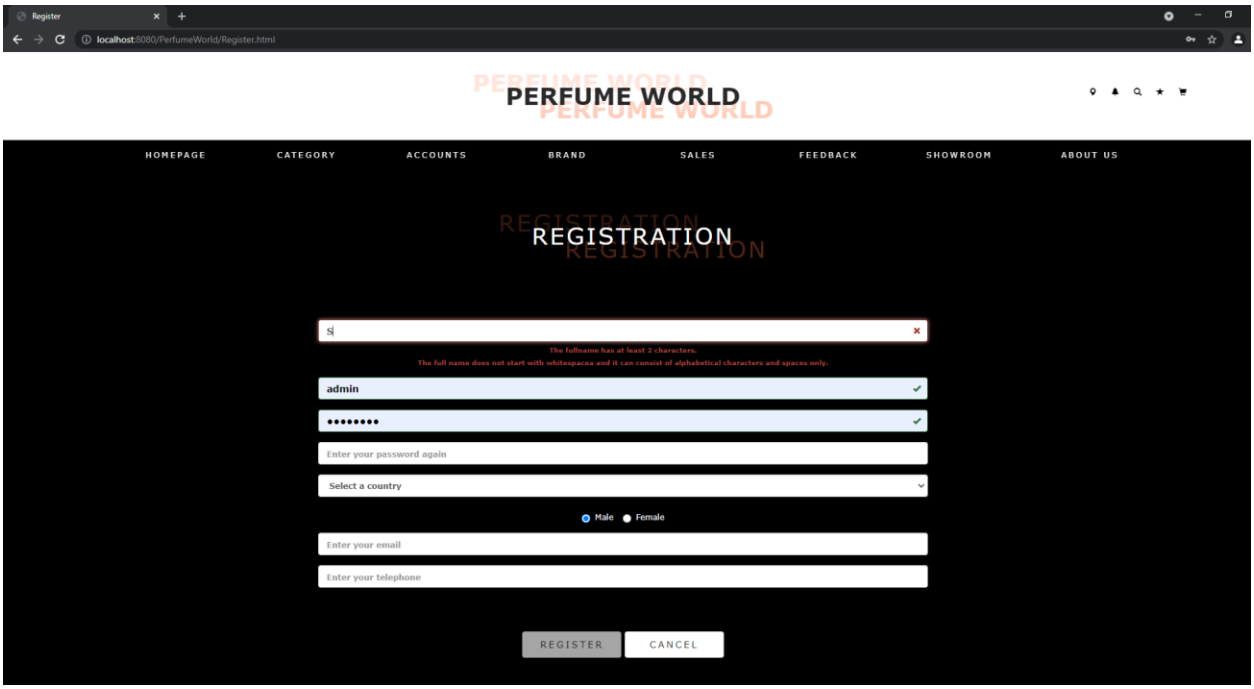


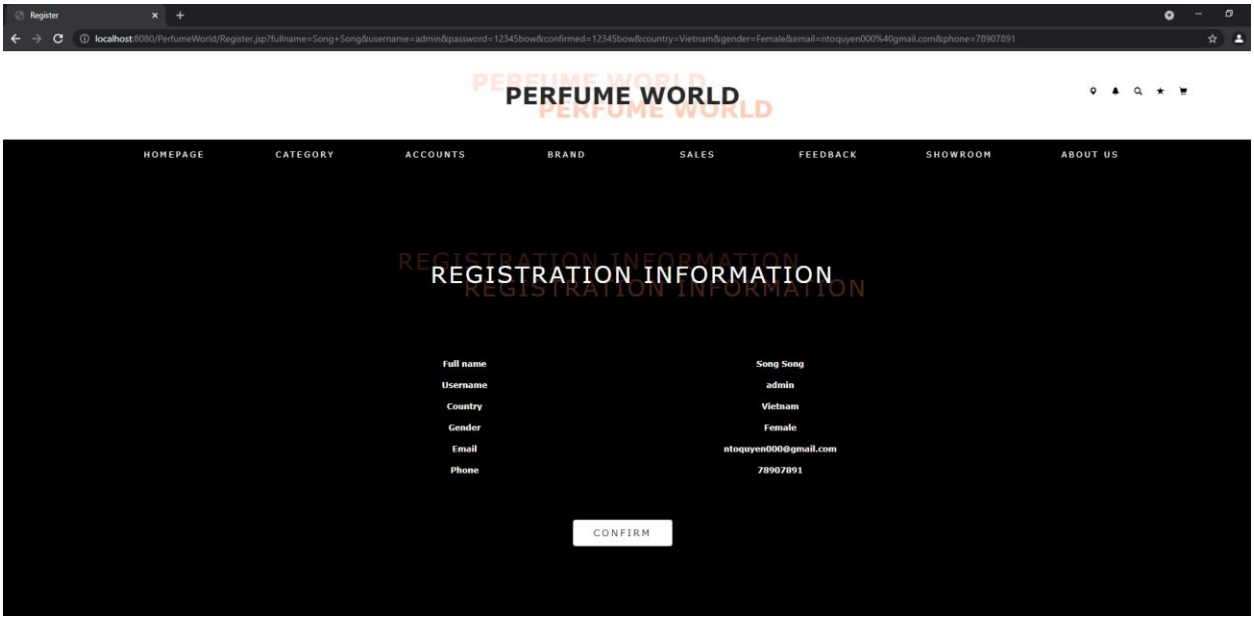
The locations of the stores are displayed on google maps. Click on each link to view each map.



