

**Computer Science & Engineering Department**

**INTERNATIONAL UNIVERSITY**

**Ho Chi Minh City, Vietnam**

***Software Engineering***

VIRTUAL

SHOPPING

SYSTEM

***REPORT***

**Instructor: Mrs. Loan Nguyen**

**Topic: Virtual Shopping System**

**Semester I - Academic year: 2019-2020**

**INDEX**

|  |  |
| --- | --- |
| Title | Page |
| *Index* | 2- 3 |
| Members and Roles | 4- 5 |
| Introduction | 7- 18 |
| Functional Requirements | 19- 21 |
| Non- Functional Requirements | 22 |
| Usage Scenarios | 23- 38 |
| Implementation | 39- 44 |

**A. MEMBERS & ROLES**

|  |  |  |
| --- | --- | --- |
| NAME | ID | ROLE |
| DIỆP PHƯƠNG QUỲNH | ITITIU17041 | * Programming FRONT- END and BACK- END * Drawing 4 ACTIVITY DIAGRAM. * Drawing CLASS DIAGRAM * Describing 3 use cases * Writing report |
| NGUYỄN HỒNG THẢO | ITITIU17074 | * Programming FRONT- END * Drawing 4 ACTIVITY DIAGRAM * Drawing 6 SEQUENCE DIAGRAMs * Describing 3 use cases * Writing report |
| HỒ ĐẶNG PHƯƠNG NGỌC | ITITIU17058 | * Programming FRONT- END * Drawing ERD and creating realtional schema. * Drawing USE CASE + describing 3 use cases * Drawing 8 SEQUENCE DIAGRAMs * Writing report. |

**B. PROJECT FINAL DOCUMENT**

1. **INTRODUCTION**
   1. ***VISION***

In the present time, demand for shopping is increasingly high. However, heavy workflow and busy schedules prevent everyone from direct visits to stores to own their favorite outfits or anything that is in need. Therefore, online shopping system is developed to tackle this problem on time.

Online shopping is a process of customers or consumers buying goods or services directly from a seller over the Internet without any intermediary service. Customers by now can own their needs by comforting themselves at home or even at work, accessing an online shopping website, choose their favorites and the goods may soon be delivered to provided addresses. Until the current time, no many efforts have been taken or any complaints about time consuming process have been reported from this shopping method.

This shopping method brings on convenience and time saving which takes as few efforts as possible. Shopping has never been easier with Wi-Fi or Internet connection, an electricity device and a targeted website, and buyers are able to obtain their on- demand things.

In particular, our website is developed to meet that common request. This website has its signature as a combination of many brands and stores providing skin care products. Categories vary from makeup removal, toner, etc. to moisturizer, or eye- cream.

* 1. ***THE PROBLEM DOMAIN***

***A screenshot of a cell phone

Description automatically generatedPRODUCT BACKLOG and USER STORIES***

* 1. ***STAKEHOLDERS***

1. ***Client/ Customer:***

Our CLIENT/ CUSTOMER is the same, which is the company asking for the website for advertising their reputation as well as having their product sold online.

1. ***User:***

***A close up of a map

Description automatically generated***We provide the website for two main target users which are: Customer and Manager.

Managers have the permission to access the website mainly for managing website including the information of products, customers and last but not least are orders. The website itself has some specific functions as below:

* Sign in/ Sign up
* Create a product
* View a product
* Delete a product
* Update a product
* View customer’ s information

Customers are able to access the website for online shopping, which means that they can perform any functions on demand in the list below:

* Sign in/ sign up
* View a product
* Add a product to cart
* Update cart
* Check out
  1. ***SOFTWARE CONTEXT***

1. **ENTITY DIAGRAM**

***A close up of text on a white background

Description automatically generated***

First, we consider the DATABASE required for the website:

