Technical Story Card

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Version No. | Date | Prepared by / Modified by | Significant Changes |
| 1 | 06 Oct 2022 | Team 1:  - Le Tan Quy  - Nguyen Minh Thanh  - Nguyen Trong Huy  - Mai Huynh Trung Nguyen | First version |
|  |  |  |  |

Glossary

|  |  |
| --- | --- |
| Abbreviation | Description |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Table of Contents

[1 Introduction 3](#_Toc116010045)

[2 Scope of Change 3](#_Toc116010046)

[3 List of impacted modules 3](#_Toc116010047)

[4 Design and Detailed technical updates 4](#_Toc116010048)

[4.1 Process model 4](#_Toc116010049)

[4.1.1 Use case Model 4](#_Toc116010050)

[4.1.2 Sequence diagram 15](#_Toc116010051)

[4.2 Proposed user Interface design 16](#_Toc116010052)

[5 Details of Alternative Design Approach 16](#_Toc116010053)

[6 Other Technical changes 17](#_Toc116010054)

[6.1 Automation tasks 17](#_Toc116010055)

[6.2 CI / Build relates tasks 17](#_Toc116010056)

[6.3 Non-functional related changes 17](#_Toc116010057)

[7 Additional details 17](#_Toc116010058)

[7.1 Open Questions / clarifications / Assumptions 17](#_Toc116010059)

[7.2 Additional notes to technical team 17](#_Toc116010060)

[8 References 17](#_Toc116010061)

# Introduction

Today, with the development of the internet, non-stores can reach more customers and sell more products with online sales tools.

The seller only needs to rely on the announced order to provide his product to the customer. Delivery services are available to assist stores in delivering products to customers.

For modern customers, they don't need to go to a store to shop for the items they like. As long as they have the internet, they will visit the website of their favorite store and choose the products they want.

Online shopping website is a highly practical project. Can be applied to many different cases and bring great benefits.

# Scope of Change

Change will not be entertained and whatever specifications mentioned in this document is final.

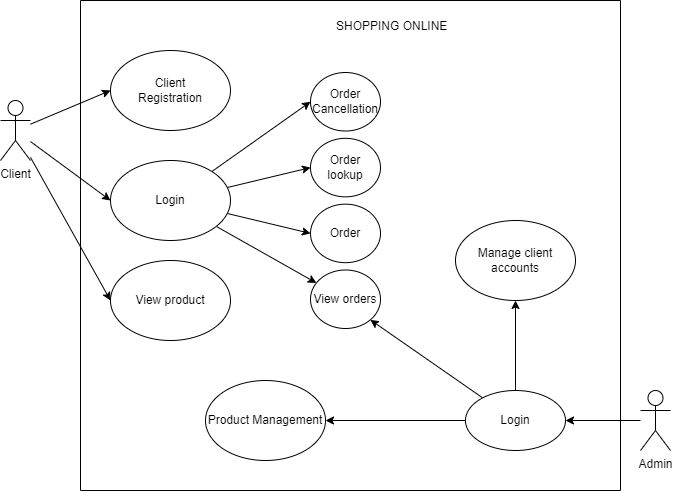
# List of impacted modules

All the functional modules will be created from scratch.