The register with validation form. After registering, the user can see the registration description.

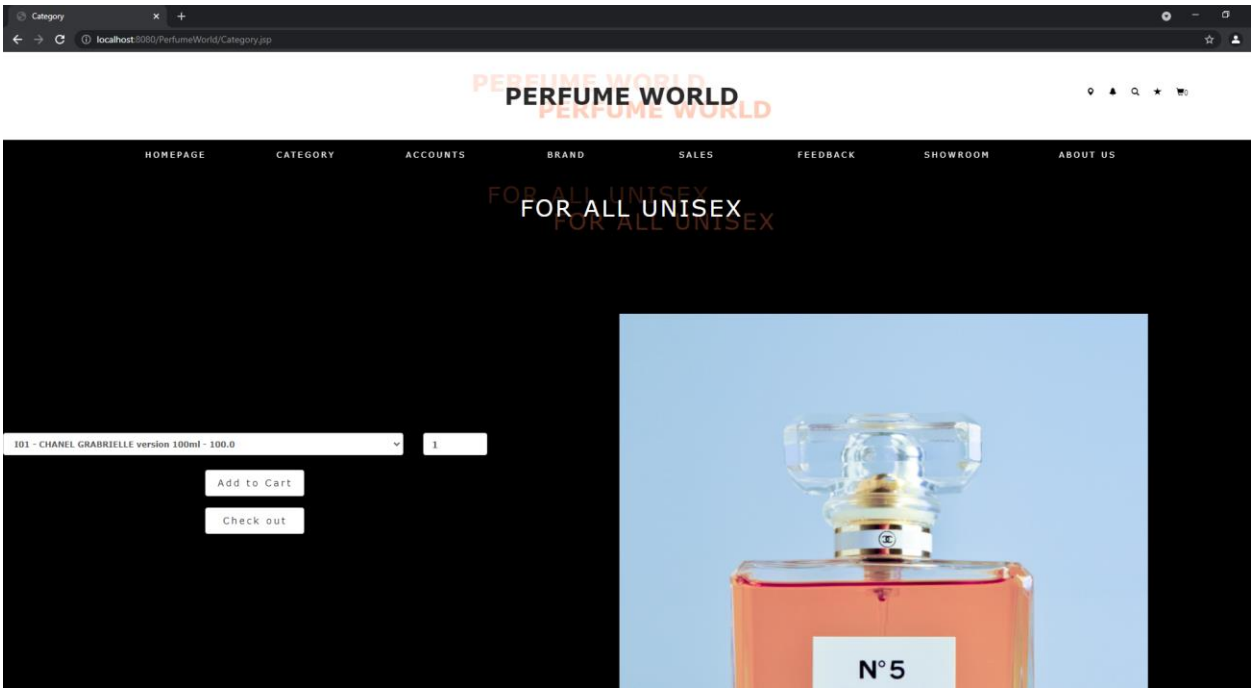