We have four entities to consider which are MANAGER, CUSTOMER, PRODUCT and ORDER and four relationships respectively which are MANAGES (manager- product), BUYS (customer- order), CONFIRMS (manager- order) and comprises (order- product)

* For each entity:
* Manager consists of: ManagerId (primary key), Name, Username, Password, Address, Contact, Email
* Customer consists of: CustomerId (primary key), Name, Usernaem, Password, Address, Contact, Email
* Product consists of: ProductId (primary key) , Name, Description, Origin, Category, Brand, Price, Discount, In- stock
* Order consists of: OrderId (primary key) ,Quantity, Date, Total, Tax, Discount, ShippingFee, ShippingMethod, CardInfo
* For each relationship:
* Manages: Manager has a relationship (1,n) to (1,n) with Product, which means that one manager can manages 1 to many product and one product can be managed by 1 or many Manager(s)
* Buys: Customer has a relationship (1,1) to (1,n) with Order, which means that one customer can buy 1 to many Order(s) and one order belongs to just one customer
* Confirms: Manager has a relationship (1,n) to (1,n) with Order, which means that one manager can confirm for 1 to many Order(s) and one Order can be confirmed by one to many Manager(s)
* Comprises: Order has a relationship (1,n) to (1,n) with Product, which means that one order comprises 1 to many Product and one Product(s) belongs to 1 to many Order(s)

1. **RELATIONAL DATABASE SCHEMA**

***A close up of text on a white background

Description automatically generated***

From the above entity diagram, we easily attain the Relational Database Schema

**Customer- Schema=** (**CusId**,CusName,CusUserName, CusPassword, CusAddress, CusEmail, CusContact)

**Manager- Schema=** (**ManaId**, ManaName, ManaUserName, ManaPassword, ManaAddress, ManaEmail, ManaContact)

**Product- Schema=** (**ProdId**, ProdName,ProdOrigin, ProdCategory, ProdInstock, ProdDescription, ProdPrice, ProdBrand, ProdDiscount)

**Order- Schema=** (**OrderId**, CustomerId, OrderQuamtity, OrderDate, OrderTotal, OrderTax, OrderDiscount, OrderShippingMethod, OrderShppingFee, OrderTotalPay, ORderCardInformation)

***From Customer- Schema. CusId to Order- Schema. CusotmerId***

**Manages- Schema=** (**ManagerId, ProductId**)

***From Manager- Schema. ManaId to Manages- Schema. ManagerId***

***From Product- Schema. ProdId to Manages- Schema. ProductId***

**Confirms- Schema= (ManagerId, OrderId**)

***From Manager- Schema. ManaId to Confirms- Schema. ManagerId***

***From Order- Schema. OrderId to Confirms- Schema. OrderId***

**Comprises- Schema=** (**OrderId, ProductId**)

***From Order- Schema. OrderId to Comprises- Schema. OrderId***

***From Product- Schema. ProdId to Conprises- Schema. ProductId***

**A screenshot of a computer

Description automatically generatedCLASS DIAGRAM**

1. **MVC MODEL (MODEL - VIEW - CONTROLLER)**

We develop our website based on the MVC model, which is the MODEL\_ VIEW\_ CONTROLLER model. To be specific, this model is implemented as followed:

* User request a function from the website
* Webserver takes in the request
* Webserver requests data from the database (which is the MODEL’ s duty)
* The database responds back to the Webserver
* Webserver takes in the response
* Webserver sends data to the interface (which is the VIEW)
* Data be rendered on the interface so that User can view the data respective to the function requested
* User gets the data displays on the screen

**PROS and CONS:**

* ***Pros:***
* Fast
* Easy to update
* Easy for collaboration
* Easy to debug
* ***Cons:***
* Strong reliability and tight connection among View, Controller and Model
* Heavy work on Model

A close up of a map

Description automatically generated

* 1. ***EVENTS***

**MANAGER:**

* Manager selects SIGN IN
* Manager select SIGN UP
* Manager selects Create a product
* Manager selects a product (click on product’ s image)
* Manager selects Update a product
* Manager selects Delete a product
* Manager inputs name of a customer

