****

Software Requirements Specification

for

**JEWELS SHOP MANAGEMENT**

**Version 1.0 approved**

**Prepared by Team 3**

**FPT University**

**05/10/2021**

**SIGNATURE PAGE**

**ORIGINATOR:** Bui Manh Truong – SE06055

**TEAM MEMBER:** Le Duy Hung – SE05743

Hoang Viet Hung – SE06068

Tran Minh Duc – HE150202

Bui Ngoc Loc – SE05457

**REVIEWERS:** Pham Ngoc Tho

[**Table of Contents iv**](#_28h4qwu)

[**Revision History v**](#_3znysh7)

[**1.**](#_2et92p0) **Introduction 1**

[1.1 Purpose 1](#_tyjcwt)

[1.2 System Purpose 1](#_1t3h5sf)

[1.3 Definitions, Acronyms and Abbreviations 2](#_4d34og8)

[1.4 Document Conventions 2](#_2s8eyo1)

[1.5 Intended Audience and Reading Suggestions 2](#_17dp8vu)

[1.6 Project Scope 2](#_3rdcrjn)

[1.7 References 2](#_26in1rg)

[**2.**](#_lnxbz9) **Overall Description 3**

[2.1 Product/System Perspective 3](#_35nkun2)

[2.2 System/Product Features 3](#_1ksv4uv)

[2.3 User requirements 4](#_44sinio)

[2.4 User Classes and Characteristics 4](#_2jxsxqh)

[2.5 Operating Environment 4](#_z337ya)

[2.6 Design and Implementation Constraints 5](#_3j2qqm3)

[2.7 User Documentation 5](#_1y810tw)

[2.8 Assumptions and Dependencies 5](#_4i7ojhp)

[**3.**](#_2xcytpi) **Specific Requirements 6**

[3.1 Functional Requirements Specification 6](#_1ci93xb)

[3.1.1 Registration 7](#_3whwml4)

[3.1.2 Login 8](#_2bn6wsx)

[3.1.3 Change information 10](#_qsh70q)

[3.1.4 Create an order 12](#_3as4poj)

[3.1.5 Add/modify a product 14](#_1pxezwc)

[3.1.6 See information of user 15](#_49x2ik5)

[3.1.7 Delete a user 16](#_2p2csry)

[3.1.8 See information of order 18](#_147n2zr)

[3.1.9 Accept/reject an order 20](#_3o7alnk)

[3.1.10 See report 21](#_23ckvvd)

[3.2 Non-Functional Requirements Specification 23](#_ihv636)

[3.2.1 External Interface Requirements 23](#_32hioqz)

[3.2.2 Other Nonfunctional Requirements 30](#_vx1227)

[**4.**](#_2u6wntf) **Other Requirements 32**

[Appendix A: Database diagram 32](#_19c6y18)

[Appendix B: Database entity 33](#_3tbugp1)

**Revision History**

| **Name** | **Date** | **Reason For Changes** | **Version** |
| --- | --- | --- | --- |
| The whole document | 05/10/2021 | Initiate version for Jewels shop | 1.0 |
|  |  |  |  |

1. **Introduction**

## Purpose

This document gives information about Shopping system website named “Jewels shop”. All functions, information of its, how it be coded, how it works, and all software component will be shown. This document also has a lot of useful for next team developer if another version be developed in the future, support for tester if need fix bug.

## System Purpose

| **Software** | **Version** |
| --- | --- |
| Microsoft Windows Server | 2010 |
| Microsoft SQL Server | 2019 |
| Microsoft .NET Framework | 4.7.2 |
| Microsoft Visual Studio | 2019 |
| Google Chrome | 94.0.4606.71 |

## Definitions, Acronyms and Abbreviations

| ID | Acronyms | Definition |
| --- | --- | --- |
| 1 | SRS | Software Requirement Specification |
| 2 | DB | Database |
| 3 | GUI | Graphic User Interface |
| 4 | OS | Operating system |
| 5 | JS | Jewels shop |
| 6 | N/A | Not applicable, not available |
| 7 | GB | Gigabyte |

## Document Conventions

This document used Template System Requirement Specification IEEE830 version 1.3. Form-based specifications will be used a lot to describe each of function, input, output, where they come, where they out, information needed, action be taken, pre, post conditions, side effect (if any) of the function.

## Intended Audience and Reading Suggestions

Developer, tester should read document before update coding or fix bugs of previous version.

## Project Scope

Name of this project is Jewels shop. It is a website for a Jewelry store that specializes in selling products for active youth and adults. This site focuses on product entry management, sales management, report management, and user management.

## References

* Software requirement specification form.
* Software engineering – Ninth Edition (by Ian Sommerville)
* Designing with Web Standards (2nd Edition)
* Model- View - Controller pattern([https://en.wikipedia.org/wiki/Model–view–controller](https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller))

1. **Overall Description**

## Product/System Perspective

* Jewels shop focus on users who:
  + Want to buy product from Jewels shop online
  + Do not have time come to shop and buy product
* On store side, they can:
  + Manage their users: registration, login
  + Manage product in shop: import, sell, …
  + See report: users, product, revenue
  + Manage import product
  + Blog
* Jewels shop is a website. It run on any operating system has web browser and Internet connection available.

## System/Product Features

* Manage importing product:
  + Add new product
  + Update quantity of product
  + Create report/bill
* Manage shopping:
  + Search a product
  + See detail of product
  + Order
  + Check status of order
  + Accept order
  + Pay
* Create report:
  + Number of products
  + Revenue
  + Number of orders
  + Number of user access
* Manage Blog:
  + View blog
  + Create blog
* Manage users:
  + Login
  + Registration
  + Update information
  + Logout

## User requirements

User need a device in any operating system has web browser and Internet connection. Then user must create an account in Jewels shop website to use full features of website.

Jewels shop shop will have admin account to use another feature.

## User Classes and Characteristics

Based on analysis, gold price and people's demand. Number of products selling for Jewels shop are increasing too. Jewels shop is a shop which realize that, so they increase limit of selling their products. 2021 is preparation year for 4.0 revolution, they think that a website will help they sell more product instead selling for a limit number of people. They want to sell product for people not only live in near shop, but also from anywhere.

## Operating Environment

* Main programming language: C# with ASP.NET.
* Display: HTML, CSS.
* IDE:
  + Microsoft visual studio 2019
  + .NET framework 3.5
  + Visual studio code version 1.60.1
* Database: manage in Microsoft Azure, query use Microsoft SQL Server Management Studio version 19.
* Project workspace, source code: Github.

## Design and Implementation Constraints

* Coding used Model-View-Controller architectural model.
* Variables used camel case (words are joined without spaces and are lowercase with first letter)
* Methods, functions must be commented to support maintenance programmers (comment before functions, before loop, before long code) to understand easily code.
* Code must be coded by convention Java code convention of Oracle.
* Html, Css must be coded by convention Designing with Web Standards (read about they in reference).

## User Documentation

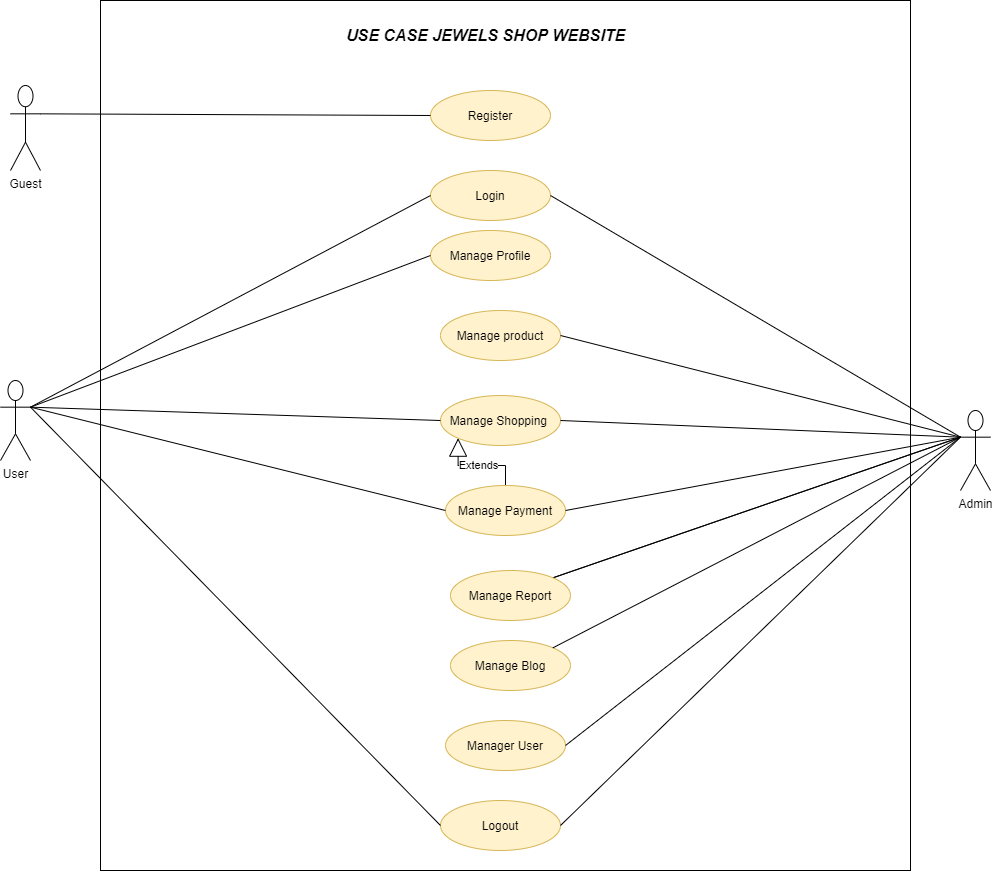
The user documentation will be able as a function of a website, it will use screen captures of website to describe all features.

## Assumptions and Dependencies

Nothing assumptions and dependencies in JS website version 1.0.