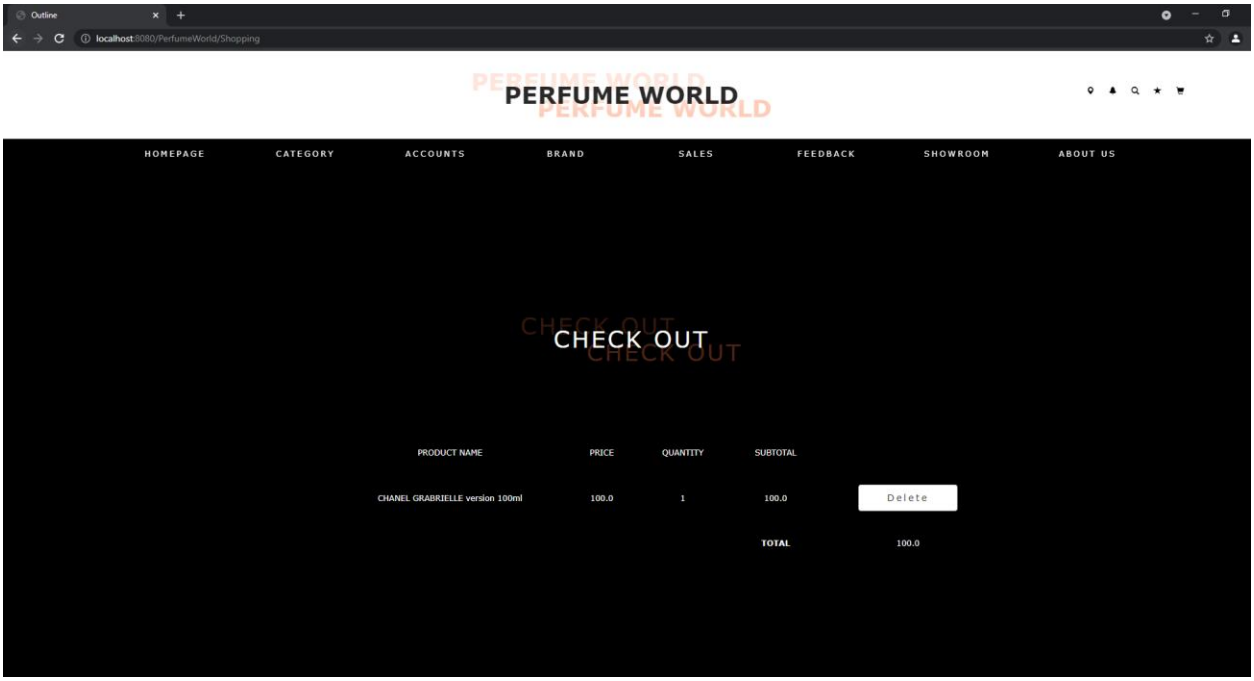


- User functions

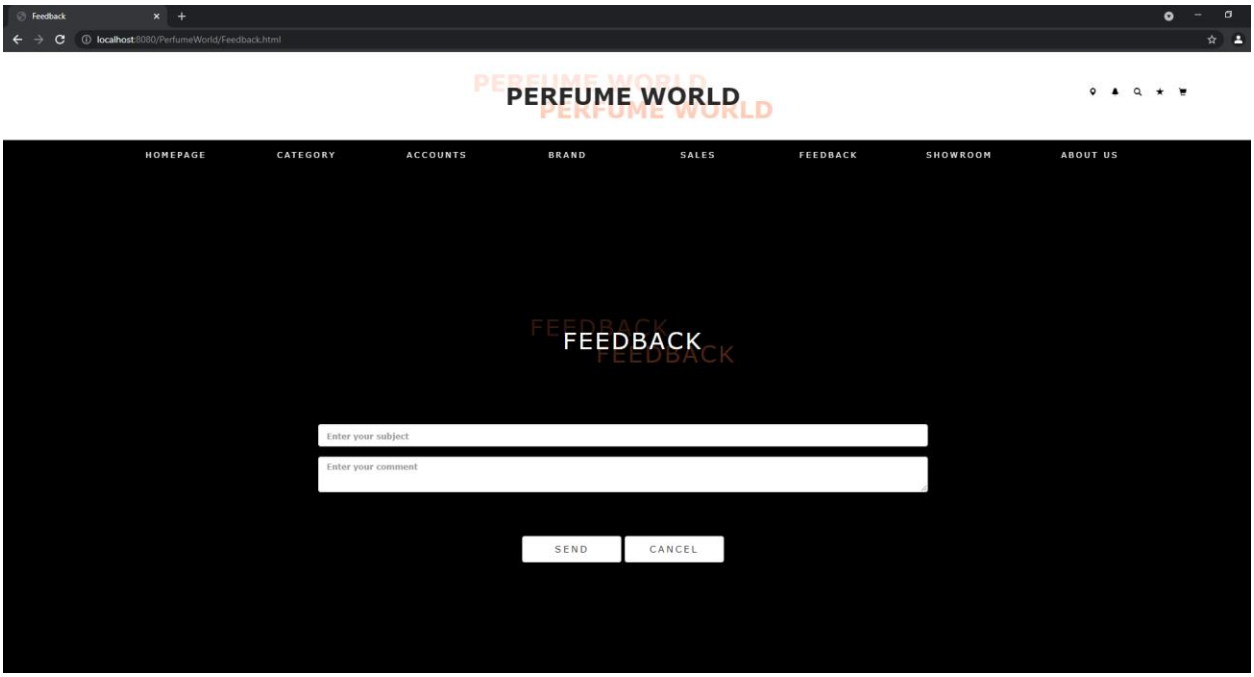
The product with the button to add or check out.