**CUSTOMER**

* Customer selects SIGN IN
* Customer selects SIGN UP
* Customer inputs a product name into search box
* Customer selects a product (click on product ‘s image)
* Customer selects a category in the menu bar
* Customer selects cart
* Customer increases quantity of product in cart
* Customer decreases quantity of product in cart
* Customer selects CHECK OUT
* Customer confirms order
* Customer accesses personal page
* Customer inputs personal information

1. **FUNCTIONAL REQUIREMENTS**

***[SEE SEQUENCE DIAGRAM APPENDED IN SEQUENCE DIAGRAM.DOC]***

***[SEE ACTIVITY DIAGRAM APPENDED IN FOLDER ACTIVITY DIAGRAM]***

1. ***FOR CUSTOMER***

|  |  |
| --- | --- |
| Function 1 | SIGN IN |
| Input | ID & Password |
| Output | Success or error message through data saved in database |

|  |  |
| --- | --- |
| Function 2 | SIGN UP |
| Input | ID, password, verified password |
| Output | Success or error message through data saved in database |

|  |  |
| --- | --- |
| Function 3 | UPDATE PERSONAL INFORMATION |
| Input | New details |
| Output | Success messages + Data saves in the database |

|  |  |
| --- | --- |
| Function 4 | DISPLAY A PRODUCT |
| Input | [click on product image] |
| Output | Product Information |

|  |  |
| --- | --- |
| Function 5 | DISPLAY A CATEGORY |
| Input | [click on category name] |
| Output | Product list in the chosen category |

|  |  |
| --- | --- |
| Function 6 | ADD A PRODUCT TO CART |
| Input | [click on Add to cart] |
| Output | Chosen product be added to cart |

|  |  |
| --- | --- |
| Function 7 | UPDATE A CART QUANTITY |
| Input | [increase/ decrease product quantity] |
| Output | Cart information be updated |

|  |  |
| --- | --- |
| Function 8 | CHECK OUT AN ORDER |
| Input | Items’ details |
| Output | Update new items |

1. ***FOR MANAGER***

|  |  |
| --- | --- |
| Function 1 | SIGN IN |
| Input | ID & Password |
| Output | Success or error message through data saved in database |

|  |  |
| --- | --- |
| Function 2 | SIGN UP |
| Input | ID, password, verified password |
| Output | Success or error message through data saved in database |

|  |  |
| --- | --- |
| Function 3 | CREATE A PRODUCT |
| Input | New details |
| Output | Success messages + Data saves in the database |

|  |  |
| --- | --- |
| Function 4 | DISPLAY A PRODUCT |
| Input | [click on product image] |
| Output | Product Information |

|  |  |
| --- | --- |
| Function 5 | UPDATE A PRODUCT |
| Input | [click on Update], input new details |
| Output | Success message + data saves in database |

|  |  |
| --- | --- |
| Function 6 | DELETE A PRODUCT |
| Input | [click on Delete] |
| Output | Success message + data saves in database |

|  |  |
| --- | --- |
| Function 7 | DISPLAY A CUSTOMER’ S INFORMATION |
| Input | Input Customer’ s email/ username |
| Output | Customer’ s shipping information |

1. **NON- FUNCTIONAL REQUIREMENTS**

We expect our system to have the below **non- functional requirements**:

* **Safety and Security**: we strictly keep the system safe and security for the user; any information of the user won’t be released out or known wisely by any individuals.
* **Portability (Constraint)**: since this is a website, it is obviously easy to access whenever and wherever. However, since we have not owned a domain name, so that this have not been public and get a website url online yet. This is a school project with low expenditure for buying a domain name, so we just perform the website within our online scope
* **Usability**: The system in general and the website in particular have easy- to- perform functions; therefore, users won’t find it difficult to perform any tasks on the website:
* If user is asked to input any data, just input directly into the search box or data box
* If user is asked to select any data, just move the mouse to the chosen point
* **Speed, Capacity and Throughput (Constraint)**: we are not guarantee that we can support user with these requirements since a website speed relies on the speed of that area’ s network; so does the capacity and throughput, it will depend on the condition of the network in the area.