1. **Specific Requirements**

## Functional Requirements Specification



**Use Case List**

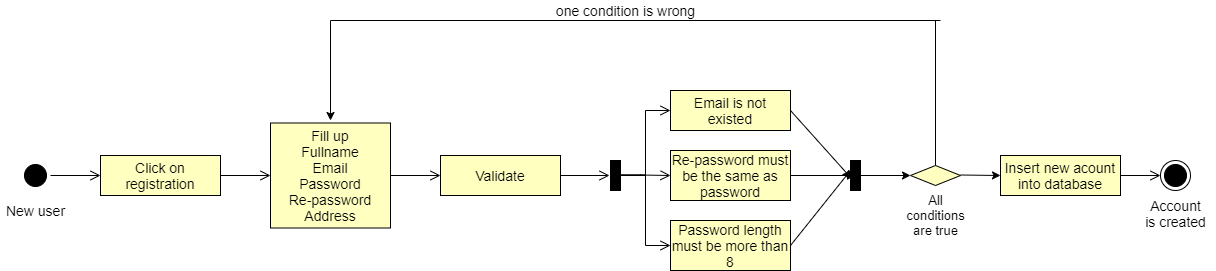
| **UC** | **Use case** | **Primary Actors** | **Secondary Actors** |
| --- | --- | --- | --- |
| 01 | Register | Guest |  |
| 02 | Login | User , Admin |  |
| 03 | Logout | User , Admin |  |
| 04 | Manage Profile | User |  |
| 05 | Manage Product | Admin |  |
| 06 | Manage Shopping | User | Admin |
| 07 | Manage Payment | User | Admin |
| 08 | Manage Report | Admin |  |
| 09 | Manage Blog | Admin |  |
| 10 | Manage User | Admin |  |

#### System Actors

| **No** | **Actor** | **Description** |
| --- | --- | --- |
| 1 | Guest | is the person who only have to see the overview of the website, not to use the function |
| 2 | User | is the user of the system's services |
| 3 | Admin | is the person who manages the website operation |

### Register

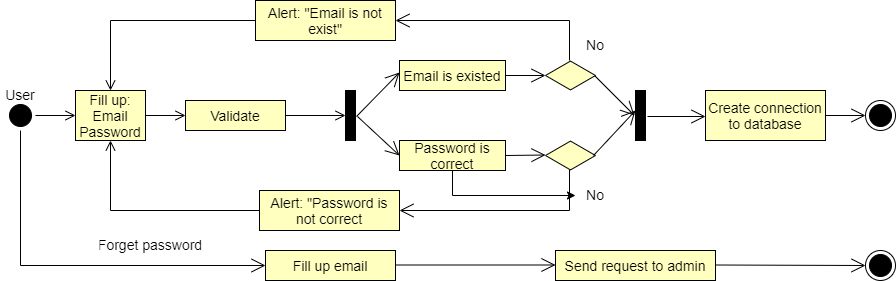
| Use Case ID: | JS01 | | |
| --- | --- | --- | --- |
| **Use Case Name:** | Sign up for user | | |
| **Created By:** | DucTM | **Last Updated By:** | DucTM |
| **Date Created:** | Oct 8, 2021 | **Date Last Updated:** | Oct 8, 2021 |
| **Actors:** | Guest | | |
| **Triggers** | User clicks on registration on the header of website | | |
| **Description:** | Registration for a new user | | |
| **Pre-conditions:** | **JS01\_PRE\_1**. User is in website JS | | |
| **Post-conditions:** | Full name, password, email, phone number, address will be added to database | | |
| **Normal Flow:** | **JS01\_NF\_1.** User clicks on registration in the header of JS website  **JS01\_NF\_2**. Website display form registration: full name, email, password, re-password, phone, submit button  **JS01\_NF\_3.** User fills up, clicks on submit button  **JS01\_NF\_4.** Website validates email, passwords and show “Registration successful” | | |
| **Alternative Flows:** | **JS01\_AT\_1.** At step JS01\_NF\_4, if email is existed in database, show error “email if existed, enter another email”  **JS01\_AT\_2.** At step JS01\_NF\_3, if re-password is different password, ask user enter re-password again | | |
| **Exceptions:** | N/A | | |
| **Priority:** | Normal | | |
| **Other information:** | Email must be valid email  Password is greater or equal 8 character  User can discard value entered by click [Cancel] button and [Back] instead of [OK] button. | | |
| **Frequency of Use:** | Usually | | |
| **Business Rules:** | N/A | | |
| **Assumptions:** | N/A | | |



### 

### Login

| Use Case ID: | JS02 | | |
| --- | --- | --- | --- |
| **Use Case Name:** | Login for user | | |
| **Created By:** | DucTM | **Last Updated By:** | DucTM |
| **Date Created:** | Oct 8, 2021 | **Date Last Updated:** | Oct 8, 2021 |
| **Actors:** | User | | |
| **Triggers** | User clicks on Login on the header of website | | |
| **Description:** | Registration for a new user | | |
| **Pre-conditions:** | **JS02\_PRE\_1**. User is in website JS | | |
| **Post-conditions:** | A connection to database is initialized. Update number of people access in database | | |
| **Normal Flow:** | **JS02\_NF\_1.** User clicks on login in the header of JS website  **JS02\_NF\_2**. Website display form registration: email, password, submit button  **JS02\_NF\_3.** User fills up, clicks on submit button  **JS02\_NF\_4.** Website validates username, email, passwords and show “Login successful”  **JS02\_NF\_5.** If user forgot password, click Forget password and fill up mail. | | |
| **Alternative Flows:** | **JS02\_AF\_1.** At step JS02\_NF\_4, if username is existed but password is not correct in database, show error: “Login information is not invalid”  **JS02\_AF\_2.** At step JS02\_NF\_4, if username is not found in database, show error: “Login information is not invalid” | | |
| **Exceptions:** | N/A | | |
| **Priority:** | Normal | | |
| **Other information:** | N/A | | |
| **Frequency of Use:** | Usually | | |
| **Business Rules:** | N/A | | |
| **Assumptions:** | N/A | | |

****

****

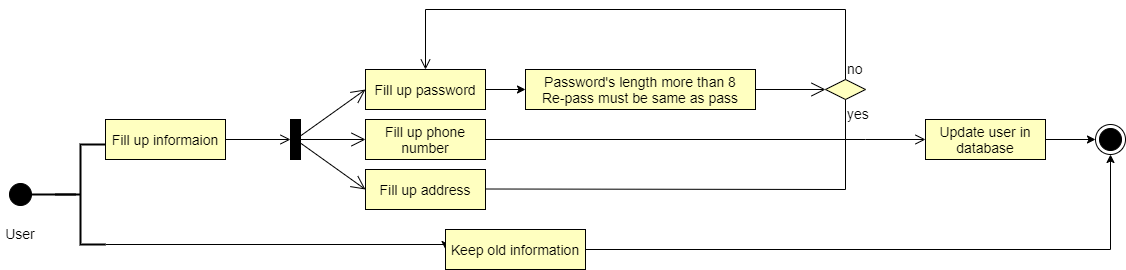
### Log out

| Use Case ID: | JS03 | | |
| --- | --- | --- | --- |
| **Use Case Name:** | Log out for user | | |
| **Created By:** | DucTM | **Last Updated By:** | DucTM |
| **Date Created:** | Oct 8, 2021 | **Date Last Updated:** | Oct 8, 2021 |
| **Actors:** | User | | |
| **Triggers** | User clicks on log out after click on their icon on the header of website | | |
| **Description:** | Log out of the website | | |
| **Pre-conditions:** | **JS03\_PRE\_1**. User is login website JS  **JS03\_PRE\_2.** User clicks on their avatar in header of website  **JS03\_PRE\_3.** User clicks on log out | | |
| **Post-conditions:** |  | | |
| **Normal Flow:** | **JS03\_NF\_1.** User clicks on their avatar in header of website  **JS03\_NF\_2**. User clicks on log out  **JS03\_NF\_3.** Website return to the default page. | | |
| **Alternative Flows:** | N/A | | |
| **Exceptions:** | N/A | | |
| **Priority:** | Normal | | |
| **Other information:** | N/A | | |
| **Frequency of Use:** | Usually | | |
| **Business Rules:** | N/A | | |
| **Assumptions:** | N/A | | |

### 

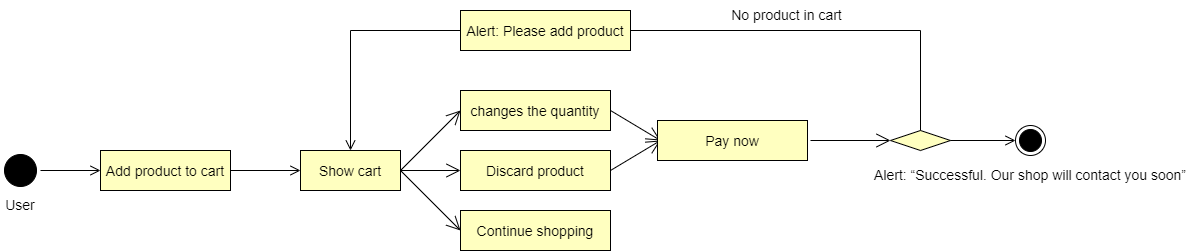
### Manage Profile

| Use Case ID: | JS04 | | |
| --- | --- | --- | --- |
| **Use Case Name:** | Manage Profile | | |
| **Created By:** | DucTM | **Last Updated By:** | DucTM |
| **Date Created:** | Oct 10, 2021 | **Date Last Updated:** | Oct 10, 2021 |
| **Actors:** | User | | |
| **Triggers** | User clicks on manage profile after click on their icon on the header of website | | |
| **Description:** | User changes information: password, phone number, address. | | |
| **Pre-conditions:** | **JS04\_PRE\_1**. User is login website JS  **JS04\_PRE\_2.** User clicks on their avatar in header of website  **JS04\_PRE\_3.** User clicks on manage profile | | |
| **Post-conditions:** | Information of user may be changed and update again to database | | |
| **Normal Flow:** | **JS04\_NF\_1.** User fills up text box password and re-password if want to change password  **JS04\_NF\_2.** User fills up text box phone number if want to change phone number  **JS04\_NF\_3.** User fills up text box address if want to change address  **JS04\_NF\_4.** User click submit button | | |
| **Alternative Flows:** | **JS04\_AF\_1.** At step JS04\_NF\_1, if password is not the same as re-password, alert to user: “Re-password must be the same as password”, user cannot submit request change information  **JS04\_AF\_1.** At step JS04\_NF1\_1, if password is not strong enough, alert to user: “Password is weak”, user cannot submit request change information | | |
| **Exceptions:** | N/A | | |
| **Priority:** | Normal | | |
| **Other information:** | Email must be valid email  Password is greater or equal 8 character  User can discard value entered by click [Cancel] button and [Back] instead of [OK] button. | | |
| **Frequency of Use:** | Usually | | |
| **Business Rules:** | N/A | | |
| **Assumptions:** | N/A | | |