The check out with delete button to edit products that customers want.

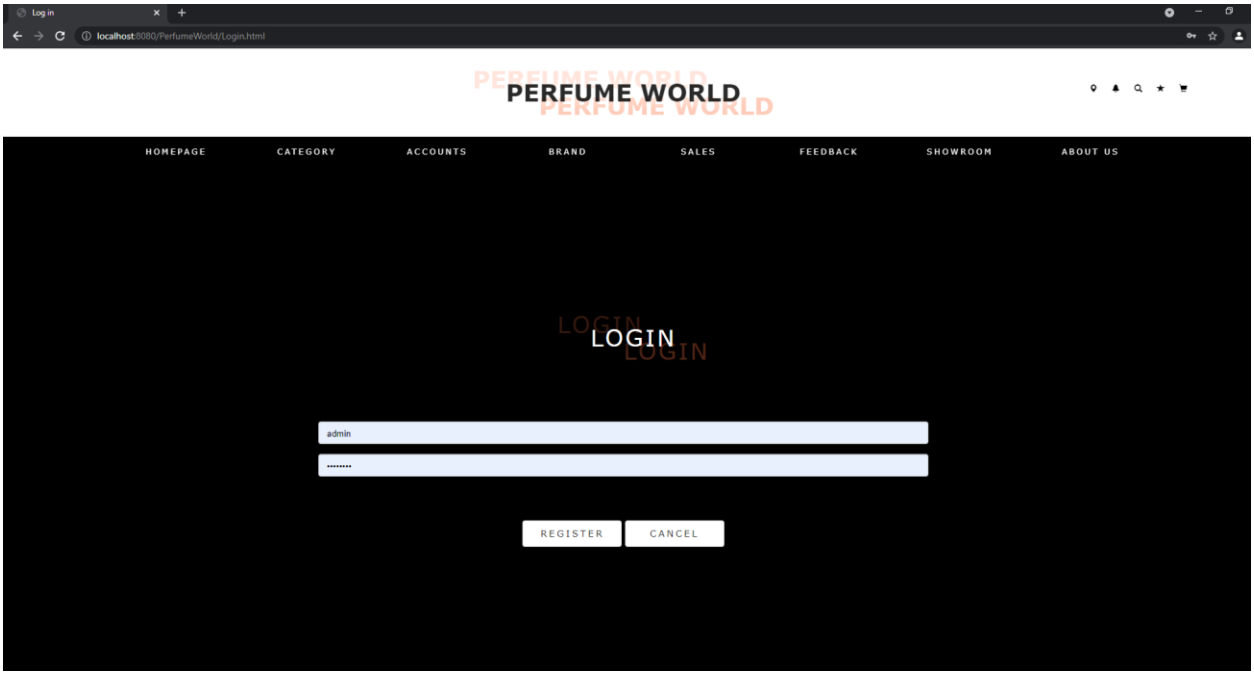


The feedback with subject and content input. If the users do not input data, then just click on the "Send" button to open the email.



- Admin functions

The account of admin is admin password 12345bow.



System source code

Log in Servlet

```
    public String username = "admin";
    public String password = "12345bow";

    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        response.setContentType("text/html");
        String reqUsername = request.getParameter("username");
        String reqPassword = request.getParameter("password");
        Cookie cookieUserName = null;
        Cookie cookiePassword = null;
        Cookie[] cookies = request.getCookies();
        if (cookies != null) {
            for (Cookie cookie : cookies) {
                if (cookie.getName().equals("username")) {
                    cookieUserName = cookie;
                }
                if (cookie.getName().equals("password")) {
                    cookiePassword = cookie;
                }
            }
        }

        if (cookieUserName == null || cookiePassword == null){
            if (reqUsername.equalsIgnoreCase(username) && reqPassword.equals(password)){
                Cookie cUsername = new Cookie("username", reqUsername);
                Cookie cPassword = new Cookie("password", reqPassword);
                response.addCookie(cUsername);
                response.addCookie(cPassword);
                RequestDispatcher dispatcher = getServletContext().getRequestDispatcher("/Category.jsp");
                dispatcher.forward(request, response);
            } else {
                RequestDispatcher dispatcher = getServletContext().getRequestDispatcher("/Login.html");
                dispatcher.forward(request, response);
            }
        } else if (cookieUserName.getValue().equalsIgnoreCase(username) && cookiePassword.getValue().equals(password)){
            RequestDispatcher dispatcher = getServletContext().getRequestDispatcher("/Category.jsp");
            dispatcher.forward(request, response);
        }
    }
```