.

1. **USAGE SCENARIOS**

4.1 ***USER***

There are two main actors in the system: CUSTOMER and MANAGER

* Manager will perform functions relating to the website management including sign in, sign up, create products, view products, delete products, update products, view customer’ s information
* Customer will perform functions relating to the Customer online shopping demand including sign in, sign up, search a product, view a product from category, add a product to cart, update cart quantity, check out, update personal information

4.2 ***USE CASES***

A close up of a map

Description automatically generated

**USE CASE DESCRIPTION**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use case name: | **Authenticate an account** | | | | |
| Brief description: | Manager/ Student wants to sign up or sign in into his/ her account | | | | |
| Actor(s): | Manager, Student | | | | |
| Preconditions: | Manager/ Student have not had an account (for sign-up function) or had already had an account (for sign-in function) | | | | |
| Postconditions: | Manager/ Student successfully signs up an account and can sign in to perform any function(s) on demand on the compatible website (for sig-up function); or he/she are directly able to sign in into his/ her account if he/ she has already had an account (for sign-in function) | | | | |
| Flow of activities: |  | For Manager | | For Student | |
| Function | Actor | Website | Actor | Website |
| Sign- up | 1. Manager access the main site  2. Manager select “Sign up” symbol  3. Manager fill in his/ her information included:   * First name * Last name * Username * Password * Email * Contact * Address   4. Manager presses the submit button | 2.1 Website browse the registration page  4.1 Website saves the input information and prompt Manager to login site for re-confirmation | 1. Student access the main site  2. Student select “Sign up” symbol  3. Student fill in his/ her information included:   * First name * Last name * Username * Password * Email * Contact * Address   4. Student presses the submit button | 2.1 Website browse the registration page  4.1 Website saves the input information and prompt Manager to login page for re-confirmation |
|  | Sign- in | 1. Manager access the main site  2. Manager select “Sign in” symbol  3. Manager fill in his/ her information included:   * Username * Password   4. Manager presses the submit button | 2.1 Website browse the login page  4.1 Website gets the input information and prompts Manager to compatible page for performing any function on demand | 1. Student access the main site  2. Student select “Sign in” symbol  3. Student fill in his/ her information included:   * Username * Password   4. Student presses the submit button | 2.1 Website browse the login page  4.1 Website gets the input information and prompts Student to compatible page for performing any function on demand |
| Exceptional conditions: | Sign- up  Sign- in | Manager/ Student fails to have their account created ( due to duplication account)  Manager/ Student forgets his/ her username  Manager/Student forgets his/ her password  Manager/Student fails to sign in into the website since the system cannot find/ recognize his/ her account | | | |

|  |  |  |
| --- | --- | --- |
| Use case name: | **View a product** | |
| Brief description: | Manager wants to view a product available on website | |
| Actor(s): | Manager | |
| Preconditions: | Manager have already signed in into the website | |
| Postconditions: | Manager successfully views and gets the information of a product or performs any related activity relating to in- need product | |
| Flow of activities: | Actor | Website |
| 1.Manager chooses the activity View a product  2. Manager insert keyword(s) relating to in- need product including:   * Name * Id * Category * Description * Brand   3. Manager selects a product from resulting list | 1.1. Website browse search box  2.1 Website searches for relating products  2.2 Website displays list of relating products  3.1 Website browses the product  3.2 Website displays product |
| Exceptional conditions: | At 2.2, Website cannot find any relating products | |