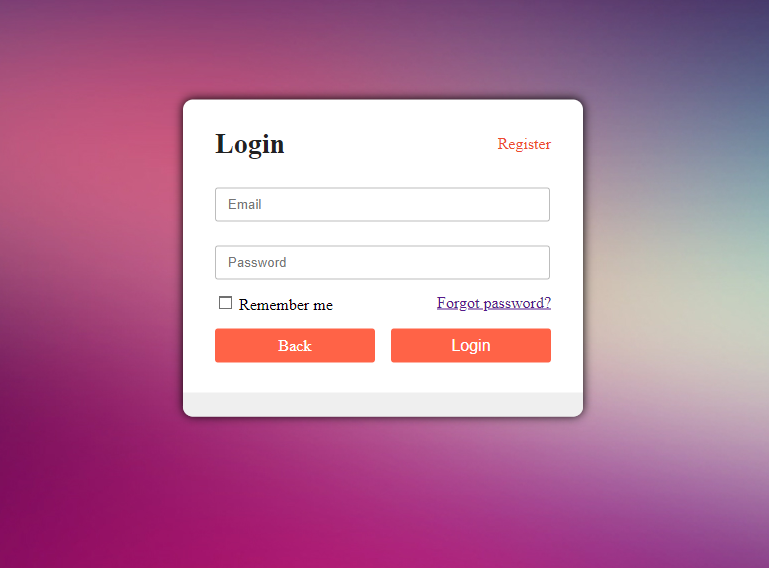
# Design and Detailed technical updates

## Process model

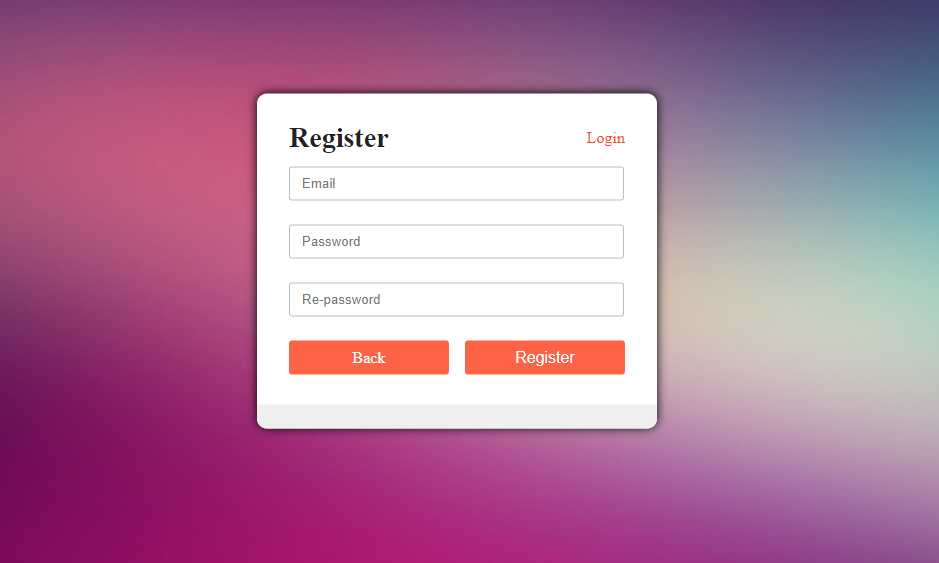
### Use case Model



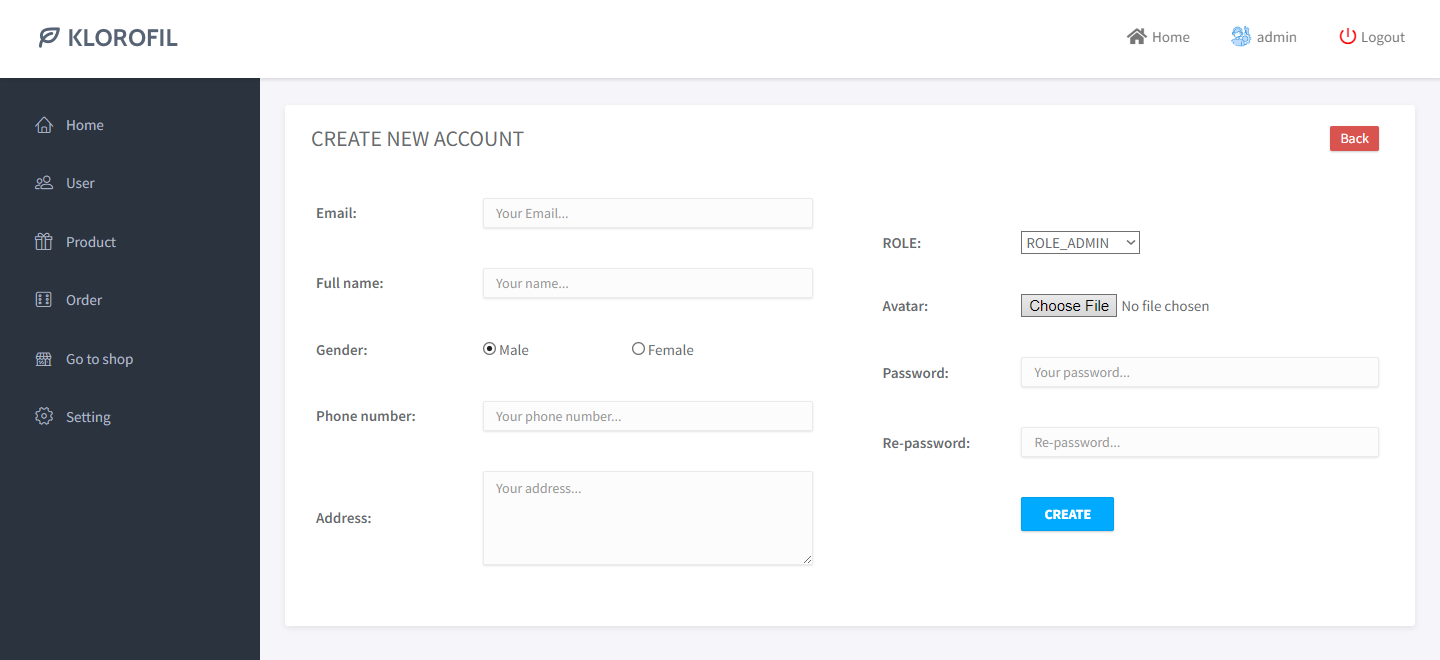
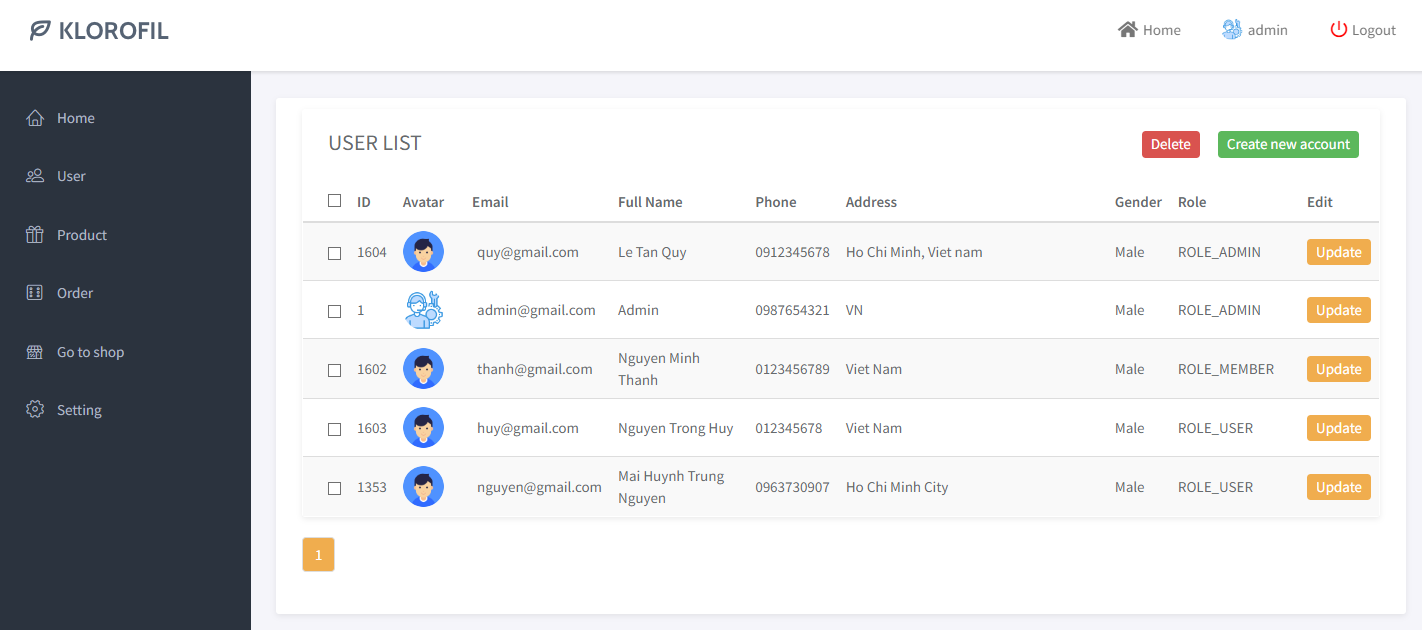
|  |  |
| --- | --- |
| **Brief Description** | User Login |
| **Basic Flow** | This use case describes how login in to the system   1. User must have a previous account 2. User enters email with correct password according to registered account 3. When you enter the wrong one, an error message will appear 4. When the correct user is entered, the user will access the allowed features according to the account's authorization |
| **Alternate Flow** | 1. When you enter the wrong one, an error message will appear 2. When the correct user is entered, the user will access the allowed features according to the account's authorization |
| **Validation** | Already user and valid password. |
| **Pre-Conditions** | User should have network access and Browser with latest updates. |
| **Post-Conditions** | Success message should be shown. |