Category Servlet

```
protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
    response.setContentType("text/html");
    //-------------------Handle Account Request----------------------
    if (this.acc == null){
        acc = new AccountBean();
        acc.setName(request.getParameter("username"));
        acc.setPassword(request.getParameter("password"));
    }
    System.out.println(acc);
    request.setAttribute("acc", acc);

    //-------------------Handle Item Request----------------------
    String itemStr = request.getParameter("item");
    String quantity = request.getParameter("quantity");

    if (itemStr != null && quantity != null){
        Item item = parseFromString(itemStr);
        Integer quantityInt = Integer.valueOf(quantity);

        inventory.put(item, inventory.getOrDefault(item, 0) + quantityInt);
    }
    request.setAttribute("inventory", inventory);

    //-------------------Handle Item Delete----------------------
    String deleteItemStr = request.getParameter("productDelete");
    System.out.println("ItemStrDelete: "+ deleteItemStr);
    Item deleteItem = null;
    if (deleteItemStr != null) {
        for (Item item : inventory.keySet()){
            if (item.getProduct().equals(deleteItemStr)){
                deleteItem = item;
                break;
            }
        }
        if (deleteItem != null){
            inventory.remove(deleteItem);
        }
    }
    System.out.println("Item Deleted: " + deleteItem);
    request.setAttribute("inventory", inventory);
```

Function JavaScript

```
function displaymap1() {
    var x1 = document.getElementById("map1");
    var x2 = document.getElementById("map2");
    if (x1.style.display === "block") {
        x1.style.display = "none";
    } else {
        x1.style.display = "block";
        x2.style.display = "none";
    }
}

function displaymap2() {
    var x1 = document.getElementById("map1");
    var x2 = document.getElementById("map2");
    if (x2.style.display === "block") {
        x2.style.display = "none";
    } else {
        x2.style.display = "block";
        x1.style.display = "none";
    }
}
```

```
function SendFeedback(){
  var subject = feedback.subject.value;
  var comment = feedback.comment.value;
  window.open('mailto:ntoquyen000@gmail.com?subject='+subject+'&body='+comment);
}
```

Format CSS

```css
/*HEADER LOGO*/
header {
    position: sticky; /*to stick the header in front of the interface while scrolling*/
    top: 0;
    width: 100%;
    z-index: 2000;
    margin-bottom: -10px; /*to eliminate the space between divs*/
    background-color: white;
}

#h1 {
    display: block;
    font-size: 3em;
    margin-top: 0.67em;
    margin-bottom: 0.67em;
    margin-left: 0;
    margin-right: 0;
    color: rgba(0,0,0, 0.84);
    font-family: Verdana, Geneva, Tahoma, sans-serif;
    text-shadow: 2px 2px 16px white, 50px 20px 0px rgba(255, 127, 80, 0.4), -50px -20px 0px rgba(255, 127, 80, 0.2);
    text-align: center;
    font-weight: 900;
}

header a:hover {
    text-decoration: none;
}

/*HEADER NAVIGATION BAR*/
#menu {
    position: initial; /*to set relationship between menu and submenu position*/
    letter-spacing: 3px;
}

#menu ul {
    list-style-type: none;
    text-align: center;
    background-color: black;
}
```

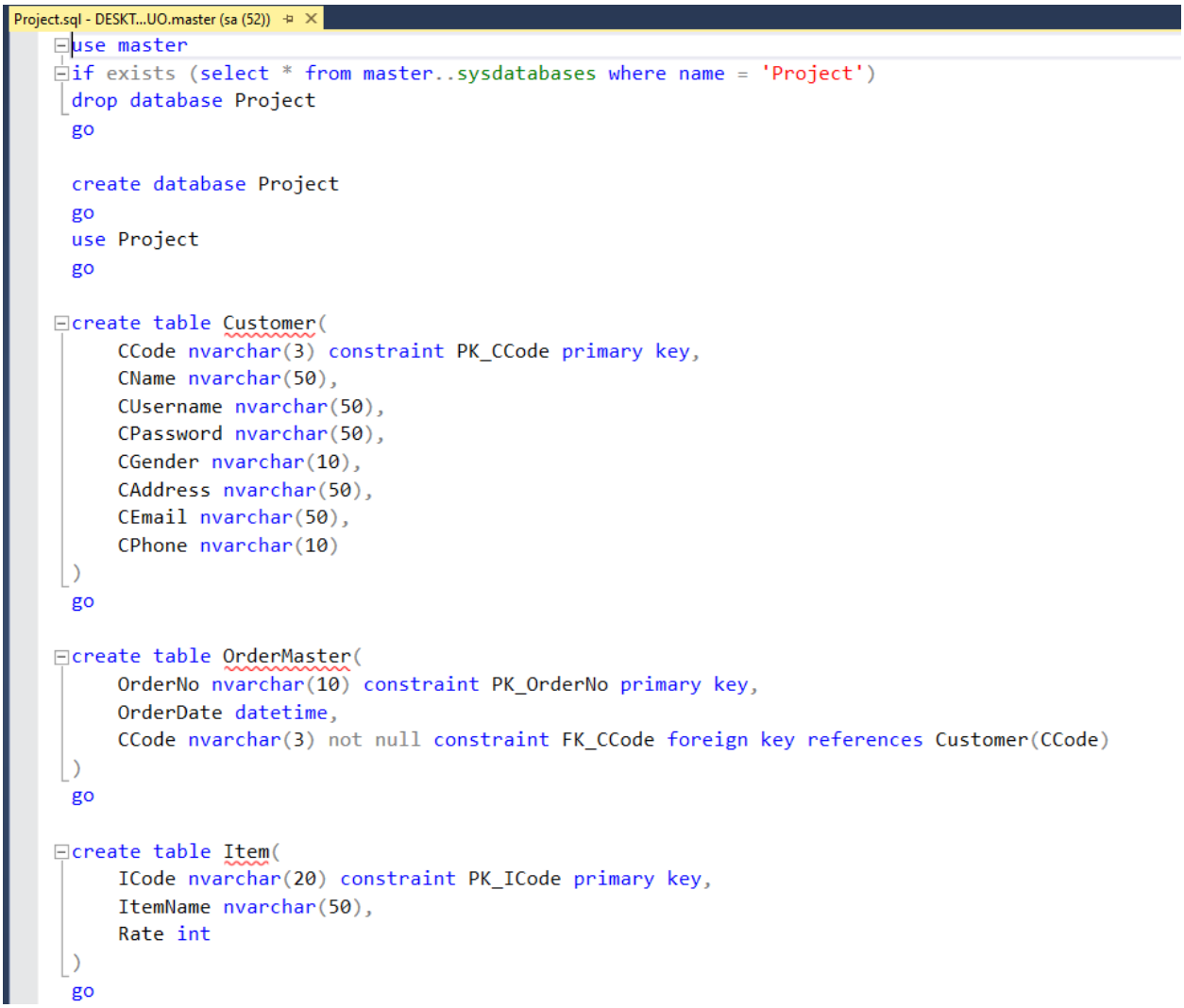
```
.glyphicon.glyphicon-menu-left, .glyphicon.glyphicon-menu-right {
    font-size: 30px; /*for carousel*/
    top: 335px;
}

.glyphicon.glyphicon-menu-left {
    left: -50px;
}

.glyphicon.glyphicon-menu-right {
    right: -50px;
}

/*BACKGROUND MATERIALS*/
.parallax {
    /* The image used */
    background-image: url('../webapp/image/background/bg.jpg');

    /* Set a specific height */
    min-height: 500px;

    /* Create the parallax scrolling effect */
    background-attachment: fixed;
    background-position: center;
    background-repeat: no-repeat;
    background-size: cover;
}

div {
    width:100%;
}

*{
    margin: 0;
    padding: 0;
    font-size: 12px;
    font-family: Verdana, Geneva, Tahoma, sans-serif;
}
```

Header and footer HTML

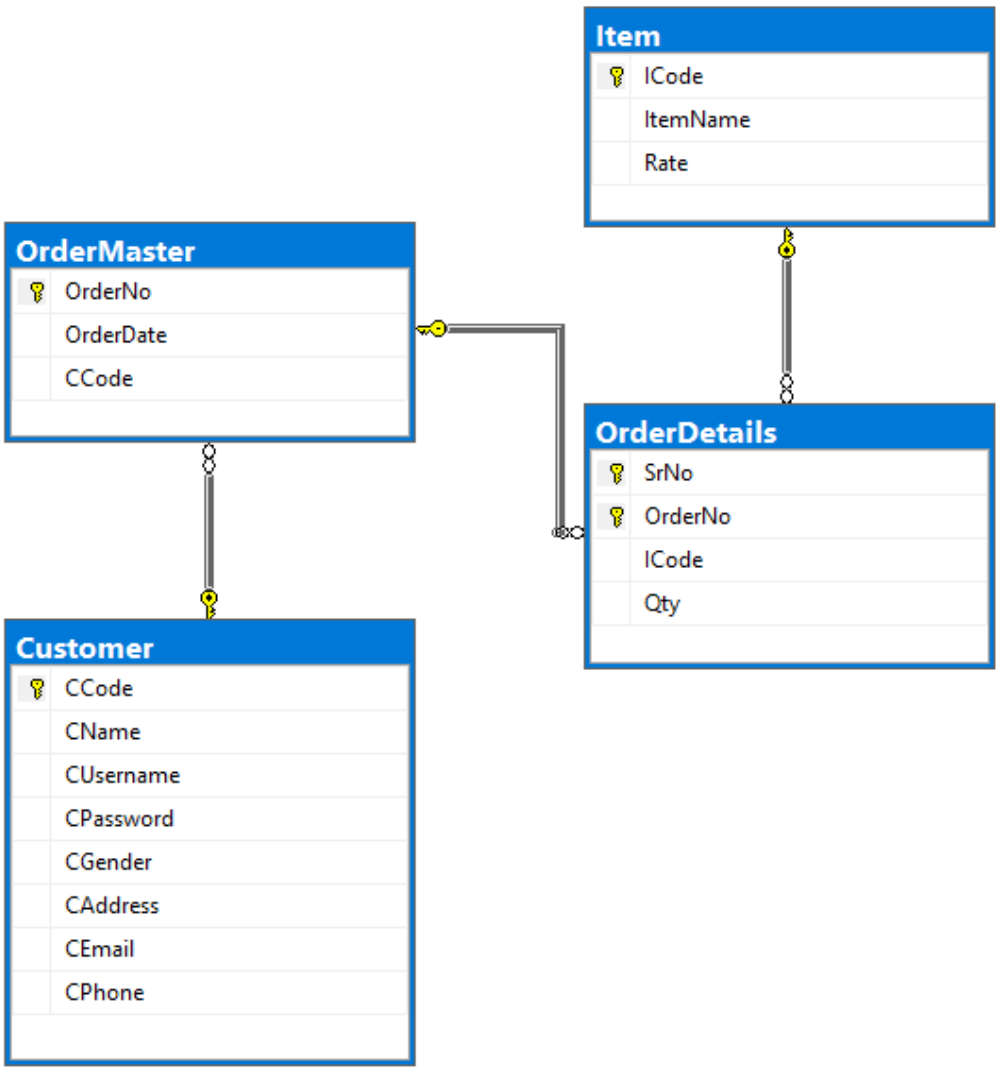
```
      <div class="col-sm-4">
        <a href="Index.html" id="h1">PERFUME WORLD</a>
        <br>
      </div>
      <!--Column 3-->
      <div class="col-sm-4">
        <ul id="icon-right" class="nav navbar-nav">
          <li><abbr title="Location"><a class="btn" href="Location.html"><span class="glyphicon glyphicon-map-marker"></span></a></abbr></li>
          <!--MODAL NOTIFICATION-->
          <li><abbr title="Notification"><a class="btn" data-toggle="modal" data-target="#modal-notification"><span class="glyphicon glyphicon-bell">
          <li><abbr title="Search"><a class="btn" href="Search.html"><span class="glyphicon glyphicon-search"></span></a></abbr></li>
          <li><abbr title="Favorite list"><a class="btn" href="Favorite.html"><span class="glyphicon glyphicon-star"></span></a></abbr></li>
          <li><abbr title="Shopping cart"><a class="btn" href="CheckOut.jsp"><span class="glyphicon glyphicon-shopping-cart"></span></a></abbr></li>
        </ul>
      </div>
    </div>
    <!--NAVIGATION BAR-->
    <div id="menu">
      <ul>
        <li><a class="active" href="Index.html">HOMEPAGE</a></li>
        <li><a href="Category.jsp">CATEGORY</a>
          <ul class="sub-menu">
            <li><a href="Unisex.html">ALL UNISEX</a></li>
            <li><a href="Women.html">ALL WOMEN</a></li>
            <li><a hrRegister.htmlen.html">ALL MEN</a></li>
            <li><a href="Products.html">ALL PRODUCTS</a></li>
          </ul>
        </li>
        <li><a href="Account.html">ACCOUNTS</a>
          <ul class="sub-menu">
            <li><a href="Register.html">REGISTER</a></li>
            <li><a href="Login.html">LOG IN</a></li>
          </ul>
```

```
<!--FOOTER-->
<footer>
  <div class="site-footer">
    <div class="row">
      <!--Column 1-->
      <div class="col-sm-6">
        <h6>OUR COMPANY</h6>
        <p class="text-justify">We are Perfume World from Song Song company. Here is the place where you can find your scent. Each of them is unique and cha
        <br><br>
        <div class="row"> <!-- for logo-->
          <div class="col-sm-4">
            <a href="https://www.chanel.com/us/"><img src="../webapp/image/footer/logo1.JPG" id="footer-logo"></a>
          </div>
          <div class="col-sm-4">
            <a href="https://www.gucci.com/int/en/"><img src="../webapp/image/footer/logo2.JPG" id="footer-logo"></a>
          </div>
          <div class="col-sm-4">
            <a href="https://www.dior.com/en_int"><img src="../webapp/image/footer/logo3.JPG" id="footer-logo"></a>
          </div>
        </div>
      </div>
      <!--Column 2-->
      <div class="col-sm-2">
      </div>
      <!--Column 3-->
      <div class="col-sm-2">
        <h6>DISCOVERY</h6>
        <ul class="footer-links">
          <li><a href="Index.html">Homepage</a></li>
          <li><a href="Category.jsp">Category</a></li>
          <li><a href="Brand.html">Brand</a></li>
          <li><a href="Sales.html">Sales</a></li>
          <li><a href="Login.html">Login</a></li>
          <li><a href="Register.html">Register</a></li>
          <li><a href="Feedback.html">Feedback</a></li>
          <li><a href="Showroom.html">Showroom</a></li>
        </ul>
      </div>
```

Database SQL



Relational database diagram



Connect database Java

```java

@WebServlet("/Register")
public class Register extends HttpServlet {
    private static final long serialVersionUID = 1L;

    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        String url = "jdbc:sqlserver://DESKTOP-9L996I3\\SQLFULL:1433;databaseName=fragrance;";
        String USER = "sa";
        String PASS = "cuong123";
        String fname = request.getParameter("fullname");
        String password = request.getParameter("password");
        String phone = request.getParameter("phone");
        String email = request.getParameter("email");
        String address = request.getParameter("country");


        try {
            Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
            Connection conn = DriverManager.getConnection(url,USER,PASS);
            Statement stmt = conn.createStatement();
            String sql;
            sql = "INSERT INTO customer VALUES(?,?,?,?,null,null)";
            PreparedStatement preparedStmt = conn.prepareStatement(sql);
            preparedStmt.setString(1,email);
            preparedStmt.setString(2,password);
            preparedStmt.setString(3,address);
            preparedStmt.setString(4,phone);
            preparedStmt.execute();
            conn.close();
        } catch (ClassNotFoundException ex) {} catch (SQLException ex) {}
        RequestDispatcher dispatcher = request.getRequestDispatcher("Register.jsp");
        dispatcher.forward(request, response);
    }
```

CHAPTER V: CONCLUSION AND DISCUSSION

1. List of accomplished work

We solved the problem of online shopping by creating a system that meets the needs of customers, such as low prices, quick delivery, high quality, and ease of use. Anyone may use this system.

The project's goal and objective have been met. When a consumer purchases a product on our website, we provide for them. We completed the project on time.

2. Strength and weakness

Strength

- We designed practically all of the functionalities on our website.
- It features a user interface that is simple enough for a novice user to understand.
- In the SQL server, we have a well-managed database.

Weakness

- There is no way to pay for actual work.
- The search tool is not fully functional.
- Despite having a well-managed database, we were unable to link it to the system.

3. Future work

Although there are many aspects of this project that can be improved in the future, it is still a good project for kids to work on.

To finish the project, we will endeavor to optimize the source code for better efficiency and improve the backend. Furthermore, we will host this website on a global server.

4. References

https://docs.microsoft.com/en-us/sql/connect/jdbc/microsoft-jdbc-driver-for-sql-server-support-matrix?view=sql-server-ver15

https://xuanthulab.net/bieu-thuc-chinh-quy-regexp.html

https://thinhnotes.com/chuyen-nghe-ba/use-case-diagram-va-5-sai-lam-thuong-gap/

https://stackjava.com/jsp-servlet/cac-scope-trong-servlet-application-request-session-va-page-scope.html

https://www.tutorialspoint.com/jsp/jsp_standard_tag_library.htm

https://www.dreamweaver-templates.org/web-template-1a.htm

https://www.lightology.com

https://www.led-linear.com

https://iviettech.vn/blog/1358-thuc-hanh-xay-dung-sequence-diagarm.html

https://www.w3schools.com/bootstrap/bootstrap_templates.asp