|  |  |  |
| --- | --- | --- |
| Use case name: | **Create a product** | |
| Brief description: | Manager wants to create a product on website | |
| Actor(s): | Manager | |
| Related use case: | Manage website | |
| Preconditions: | Manager have already signed in into the website | |
| Postconditions: | Manager successfully creates a new product and has it uploaded on the website for customer to view and buy | |
| Flow of activities: | Actor | Website |
| 1.Manager chooses the activity Create a product  2.Manager fills in the information of the new product including:   * Name * Description * Origin * Brand * Category * Price * Discount   3. Manager presses the submit button | 1.2 Website browses the Product page for Manager to fill in the information  3.1 Website takes in and saves the input information of the new product including:   * Name * Description * Origin * Brand * Category * Price * Discount   3.2 Website displays success message |
| Exceptional conditions: | N/A | |

|  |  |  |
| --- | --- | --- |
| Use case name: | **Delete a product** | |
| Brief description: | Manager wants to delete a product on website | |
| Actor(s): | Manager | |
| Related use case: | View a product | |
| Preconditions: | Manager have already signed in into the website and is currently viewing the product to be deleting | |
| Postconditions: | Manager successfully delete a product out of the website | |
| Flow of activities: | Actor | Website |
| 1.Manager chooses the activity Delete a product  2. Manager confirms to delete | 1.1 Website browses confirmation box  2.1 Website delete the product on the website and saves the new update of website  2.2 Website displays success message |
| Exceptional conditions: | N/A | |

|  |  |  |
| --- | --- | --- |
| Use case name: | **Update a product** | |
| Brief description: | Manager wants to update a product on website | |
| Actor(s): | Manager | |
| Related use case: | Manage website | |
| Preconditions: | Manager have already signed in into the website and is currently viewing the product to be updating | |
| Postconditions: | Manager successfully delete a product out of the website | |
| Flow of activities: | Actor | Website |
| 1. Manager chooses the activity Update product  2. Manager fills in the information in the information box for update | 1.2 Website browses information box  2. 1Website updates and saves the information of the product  2.2 Website displays success message |
| Exceptional conditions: | N/A | |

|  |  |  |
| --- | --- | --- |
| Use case name: | **Manage a customer’ s account** | |
| Brief description: | Manager wants to check information of a customer due to wrong or unreachable delivery, or non- exist identity | |
| Actor(s): | Manager | |
| Related use case: | N/A | |
| Preconditions: | Manager have already signed in into the website | |
| Postconditions: | Manager successfully views any needed information of a customer | |
| Flow of activities: | Actor | Website |
| 1.Manager chooses the activity View a customer  2.Manager searches for any information of the needed customer including:   * [First name]- [Last name] * Username * Contact * Email * Address   3.Manager chooses customer from the returning result(s) for viewing | 1.2 Website browses the Customer page for Manager to choose a customer to view  2.1 Website searches for realting customer  2.2 Website displays list of relating customers  3. Website creates customer reports consists of customer’s public accessible information including:   * First Name- Last Name * Address * Contact * Email   and order history including:   * Shopping cart * Delivery information   3.2 Website displays report |
| Exceptional conditions: | At stage 2.1, Website cannot find out the customer for viewing | |

|  |  |  |
| --- | --- | --- |
| Use case name: | **View a product from category** | |
| Brief description: | Customer wants to view information of a product available on given category | |
| Actor(s): | Customer | |
| Related use case: | N/A | |
| Preconditions: | Customer have already signed in into his/ her account | |
| Postconditions: | Customer successfully view and get the information of needed product | |
| Flow of activities: | Actor | Website |
| 1.Customer looks for given category on menu bar  2.Customer chooses a category  3. Customer chooses a product from resulting category | 1.2 Website browses the Category page for Customer to look for needed product(s)  2.1 Website browses category  3.1 Website browses products  3.2 Website displays chosen product’s information including:   * Name * Description * Origin * Brand * Category * Price * Discount |
| Exceptional conditions: | At stage2.1, Website cannot find product | |