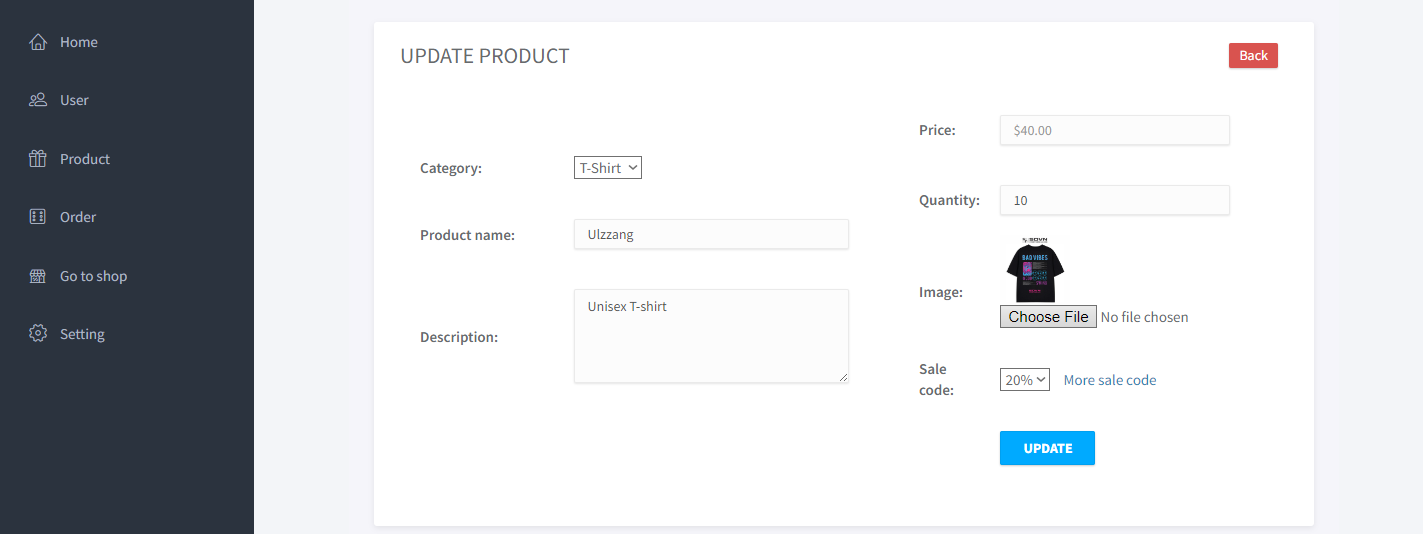
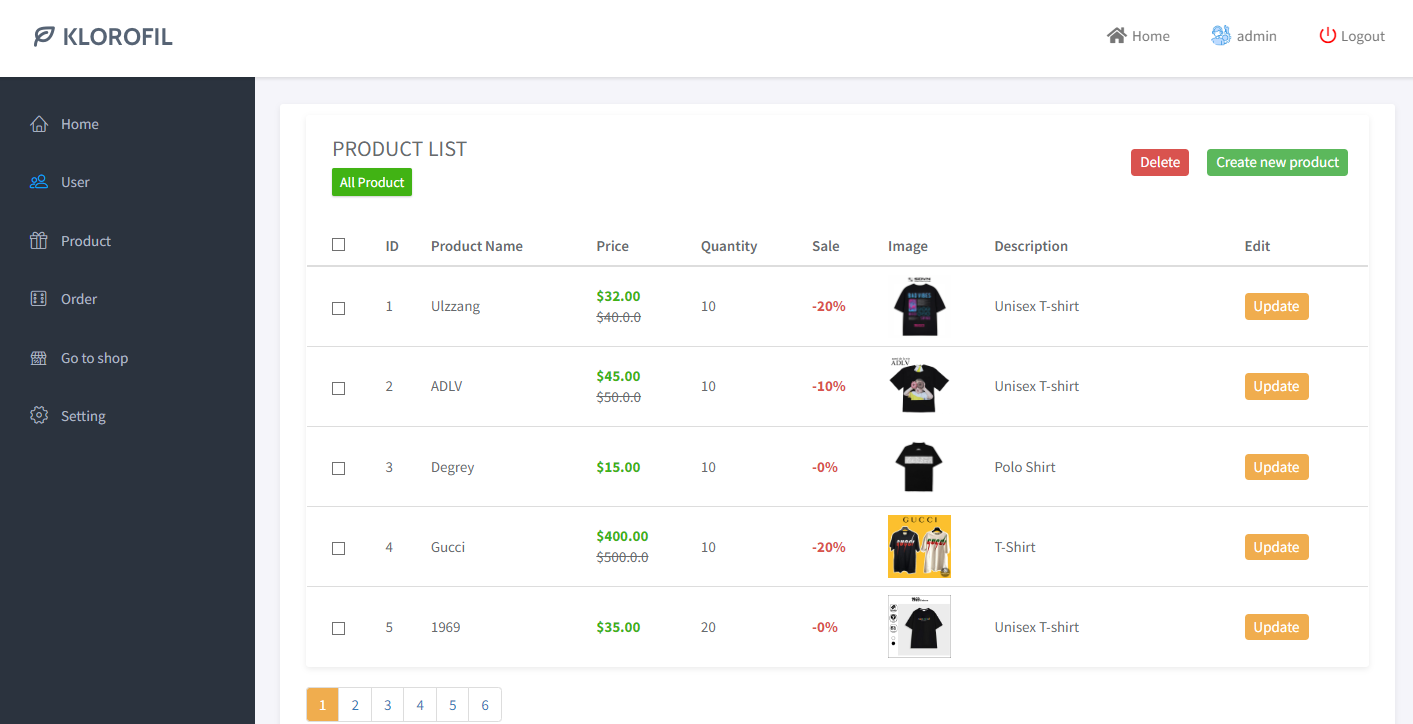
|  |  |
| --- | --- |
| **Brief Description** | Client Registration |
| **Basic Flow** | This use case describes how a client register in to the system   1. The client has to register themself into the system. 2. After the successful registration, client will get a success message. 3. The following information is required during registration.  * Email * Password * Re-enter password  1. Others informations can update later. |
| **Alternate Flow** | The system will validate the information provided. If any invalid data is found, the input form will be redirected with error message. |
| **Validation** | 1. Email doesn’t already exists. 2. Password and re-password must match. |
| **Pre-Conditions** | User should have network access and Browser with latest updates. |
| **Post-Conditions** | Success message should be shown. |