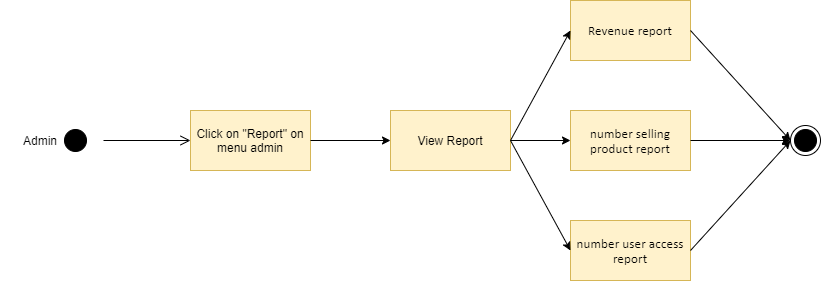
### Manage Shopping

| Use Case ID: | JS05 | | |
| --- | --- | --- | --- |
| **Use Case Name:** | Manage Shopping | | |
| **Created By:** | DucTM | **Last Updated By:** | DucTM |
| **Date Created:** | Oct 10, 2021 | **Date Last Updated:** | Oct 10, 2021 |
| **Actors:** | User | | |
| **Triggers** | User clicks on pay a product, click on cart and click on pay cart | | |
| **Description:** | User add a product to cart | | |
| **Pre-conditions:** | **JS05\_PRE\_1**. User is login website JS  **JS05\_PRE\_2.** User clicks on product their want to buy  **JS05\_PRE\_3.** User clicks on add to cart | | |
| **Post-conditions:** | A cart with product is create and add to database, a request pay cart send to admin | | |
| **Normal Flow:** | **JS05\_NF\_1.** User changes quantity of a product in the box cart display.  **JS05\_NF\_2.** User click on “x” if they want to discard this product.  **JS05\_NF\_3.** User click on Continue shopping if want to buy more  **JS05\_NF\_4.** User click on Pay if want to pay now  **JS05\_NF\_5**. User clicks on pay, message alert: “Successful. Our shop will contact you soon” | | |
| **Alternative Flows:** | **JS05\_AF\_1.** At step JS05\_NF\_1, quantity must be a number greater than 0  **JS05\_AF\_2**. Before step JS05\_NF\_5, must be at least 1 product | | |
| **Exceptions:** | N/A | | |
| **Priority:** | Normal | | |
| **Other information:** | User can discard value entered by click [Cancel] button and [Back] instead of [OK] button. | | |
| **Frequency of Use:** | Usually | | |
| **Business Rules:** | N/A | | |
| **Assumptions:** | N/A | | |



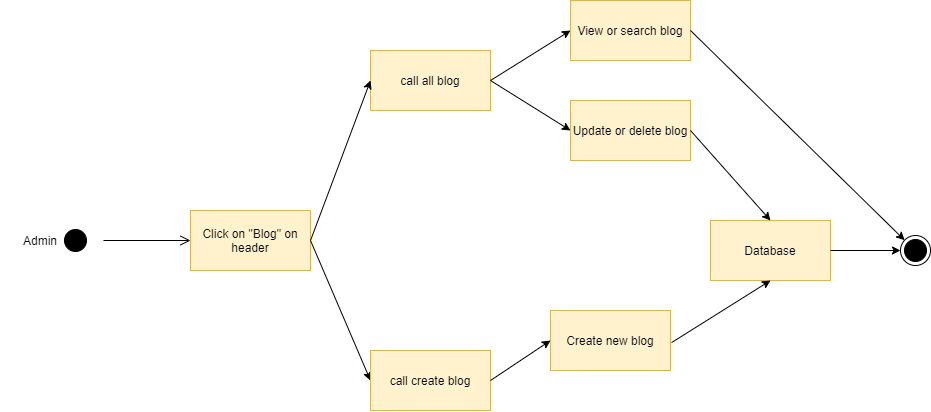
### Manage Report

| Use Case ID: | JS06 | | |
| --- | --- | --- | --- |
| **Use Case Name:** | Manage Report | | |
| **Created By:** | DucTM | **Last Updated By:** | DucTM |
| **Date Created:** | Oct 10, 2021 | **Date Last Updated:** | Oct 10, 2021 |
| **Actors:** | Admin | | |
| **Triggers** | Admin click on manage report on menu | | |
| **Description:** | User’s report may be resolved/rejected | | |
| **Pre-conditions:** | **JS06\_PRE\_1**. Admin is login website JS by admin account | | |
| **Post-conditions:** | Revenue report, number selling product report, number user access report will be shown by chart though each month. | | |
| **Normal Flow:** | N/A | | |
| **Alternative Flows:** | N/A | | |
| **Exceptions:** | N/A | | |
| **Priority:** | Normal | | |
| **Other information:** | N/A | | |
| **Frequency of Use:** | Usually | | |
| **Business Rules:** | N/A | | |
| **Assumptions:** | N/A | | |



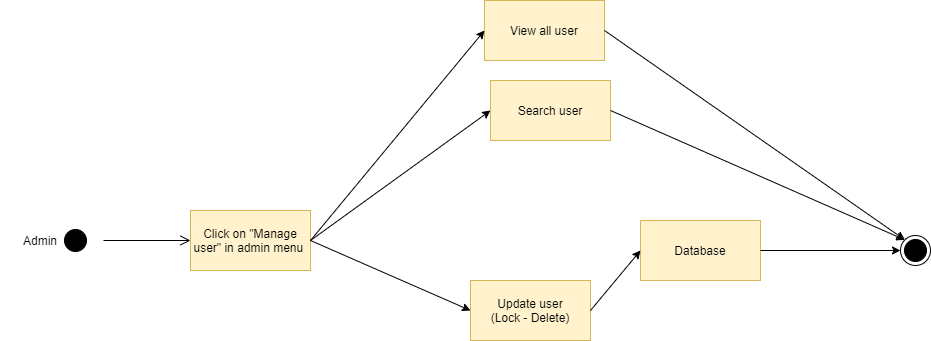
### Manage Blog

| Use Case ID: | JS07 | | |
| --- | --- | --- | --- |
| **Use Case Name:** | Manage Blog | | |
| **Created By:** | LocBN | **Last Updated By:** | LocBN |
| **Date Created:** | Oct 10, 2021 | **Date Last Updated:** | Oct 10, 2021 |
| **Actors:** | Admin | | |
| **Triggers** | Admin click on “Blog” in homepage | | |
| **Description:** | Admin manage articles on your website | | |
| **Pre-conditions:** | **JS08\_PRE\_1**. Admin is login website JS by admin account | | |
| **Post-conditions:** | Add , delete the featured article related to diamond field : price , trend , .... | | |
| **Normal Flow:** | N/A | | |
| **Alternative Flows:** | N/A | | |
| **Exceptions:** | N/A | | |
| **Priority:** | Normal | | |
| **Other information:** | N/A | | |
| **Frequency of Use:** | Usually | | |
| **Business Rules:** | N/A | | |
| **Assumptions:** | N/A | | |



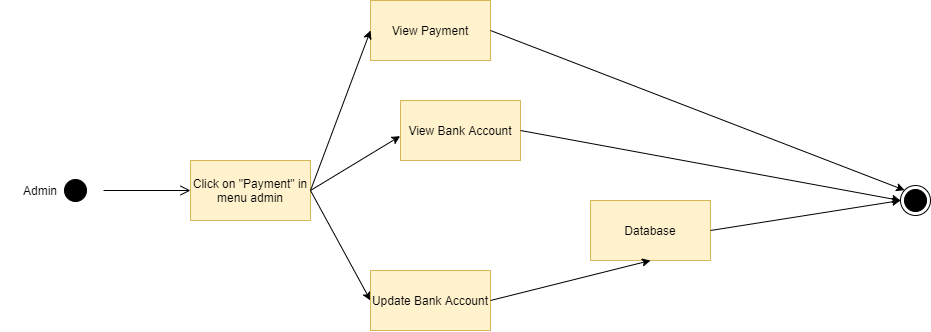
### Manage User

| Use Case ID: | JS08 | | |
| --- | --- | --- | --- |
| **Use Case Name:** | Manage User | | |
| **Created By:** | DucTM | **Last Updated By:** | DucTM |
| **Date Created:** | Oct 10, 2021 | **Date Last Updated:** | Oct 10, 2021 |
| **Actors:** | Admin | | |
| **Triggers** | Admin click on manage user on menu | | |
| **Description:** | Admin can see information of all users | | |
| **Pre-conditions:** | **JS08\_PRE\_1**. Admin is login website JS by admin account  **JS08\_PRE\_2**. Admin is click on manage user on menu | | |
| **Post-conditions:** | List of users is shown. | | |
| **Normal Flow:** | **JS08\_NF\_1.** Admin clicks on manage user | | |
| **Alternative Flows:** | N/A | | |
| **Exceptions:** | N/A | | |
| **Priority:** | Normal | | |
| **Other information:** | N/A | | |
| **Frequency of Use:** | Usually | | |
| **Business Rules:** | N/A | | |
| **Assumptions:** | N/A | | |



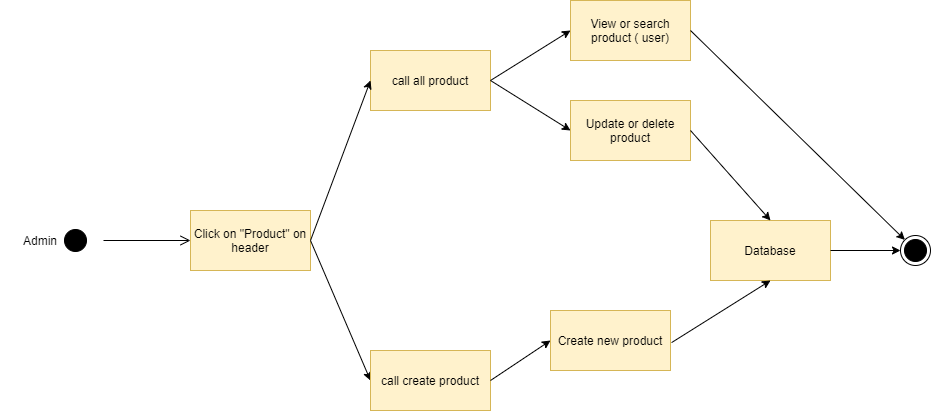
### Manage Payment

| Use Case ID: | JS09 | | |
| --- | --- | --- | --- |
| **Use Case Name:** | Manage Payment | | |
| **Created By:** | DucTM | **Last Updated By:** | DucTM |
| **Date Created:** | Oct 10, 2021 | **Date Last Updated:** | Oct 10, 2021 |
| **Actors:** | User, Admin | | |
| **Triggers** | Admin want to add bank accounts or payments.  User want to pay for each order. | | |
| **Description:** | Everyone accesses the JS , manages payment information, admin can add bank accounts and create payments. | | |
| **Pre-conditions:** | **JS09\_PRE\_1**. Everyone are logged into JS.  **JS09\_PRE\_1**. Admin click on link “Bank accounts.” in footer | | |
| **Post-conditions:** | User and Admin are perform to exchange the money. | | |
| **Normal Flow:** | **JS09\_NF\_1.** Everyone are logged into JS Shop  **JS09\_NF\_2.** Admin can add bank account or payment  **JS09\_NF\_3.** Admin fill up information.  **JS09\_NF\_4.** Admin click “Add” button  **JS09\_NF\_5.** JS displays add bank account/payment successfully. **JS09\_NF\_6.** User and Admin are perform to exchange the money | | |
| **Alternative Flows:** | **JS04\_AF\_1.** At step JS04\_NF\_1, All data fields cannot be left blank and cannot have special characters, the payment method to use the momo e-wallet must be Admin’s phone number | | |
| **Exceptions:** | No complain from user  Flash informs admin that no complain are available for the date | | |
| **Priority:** | Low | | |
| **Other information:** | Users transfer money through the system. After the service package is completed, the system will return the money to Admin. | | |
| **Frequency of Use:** | Usually | | |
| **Business Rules:** | N/A | | |
| **Assumptions:** | N/A | | |



### Manage Product

| Use Case ID: | JS10 | | |
| --- | --- | --- | --- |
| **Use Case Name:** | Manage Product | | |
| **Created By:** | LocBN | **Last Updated By:** | LocBN |
| **Date Created:** | Oct 10, 2021 | **Date Last Updated:** | Oct 10, 2021 |
| **Actors:** | Admin | | |
| **Triggers** | Admin click on “Product” in homepage | | |
| **Description:** | Admin manage product of JS | | |
| **Pre-conditions:** | **JS11\_PRE\_1**. Admin is login website JS by admin account  **JS11\_PRE\_2**. Admin is click on “Product” in homepage | | |
| **Post-conditions:** | Product will be updated in database | | |
| **Normal Flow:** | **JS11\_NF\_1.** Admin is logged into JS Shop  **JS11\_NF\_2**. Admin click on “Product ” in homepage  **JS11\_NF\_3**. Admin cansearch, add, edit, delete products | | |
| **Alternative Flows:** | N/A | | |
| **Exceptions:** | N/A | | |
| **Priority:** | Normal | | |
| **Other information:** | N/A | | |
| **Frequency of Use:** | Usually | | |
| **Business Rules:** | N/A | | |
| **Assumptions:** | N/A | | |

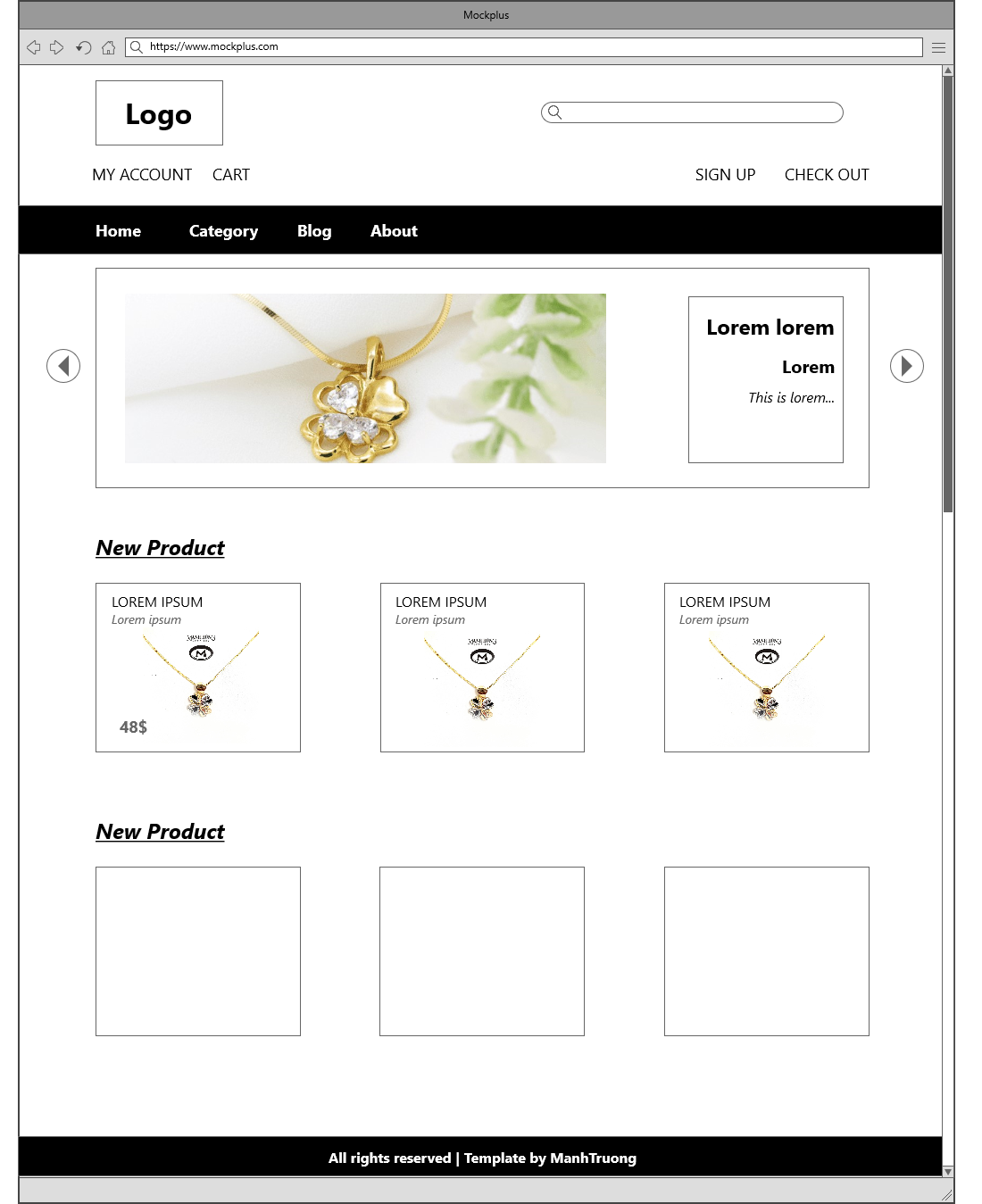


## Non-Functional Requirements Specification

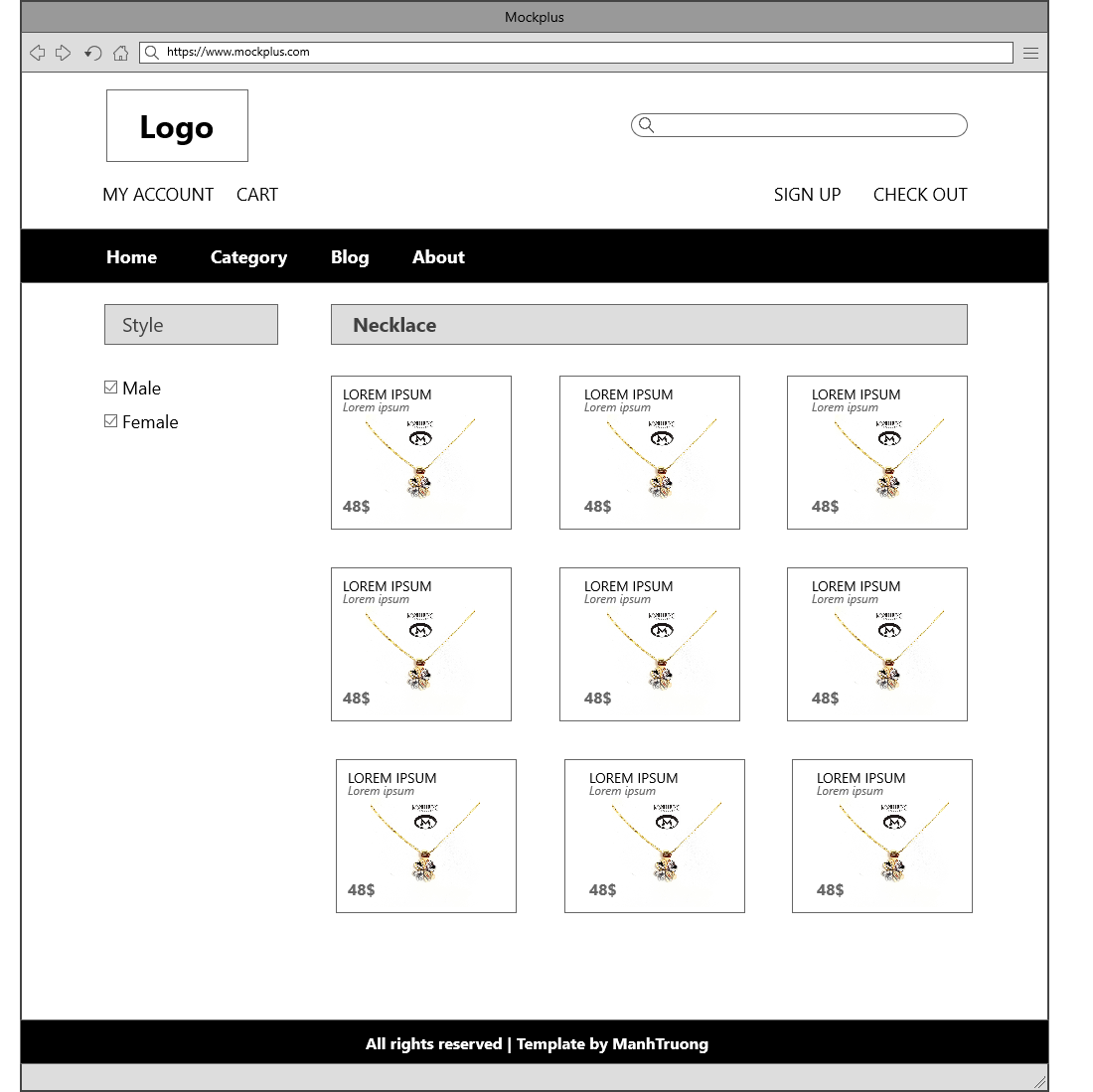
### External Interface Requirements

#### User Interfaces

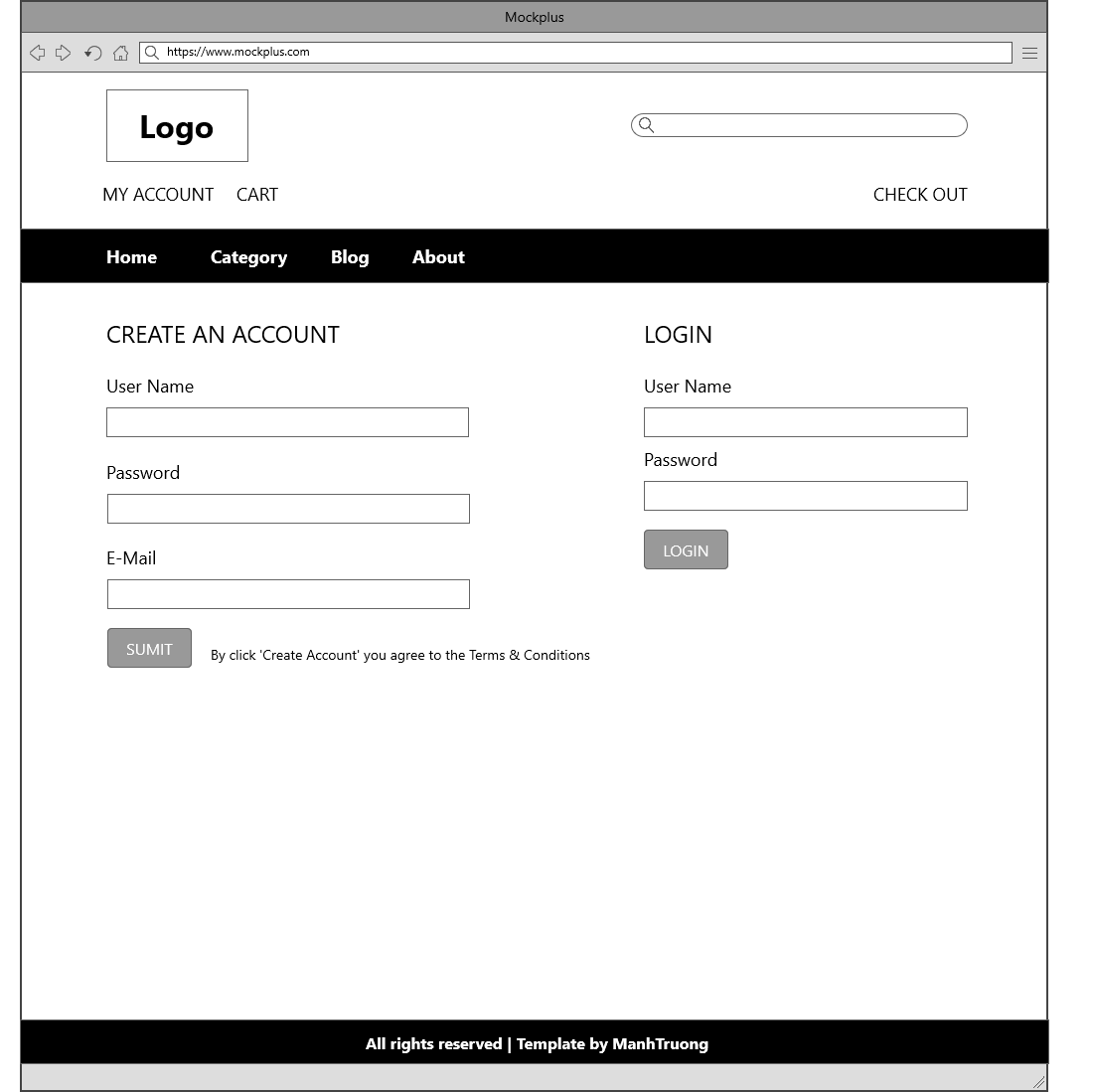
*Screen 1: user accesses JS website*

**

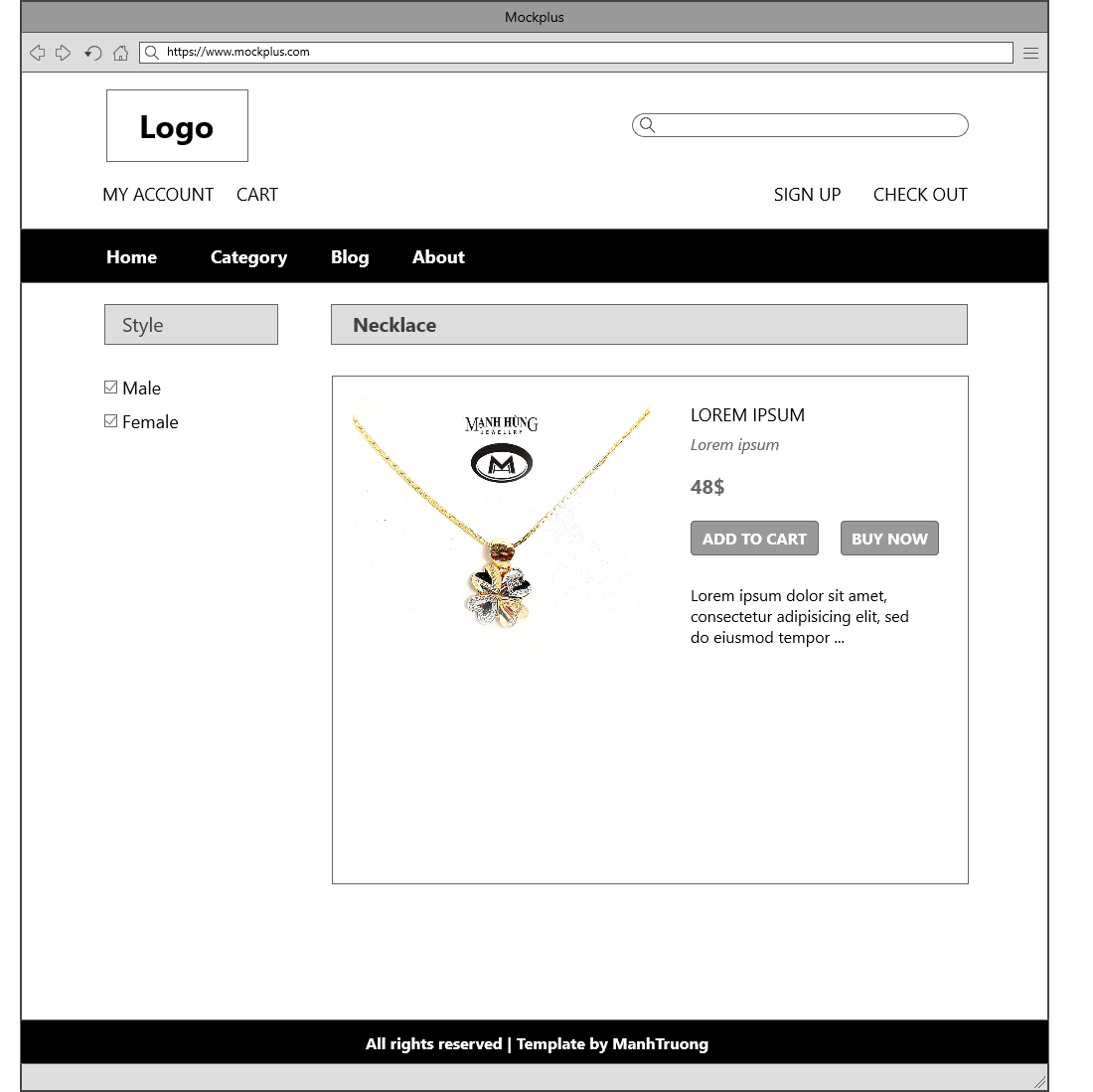
*Screen 2: user clicks on product type*

**

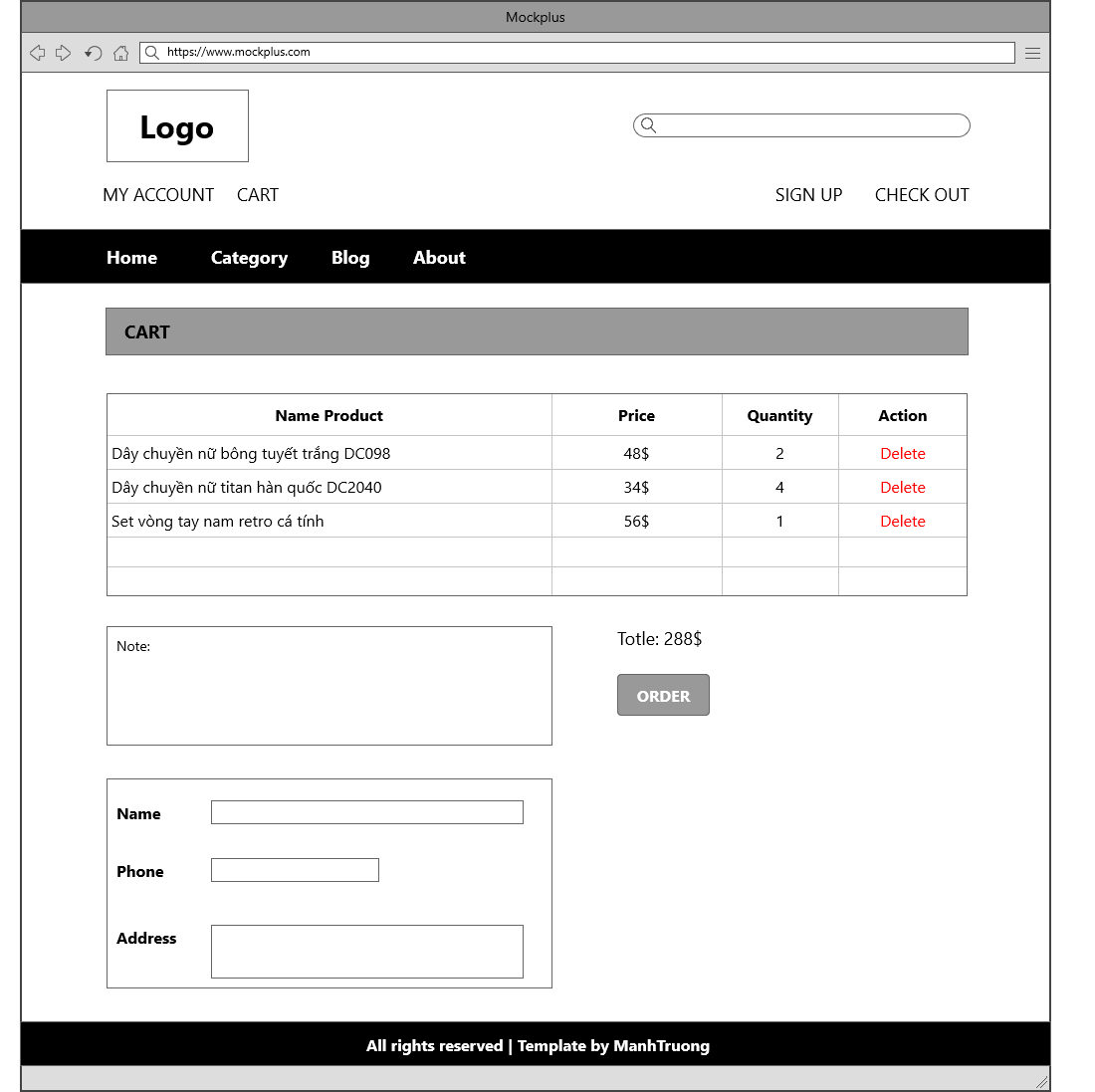
*Screen 3: user clicks on login & create account*

**

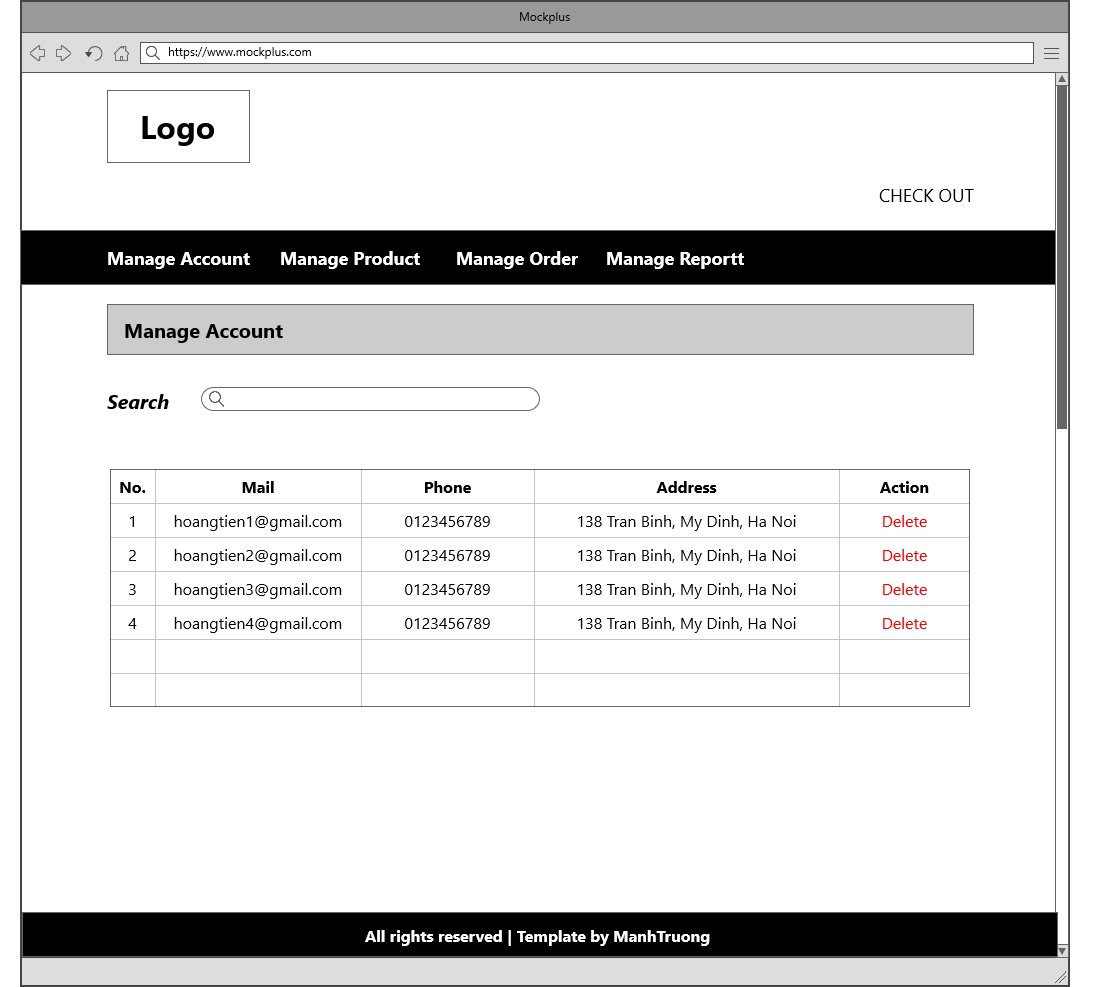
*Screen 4: user clicks on a product*

**

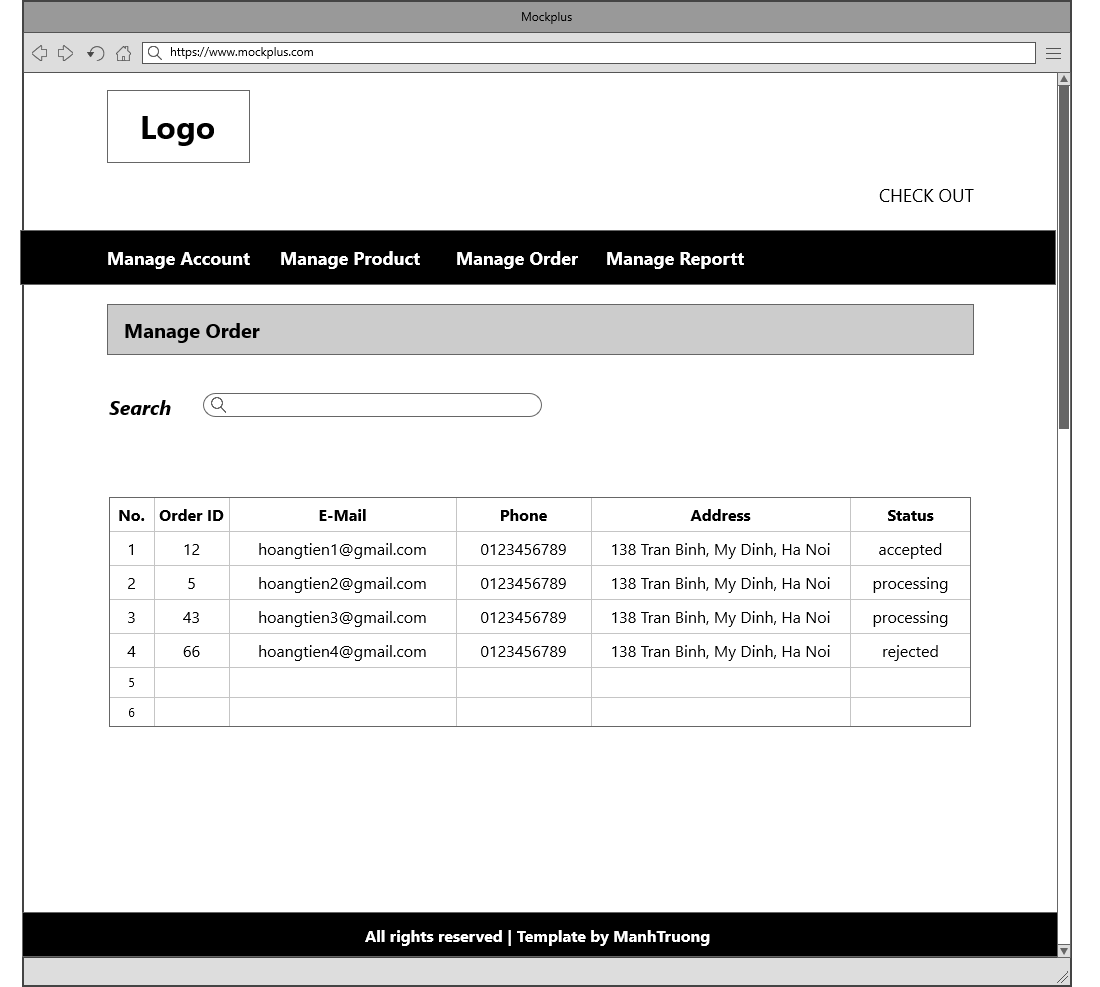
*Screen 5: user clicks on cart*

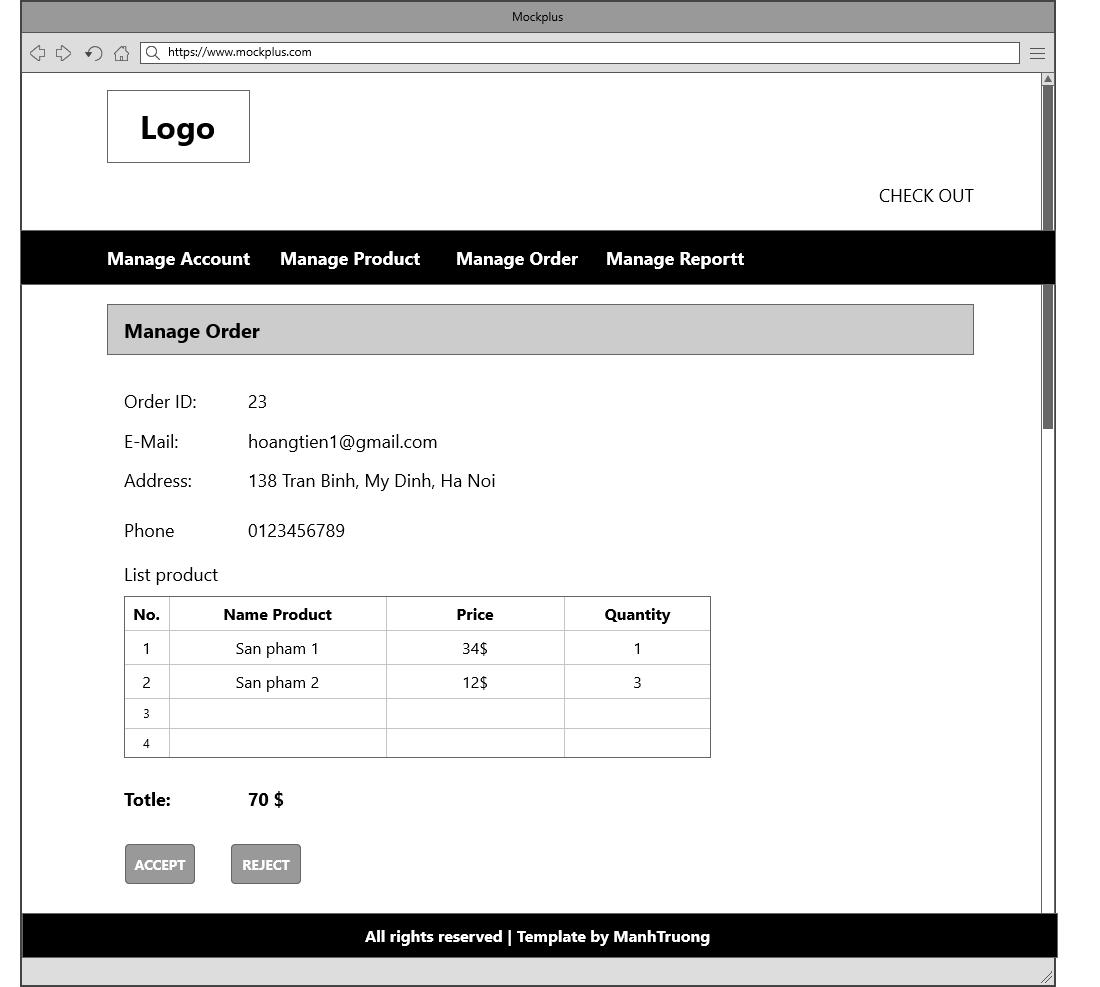
**

*Screen 6: admin clicks on manage account*

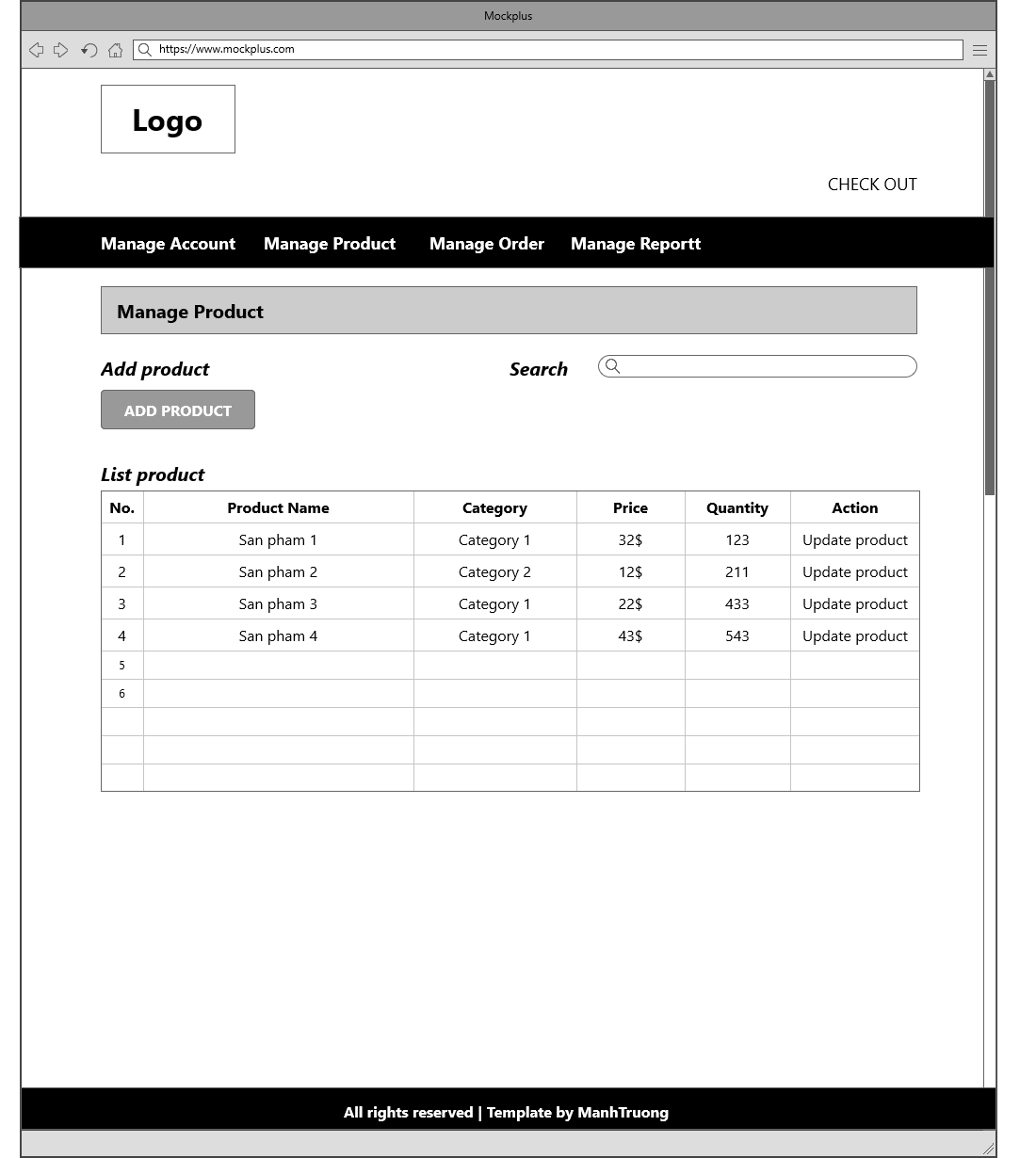
**

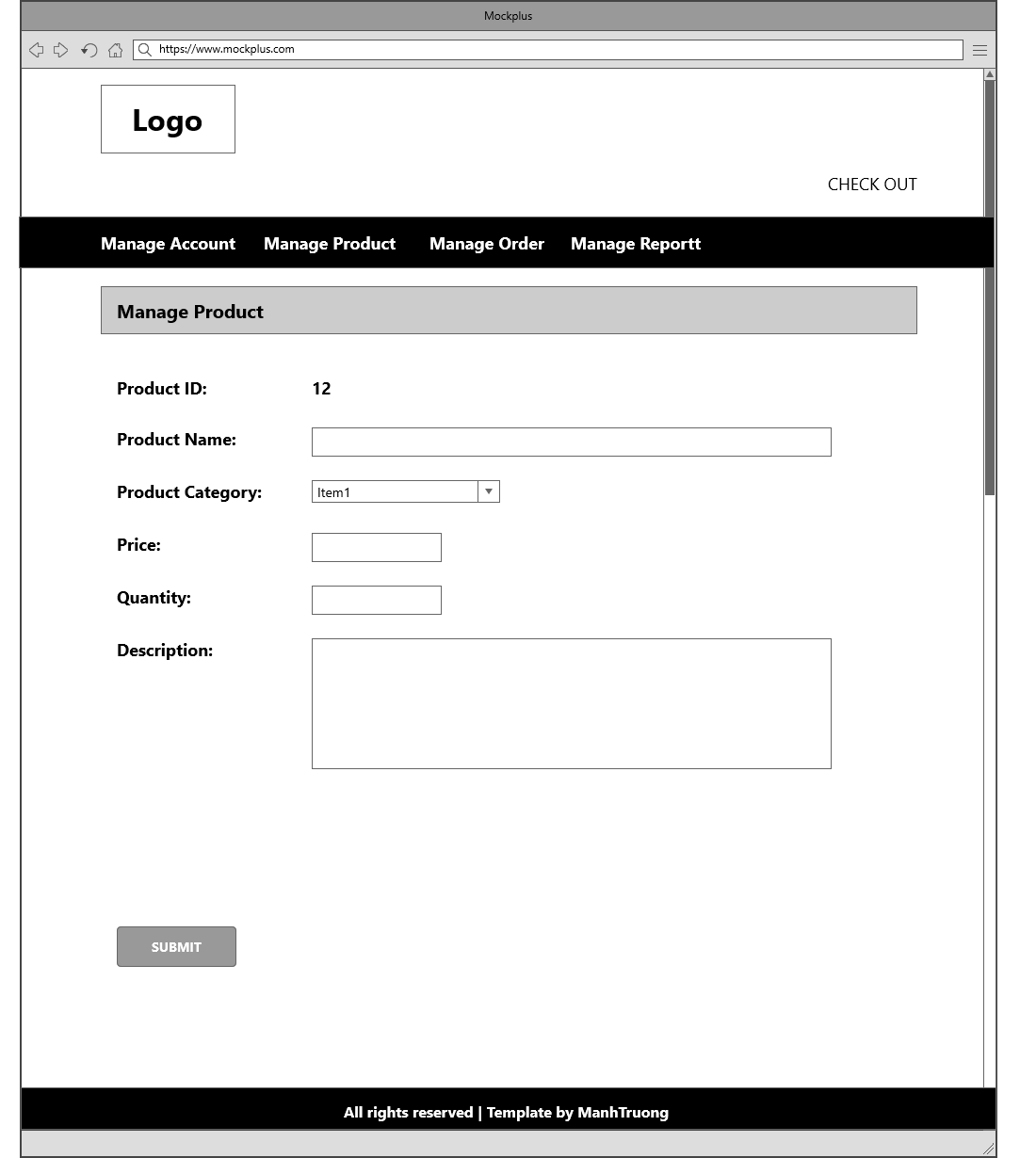
*Screen 7: admin clicks on manage order*

**

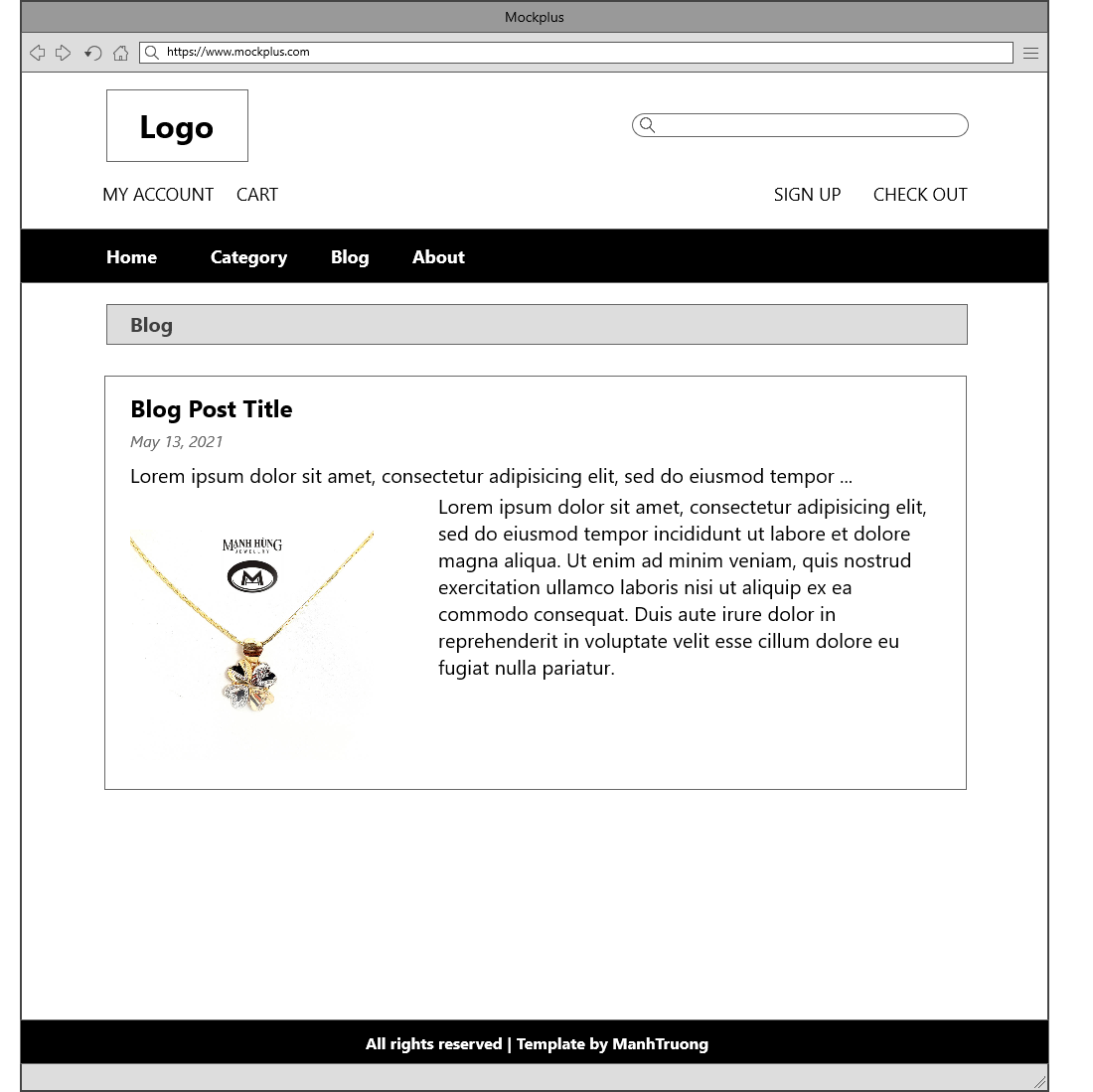
**

*Screen 8: Admin clicks on manage product*

**

**

*Screen 8: Blog*

**

#### Hardware Interfaces

It strongly recommended that user use a desktop device instead of mobile device because some feature may be not work well in mobile in version 1.0 of JS website. Don’t forget that you must have an internet connection like Wan – Lan.

Recommend that user should use new version of Chrome, Firefox, Edge. Internet Explorer may be not fully supported this website.

#### Software Interfaces

As it has already been started, JS run on almost browser with internet on any OS on client side. With server side, all most functions depend on database. See information of database in Appendix.

#### Communications Interfaces

JS use HTTP for communication over the internet.

### Other Nonfunctional Requirements

#### Performance Requirements

First version 1.0 of JS website does not require powerful processor, more ram or need card graphic. It will run fast all most feature. But this version just used connection database with 5 database transaction units and 2GB memory from Azure DB. if there are over 500 connections to website, connection to database may be slowly. Speed and bandwidth of connection database should be upgrade in next version.

#### Design constraints

Coding design constraints uses Model-View-Controller pattern.

#### Safety Requirements

If developers are not careful, data in database can be loss. You must remember to save backup data to rollback if something wrong happened. JS website version 1.0 doesn’t have more feature for admin because if admin lost account administrator, it’s very difficult to rollback data when database is deleted.

#### Security Requirements

Admin never sees another account password. System back-end servers is only accessible by administrator, all database is encrypted.

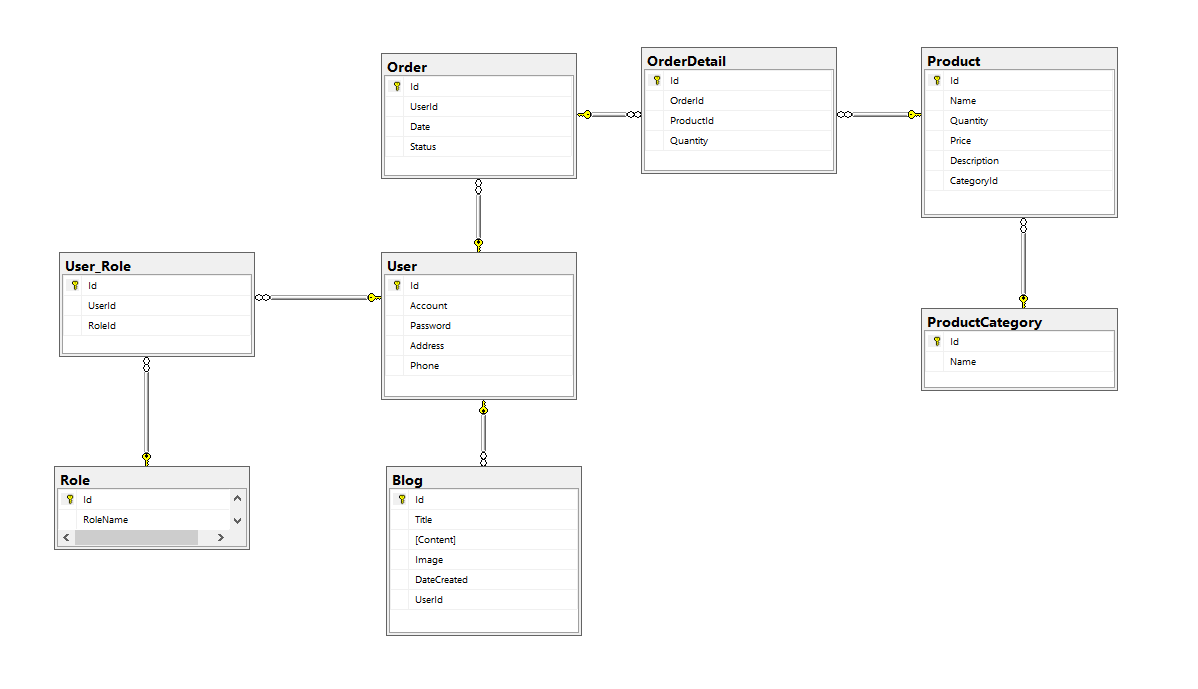
First version 1.0 is also initialization version so DDOS server may be happened. Delay time access will apply if DDOS happen to decrease the number of DDOS happens in the future. Next version must improve plan to defense out of DDOS.

#### Software Quality Attributes

JS website has quite easy GUI with all features can be access from homepage. User can easy accesses to a feature from header of website. All feedback can be sent to email on the footer of website.

1. **Other Requirements**

**4.1: Database diagram**

****

**4.2: Database entity**

1. User

| ***User*** | | |
| --- | --- | --- |
| Field | Data Type | Key |
| Id | INT | PK |
| Account | NVARCHAR(MAX) |  |
| Password | NVARCHAR(MAX) |  |
| Address | NVARCHAR(MAX) |  |
| Phone | NVARCHAR(MAX) |  |

1. Role

| ***Role*** | | |
| --- | --- | --- |
| Field | Data Type | Key |
| Id | INT | PK |
| RoleName | NVARCHAR(MAX) |  |

1. User\_Role

| ***User\_Role*** | | |
| --- | --- | --- |
| Field | Data Type | Key |
| Id | INT | PK |
| UserId | INT | FK |
| RoleId | INT | FK |

1. Blog

| ***Blog*** | | |
| --- | --- | --- |
| Field | Data Type | Key |
| Id | INT | PK |
| Title | NVARCHAR(MAX) |  |
| Content | NVARCHAR(MAX) |  |
| Image | NVARCHAR(MAX) |  |
| DateCreated | DATE |  |
| UserId | INT | FK |

1. Order

| ***Order*** | | |
| --- | --- | --- |
| Field | Data Type | Key |
| Id | INT | PK |
| UserId | INT | FK |
| Date | DATE |  |
| Status | NVARCHAR(MAX) |  |

1. OrderDetail

| ***OrderDetail*** | | |
| --- | --- | --- |
| Field | Data Type | Key |
| Id | INT | PK |
| OrderId | INT | FK |
| ProductId | INT | FK |
| Quantity | INT |  |

1. Product

| ***Product*** | | |
| --- | --- | --- |
| Field | Data Type | Key |
| Id | INT | PK |
| Name | NVARCHAR(MAX) |  |
| Quantity | INT |  |
| Price | NVARCHAR(MAX) |  |
| Description | NVARCHAR(MAX) |  |
| CategoryId | INT | FK |

1. ProductCategory

| ***ProductCategory*** | | |
| --- | --- | --- |
| Field | Data Type | Key |
| Id | INT | PK |
| Name | NVARCHAR(MAX) |  |