|  |  |  |
| --- | --- | --- |
| Use case name: | **Search a product** | |
| Brief description: | Customer wants to search for a product via the search box | |
| Actor(s): | Customer | |
| Related use case: | N/A | |
| Preconditions: | Customer have already signed in into his/ her account | |
| Postconditions: | Customer successfully search, view and get the information of needed product | |
| Flow of activities: | Actor | Website |
| 1.Customer looks for a search box on menu bar  2.Customer inputs any keywords relating to the product on demand inside the search box including:   * Name * Description * Origin * Brand * Category   3. Customer chooses a product from resulting list | 2.1 Website takes in the input and browses products  2.2 Website creates and displays list of relating products  3.1 Website browses the chosen product  3.2 Website displays information of product including:   * [Image] * Name * Description * Origin * Brand * Category * Price * Discount |
| Exceptional conditions: | At stage2.1, Website cannot find any relating products | |

|  |  |  |
| --- | --- | --- |
| Use case name: | **Buy an order** | |
| Brief description: | Customer wants to make an order including adding products to his/ her cart, updating his/ her cart, and finally check out his/ her cart | |
| Actor(s): | Customer | |
| Related use case: | View a product from category  Search a product  Add a product to cart  Update a cart  Check out a cart | |
| Preconditions: | Customer have already signed in into his/ her account and is currently viewing the product he/ she wants to buy | |
| Postconditions: | Customer successfully makes an order | |
| Flow of activities: | Actor | Website |
| 1.Customer is in the present time viewing the product he/ she wants to buy  2.Customer looks for the button “Add to cart” in the page displaying the product on demand’ s information and presses that button   * 3. Case 1: Customer wants to Update the product(s) quantity: * 4.Customer chooses activity Update cart * 5.Customer adjusts the quantity of the product in cart by increasing(upward arrow or decreasing (downward arrow ) the number of products on demand; if he/ she wants to permanently delete the product out of the cart, decreases the number to 0 * 3.Case 2: Customer wants to Check out * 4.Customer chooses activity Check out * 5. Customer fills in Delivery Information in confirmation box | 2.1 Website browses product in the system  2.2 Website adds product to customer’ cart  4.1 Website browses cart  5.1 Website adjust quantity of product in system stock  5.2 Website calculate new price  5.3 Website return updated cart  4.1 Website browses cart  4.2 Website requests Customer to confirm  5.1 Website check for information validation  5.2 Website creates new order for Customer  5.3 Website saves order  5.4 Website sends success message informing that Customer successfully makes an order |
| Exceptional conditions: | At stage 2.1, Website found no remaining product in system stock  Case 1: At 5.1, Website found no remaining product in system stock for adjustment  Case 2: At 5.1, Website denies order confirmation due to unreliable or missing information | |

|  |  |  |
| --- | --- | --- |
| Use case name: | **Update an account** | |
| Brief description: | Customer wants to update the personal information saved on the Website | |
| Actor(s): | Customer | |
| Related use case: | N/A | |
| Preconditions: | Customer have already signed in into his/ her account | |
| Postconditions: | Customer successfully updates the product(s)’ s quantity(s) in his/ her cart | |
| Flow of activities: | Actor | Website |
| 1.Customer looks for personal account setting symbol on the menu bar and chooses the symbol  2. Customer edit the personal information in the account page including:   * First name * Last name * Username * Contact * Email * Address   3. Customer presses the “Update” button | 1.1 Website browse the personal account page    3.1 Website takes in the update and saves the new information of Customer  3.2 Website sends a message informing that Customer has successfully updated his/ her information |
| Exceptional conditions: | At stage3.1, Website fails to save the Customer updated information | |