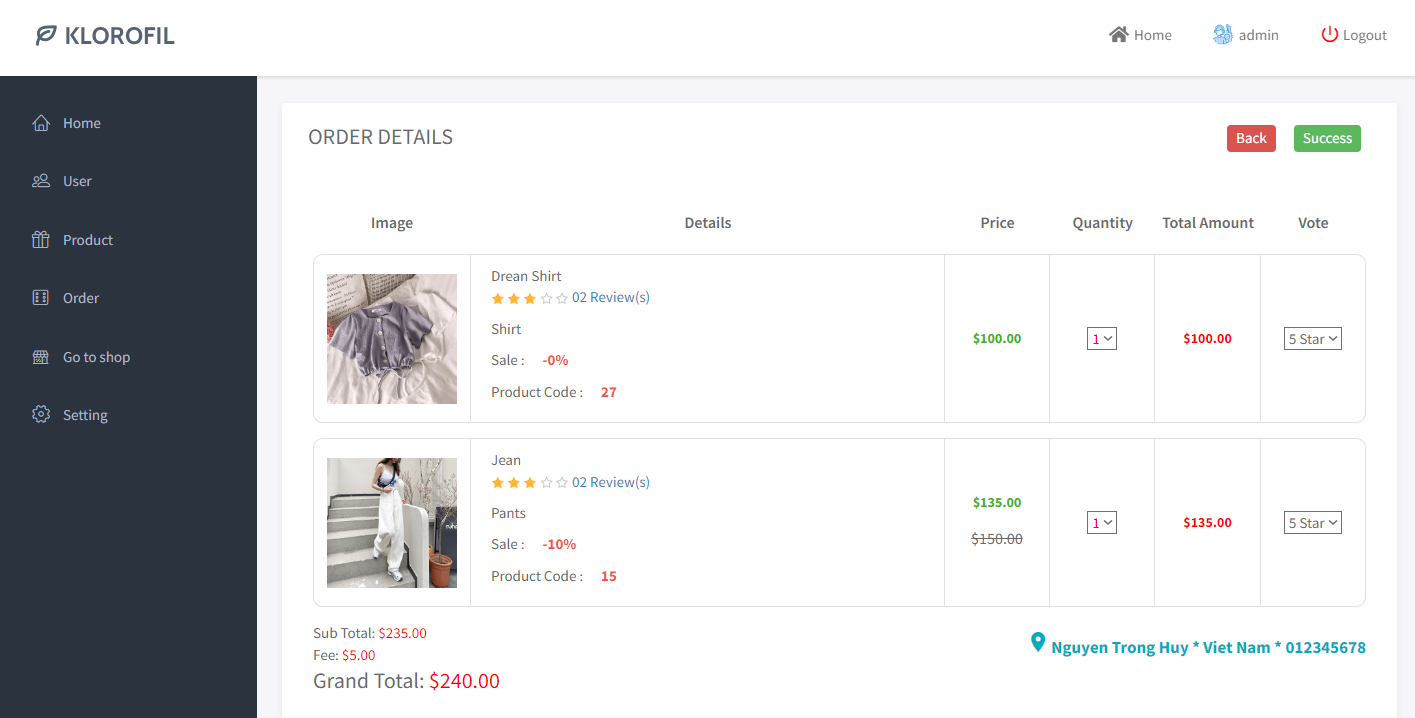
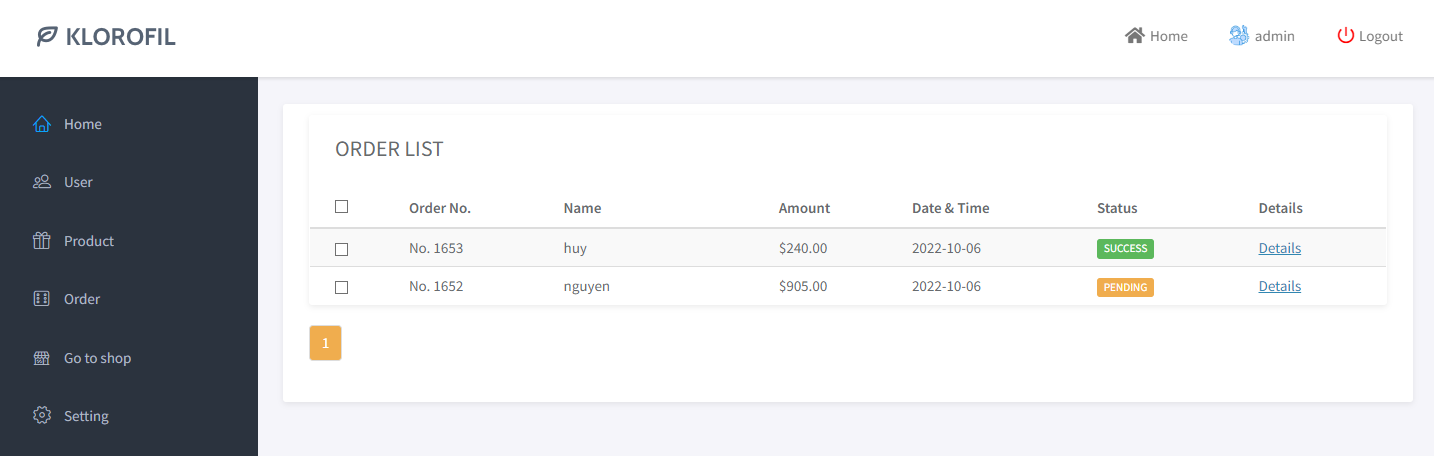
|  |  |
| --- | --- |
| **Brief Description** | User Management (Admin) |
| **Basic Flow** | User with role is “admin” can access “Admin Page”. At the “User management Page”, admin can:   1. View full list of user (with full of informations). 2. Create a new user. 3. Delete user. 4. Update information of users. |
| **Alternate Flow** | The system will validate the information provided. If any invalid data is found, the input form will be redirected with error message. |
| **Validation** |  |
| **Pre-Conditions** | User should have network access and Browser with latest updates. |
| **Post-Conditions** | Success message should be shown. |



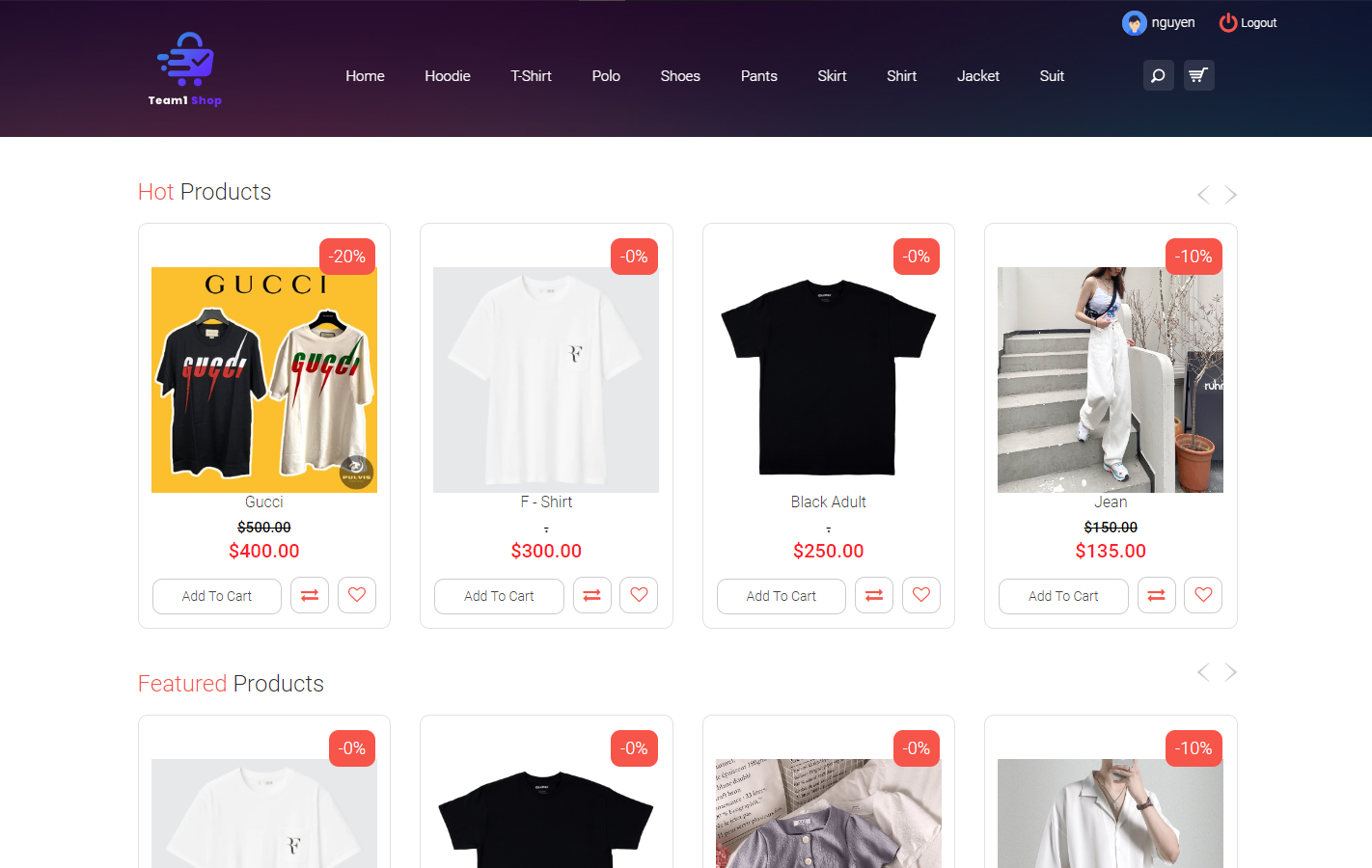
|  |  |
| --- | --- |
| **Brief Description** | Product Management (Admin) |
| **Basic Flow** | Admin can:   1. View full list of products (with full of informations). 2. Create a new product. 3. Delete product. 4. Update information of products. |
| **Alternate Flow** | The system will validate the information provided. If any invalid data is found, the input form will be redirected with error message. |
| **Validation** |  |
| **Pre-Conditions** | User should have network access and Browser with latest updates. |
| **Post-Conditions** | Success message should be shown. |

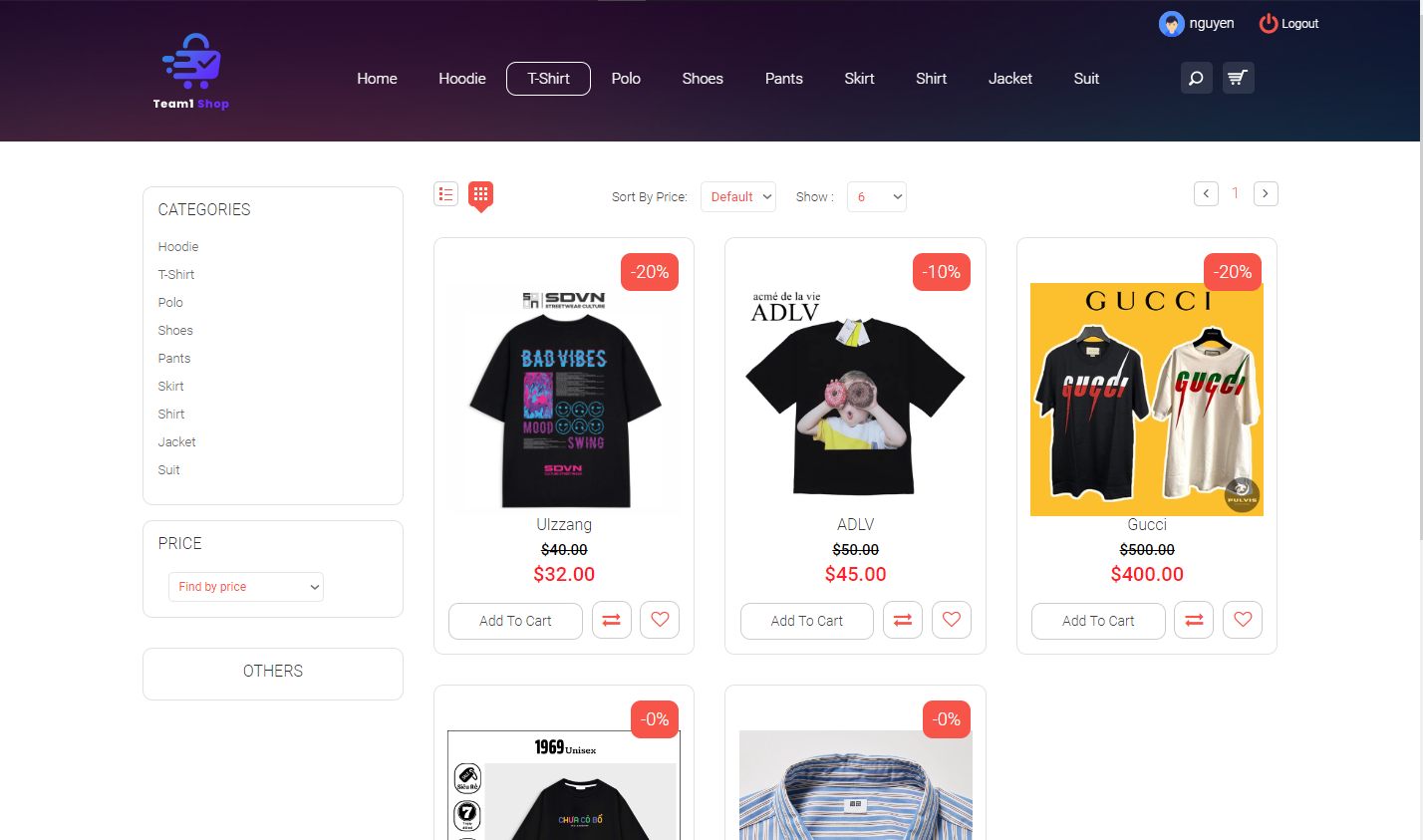


|  |  |
| --- | --- |
| **Brief Description** | Order Management (Admin) |
| **Basic Flow** | Admin can:   1. View full list of order (with full of informations). 2. View detail of a order. 3. Update status of a order (from PENDING to SUCCESS). |
| **Alternate Flow** | The system will validate the information provided. If any invalid data is found, the input form will be redirected with error message. |
| **Validation** |  |
| **Pre-Conditions** | User should have network access and Browser with latest updates. |
| **Post-Conditions** | Success message should be shown. |

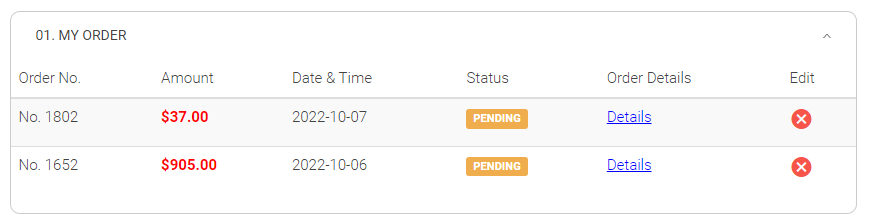
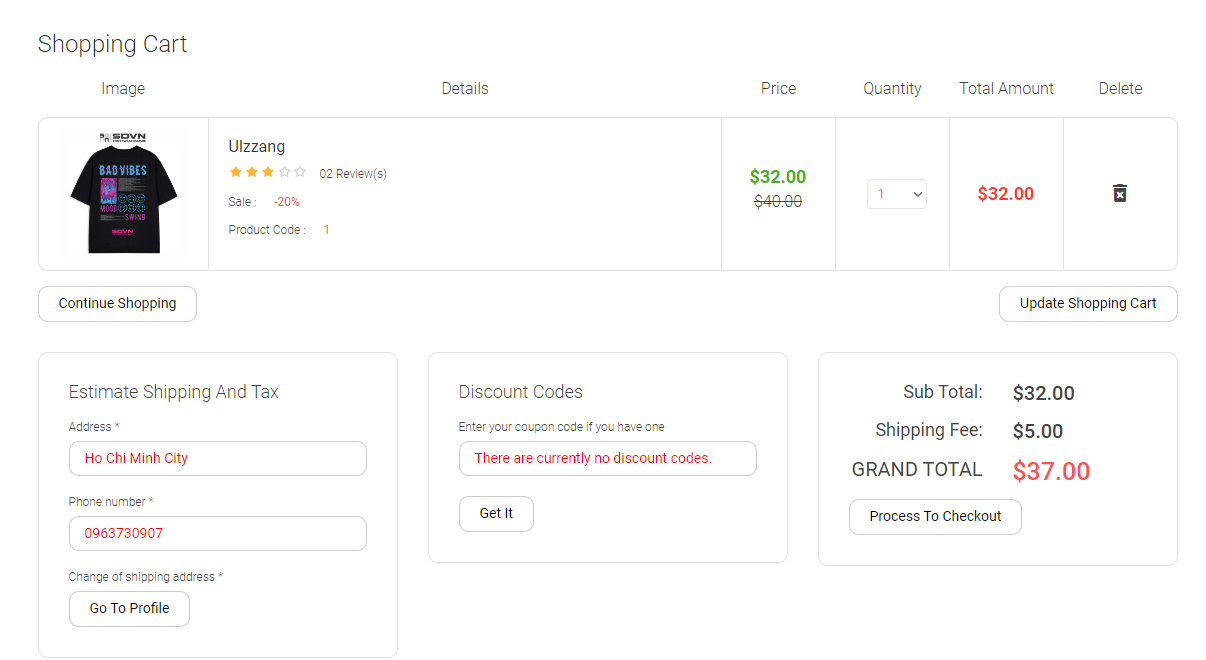


|  |  |
| --- | --- |
| **Brief Description** | View Client Home Page |
| **Basic Flow** | User don’t need to login system. Everyone can view:   1. Hot and featured products. 2. List of producted by categories. |
| **Alternate Flow** | The system will validate the information provided. If any invalid data is found, the input form will be redirected with error message. |
| **Validation** |  |
| **Pre-Conditions** | User should have network access and Browser with latest updates. |
| **Post-Conditions** | Success message should be shown. |

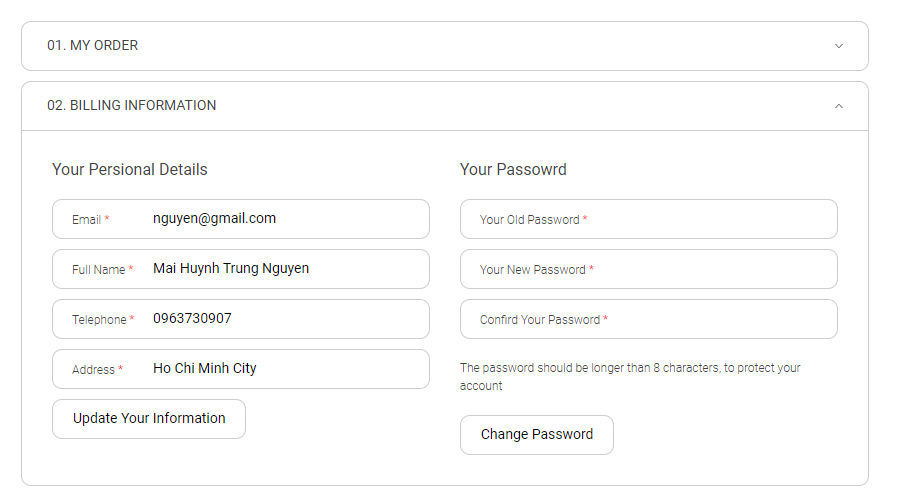




|  |  |
| --- | --- |
| **Brief Description** | View Cart (User) |
| **Basic Flow** | 1. User click “Add To Card” at the product they like. 2. At “Cart Page”, users can continue to shopping or checkout. 3. After Checkout, their order will be sent to system to processing. |
| **Alternate Flow** | The system will validate the information provided. If any invalid data is found, the input form will be redirected with error message. |
| **Validation** |  |
| **Pre-Conditions** | User should have network access and Browser with latest updates. |
| **Post-Conditions** | Success message should be shown. |



|  |  |
| --- | --- |
| **Brief Description** | View Profile (User) |
| **Basic Flow** | 1. Users can click to their avatar to view the profile. 2. They can change information, password here. |
| **Alternate Flow** | The system will validate the information provided. If any invalid data is found, the input form will be redirected with error message. |
| **Validation** |  |
| **Pre-Conditions** | User should have network access and Browser with latest updates. |
| **Post-Conditions** | Success message should be shown. |

**Architecture Diagram:**

**Logical View**

**Technology/ Framework**

**Layer**

HTML/ CSS/ JavaScript/ JSP

UI Components

Presentation Layer

Resources

Validations

Export (pdf)

Report

Web-Server Classes

Java 1.8/ Spring MVC

Application/ Business Layer

Controllers

Configuration

REST Controllers

Service

Entity/ Model Classes

Data Access Layer

Hibernate ORM

Repository classes

Oracle 11g Database

**Presentation Layer:**

This layer consists of all UI components.

* Export component: The plug-in is used to export the reports in pdf format. It helps to reduce the workload on the web/ application server.
* Validation component: All basic level data validations should be done at UI level.
* Report component: All reports to be generated to the user, will be processed and generated at the browser end.
* Resources: All HTML/CSS/ images, which are required for the page design.

**Application Layer:**

This layer comprises of all server and business classes.

* Configuration settings (XML or Class based) will define the application and server configuration.
* Spring Controllers defines the server classes, which are required to process the incoming http requests.
* Service classes are required to perform the required business services.
* Controllers to process the HTTP requests.
* Model classes are used the define the functions of the entities present in the system.

**Data Layer:**

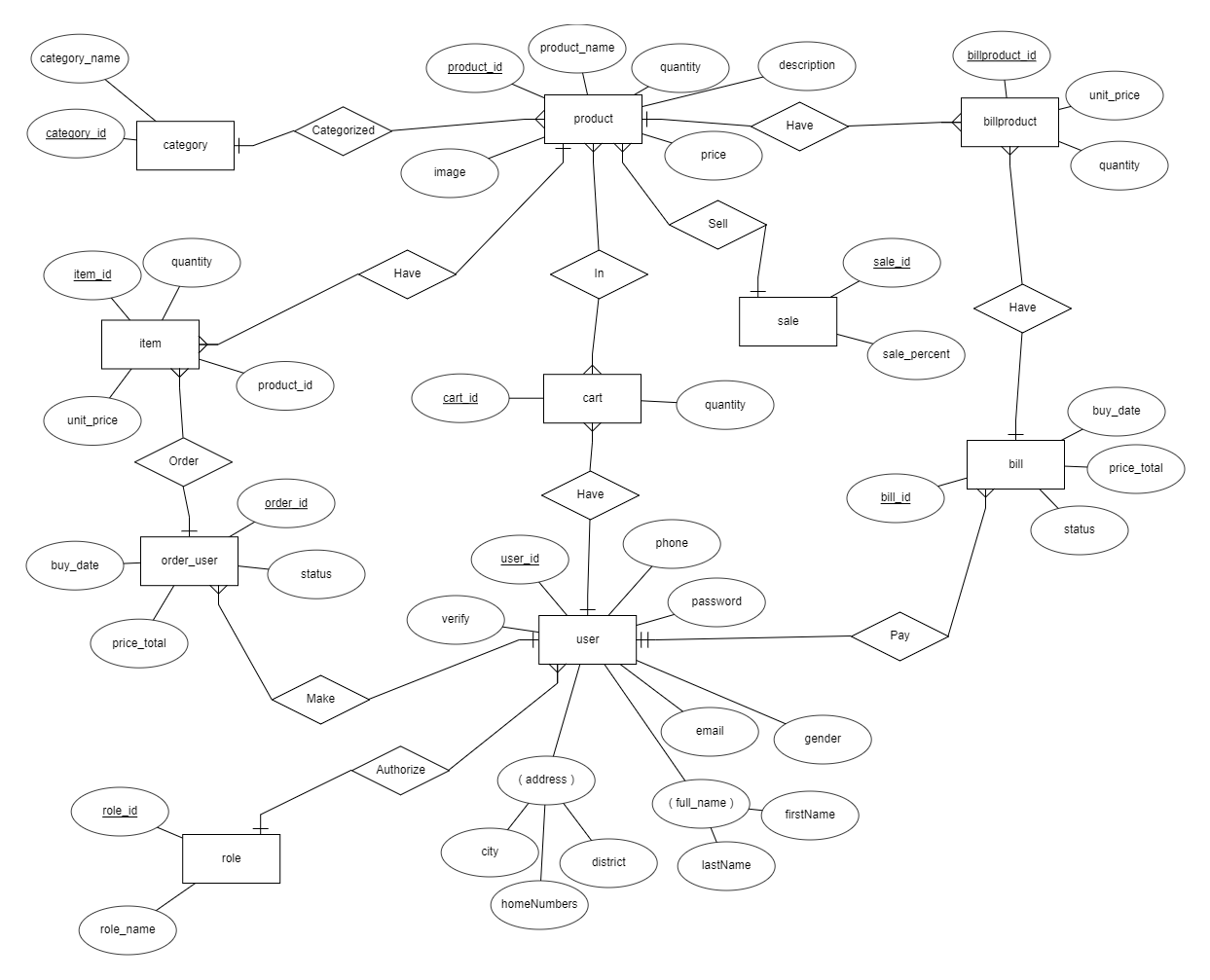
Data layer is implemented through Hibernate ORM. It will contain the repository classes, which provides interface to the table.

**Class Diagram:**



### Sequence diagram

**ERD Diagram:**



## Proposed user Interface design

The interface is similar to what is described in the modules in the previous section.

# Details of Alternative Design Approach

<Details of the alternative approach to resolve the maintenance request>

# Other Technical changes

## Automation tasks

<if the story is about automation (of test cases or some POC etc.), mention the details here>

## CI / Build relates tasks

<If the story requirements are about implementation of continuous integration or build, update the details to be done here>

## Non-functional related changes

<Non-functional requirements related to response time, System load or Network load>

# Additional details

## Open Questions / clarifications / Assumptions

<Mention the details of any open technical questions/clarifications or assumptions made, pertaining to story >

## Additional notes to technical team

<provide details if any>

# References

< Provide reference details if any >