1. **IMPLEMENTATION**

A screenshot of a cell phone

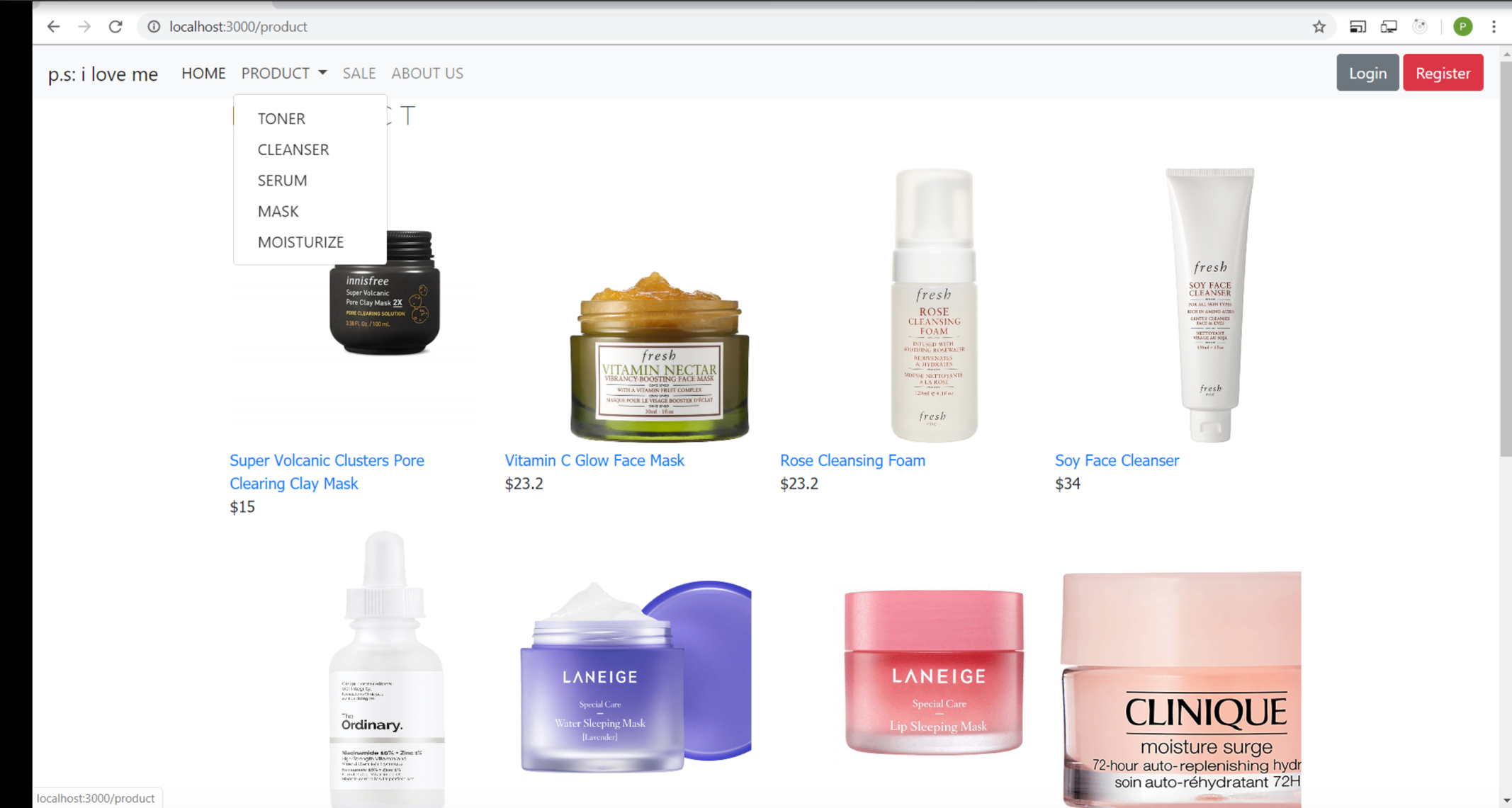
Description automatically generated

A screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated



A screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated