



# Software Requirements Specification

Version 1.0

August 13, 2021

**MASKM-Store**  
E-Commerce Solution

**M**ilton Gonzalez S#:612356  
**A**hmed Elghannam S#:612398  
**S**oufiane Ennali S#:612359  
**K**arim A R Aburejila S#:612403  
**M**aher Samoudi S#:612382

Supervised By Prof.  
Muhyieddin Al-Tarawneh

Compro Master Program  
WAA

## Table of Contents

Table of Contents .....	i
1.0. Introduction .....	2
1.1. Purpose .....	2
1.2. Scope of Project.....	2
1.3. Glossary.....	2
1.4. Overview of Document .....	2
2.0. Overall Description .....	3
2.1 System Environment.....	3
2.2 Functional Requirements Specification .....	4
2.2.1 Register Buyer .....	4
Use case: Register A Buyer .....	5
2.2.2 Buyer Use Case .....	5
Use case: add review for the product.....	5
2.2.3 Buyer Use Case .....	6
Use case: Place Order/edit/delete/retrieve .....	6
2.2.4 Buyer Use Case .....	7
Use case: Update Order Status (Delivered/Returned) .....	7
2.2.5 Buyer Use Case .....	8
Use case: Follow/Un-follow Seller.....	8
2.2.6 Seller Use Case.....	9
Use case: Register a Seller.....	10
2.2.7 Seller Use Case.....	10
Use case: register a product for approval.....	10
2.2.8 Seller Use Case.....	11
Use case: Update order status Shipped / Cancelled.....	11
2.2.9 Admin Use Case .....	11
Use case: Register as admin .....	11
2.2.10 Admin Use Case .....	12
Use case: Approve Review.....	12
2.2.11 Admin Use Case .....	13
Use case: Approve Seller.....	13
2.3 User Characteristics .....	13
2.4 Non-Functional Requirements .....	14
3.0. Requirements Specification .....	14
3.1 features .....	14



## **1.0. Introduction**

### ***1.1. Purpose***

The purpose of this document is to present a detailed description of the MASKM-STORE E-Commerce Solution. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the stakeholders and the developers of the system.

### ***1.2. Scope of Project***

MASKM-STORE E-Commerce Solution is an E-Commerce Solution for intended for educational purpose for Web Application Architecture in the Compro Master Program at Maharishi International University. This system intended to automate the buying and selling process by creating an online Ecommerce platform.

More specifically, the system will serve 3 main stakeholders:

- 1- Seller who will create his products on the system and can advertise his product by requesting the admin to make the product *Featured* which will allow these product to be presented in the main page .
- 2- Buyer place orders and add his payment details to allow him to do the payment (payment feature is not in this scope and can be integrated in future version with a 3<sup>rd</sup> party payment gateway)
- 3- Admin who will approve the sellers , and buyer's reviews . admins can also add category (in future admin can have more functionalities )

### *1.3. Glossary*

<b>Term</b>	<b>Definition</b>
Buyer	Any user who register in the system as a buyer and want to purchase products from the system
Seller	Any user who register as seller to be a supplier of goods or products in the system
Admin	The user who register with admin role to manage the system
Backend	The system use spring-boat project on top of spring framework to handle the model, business and persistence layer
Frontend	The system uses React to handle the presentation layer and GUI
GUI	Graphical User Interface
Repository	Persistence layer
Model/Domain	Have the definition and attributes of all domains
Service	Business Layer
Controller	Is used to route the get, post, put and delete mapping
DTO	Is user to be the intermediate layer between the domain objects and the presentation layer
Authentication	Is where the system define the roles of the users in the system

### *1.4. Overview of Document*

The next chapter, the Overall Description section, of this document gives an overview of the functionality of the product. It describes the informal requirements and is used to establish a context for the technical requirements specification in the next chapter.

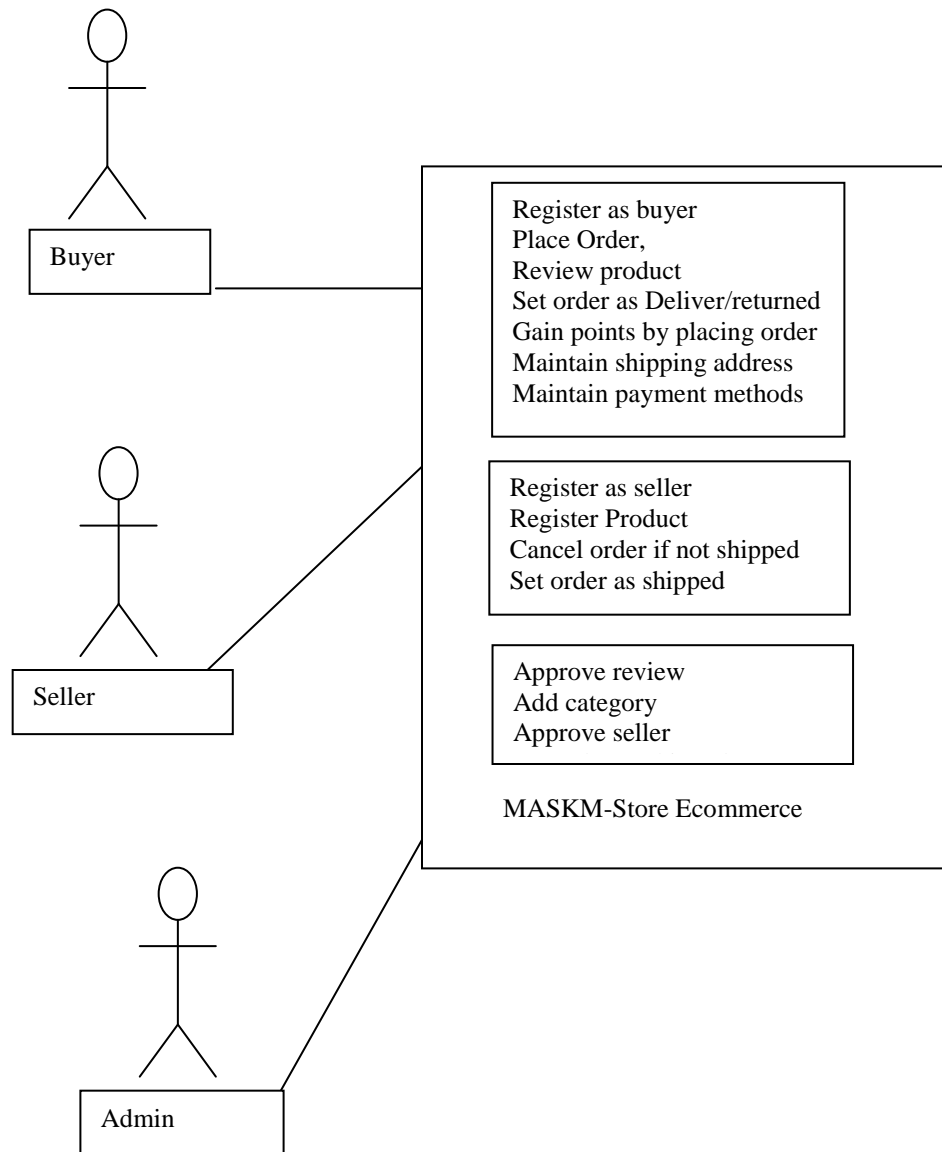
The third chapter, Requirements Specification section, of this document is written primarily for the developers and describes in technical terms the details of the functionality of the product.

Both sections of the document describe the same software product in its entirety, but are intended for different audiences and thus use different language.

## 2.0. Overall Description

### 2.1 System Environment

Figure 1 - System Environment



The MASKM-STORE E-Commerce Solution has three active actors.



The Seller, Buyer, or Admin accesses the online store. Seller and Admin interact in the system to make admin approves the seller. The Buyer accesses the system to place order and set a review for a product.

## 2.2 Functional Requirements Specification

This section outlines the use cases for each of the active Buyers separately. The Buyer, the Seller and the Admin have only one use case apiece while the editor is main actor in this system.

2.2.0 Buyer Dash Board system will have a buyer Dash Board in which he can see his points his follow sellers and recent orders he placed

order id	date	status	
100	2021-08-13	Shipped	
116	2021-08-13	Pending	
120	2021-08-13	Pending	

### 2.2.1 Register Buyer



### LOGIN

Username

Password

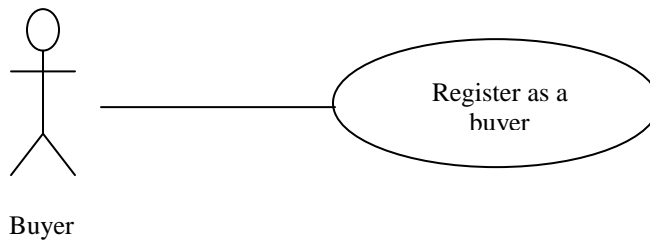
### NEW CUSTOMER

CREATE A ACCOUNT  

Sign up for a free account at our store. Registration is quick and easy. It allows you to be able to order from our shop. To start shopping click register.

Use case: **Register A Buyer**

**Diagram:**



### Brief Description

The Buyer register in **MASKM**-Store and add his details .

### Initial Step-By-Step Description

Before this use case can be initiated, the Buyer has already accessed the buyer section of the **MASKM**-Store.

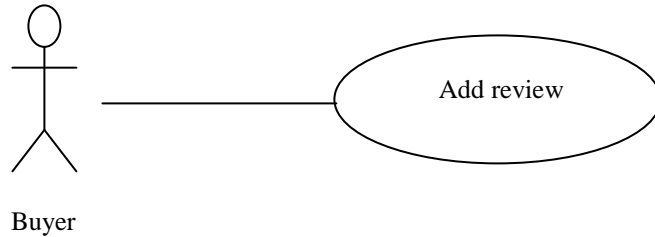
The Buyer

- 1- Add his personal details (first name and last name, username and password)
- 2- Shipping Address
- 3- Billing Address
- 4- Email will be used to get email notification after placing order
- 5- Payment Methods etc...

### 2.2.2 Buyer Use Case

Use case: **add review for the product**

**Diagram:**



**Brief Description**

The Buyer accesses the **MASKM**-Store and add review for a product.

**Initial Step-By-Step Description**

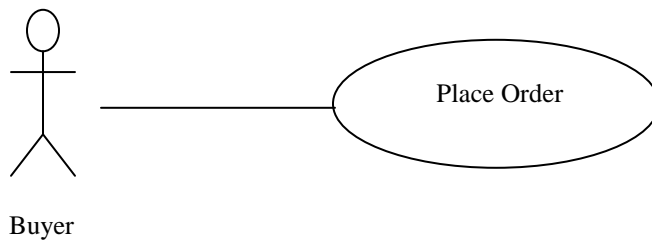
Before this use case can be initiated, the Buyer has already accessed the buyer section of the **MASKM**-Store .

- 1- The Buyer chooses a product
- 2- Add a review for this product.
- 3- Added review will be pending approval from admin

2.2.3 Buyer Use Case

Use case: **Place Order/edit/delete/retrieve**

**Diagram:**



**Brief Description**

The Buyer accesses the **MASKM**-Store and add review for a product.

**Initial Step-By-Step Description**

Before this use case can be initiated, the Buyer has already accessed the buyer section of the **MASKM**-Store.

- 1- The Buyer place order
- 2- By default order will take order creation date as system date
- 3- Order will have a default status Pending
- 4- The Buyer gets points for each order placed
- 5- Buyer can change the order status to delivered when received





- 6- Buyer can change the order status to returned and when it is returned it he will lose the points he gain when he placed that order
- 7- Buyer also can review update, and delete his order.

Welcome to Our store MASKM ☎ Call Us: 123 - 456 - 7890 Hello: samer

CATEGORIES ▾ PAGES ▾

---

CART HOME / CART

IMAGE	PRODUCT NAME	PRICE	QUANTITY	ACTION	TOTAL
	Shoes Nike	\$250	1	✕	\$250

Total Price: **\$ 250**

[CONTINUE SHOPPING](#)
[CHECK OUT](#)

---

### Billing Details

First Name

Last Name

Phone

Email Address

Country

Address

Town/City

State / County

Postal Code

Product	Total
Shoes Nike x 1	\$250
<hr/>	
Subtotal	\$250
Shipping	<input type="checkbox"/> Free Shipping <input type="checkbox"/> Local Pickup
<hr/>	
<b>Total</b>	<b>\$250</b>

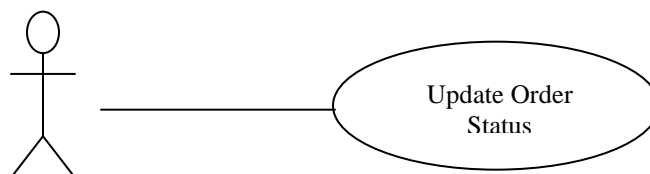
☒ Stripe  
☐ PayPal

[PLACE ORDER](#)

## 2.2.4 Buyer Use Case

Use case: **Update Order Status (Delivered/Returned)**

**Diagram:**



### Brief Description

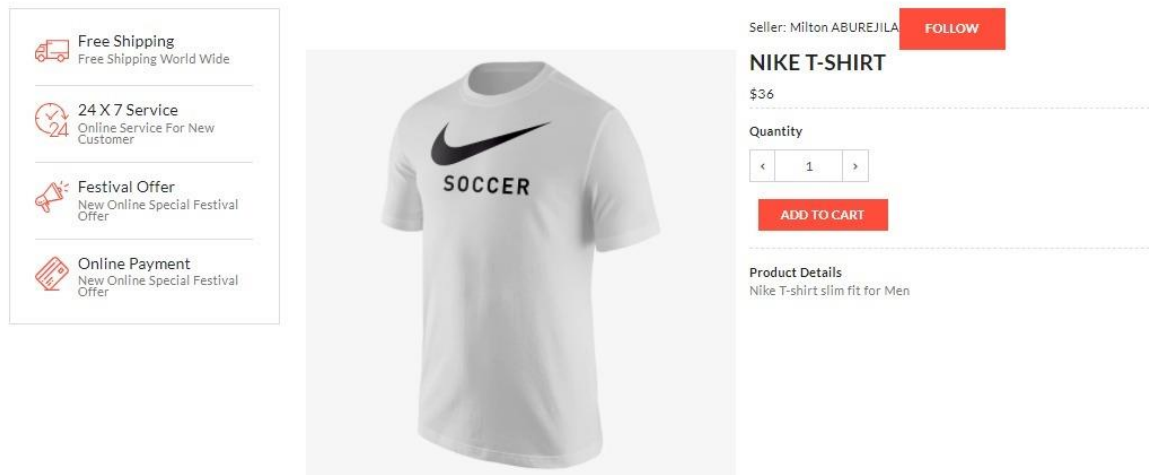
The Buyer login to accesses the **MASKM**-Store update the order status to Delivered when received or Returned in case he want to return an order .

### Initial Step-By-Step Description

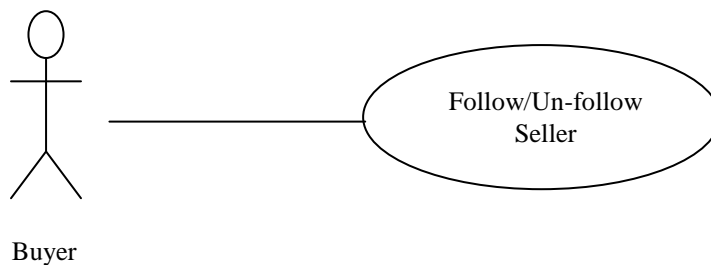
Before this use case can be initiated, the Buyer has already accessed the buyer section of the **MASKM**-Store.

- 1- The Buyer can update the order status to delivered when he received the order
- 2- The buyer also can change the order status to returned and when it is returned he lose the points he gains from this order.

### 2.2.5 Buyer Use Case



Use case: **Follow/Un-follow Seller**  
Diagram:



### Brief Description

The Buyer need to login and can follow and un-follow seller and the sellers will be added to the buyer's followed list.

### Initial Step-By-Step Description

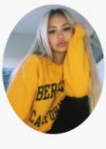
Before this use case can be initiated, the Buyer has already accessed the buyer section of the **MASKM**-Store.

- 1- The Buyer can follow a seller
- 2- The Buyer can un- follow a seller

### 2.2.6.0 Seller Dash Board

SELLER DASHBOARD

HOME / SELLER DASHBOARD



**Milton**  
Milton@gmail.com

- Dashboard
- Products
- Order
- Logout

6 Total Products

13 Order Pending

Trending Products

Image	Product Name	Price	Category
	Nike T-Shirt2	36	Clothes
	Shoes Nike	250	Clothes
	Socks	500	Clothes


Recent Orders

Order Id	Creation Date	Status
102	2021-08-13	Cancelled
107	2021-08-13	Cancelled
110	2021-08-13	Cancelled
113	2021-08-13	Pending
122	2021-08-13	Pending

### 2.2.6.0 Seller List All seller's product

SELLER DASHBOARD

HOME / SELLER DASHBOARD



**Vintage**  
FASHION STORE

- Dashboard
- Products
- Order
- Logout

All Products

Add Product

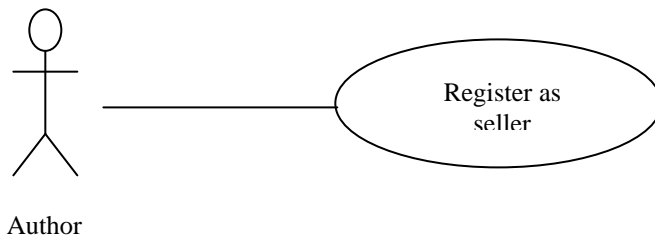
Image	Product Name	Category	Price	Action
	Nike T-Shirt	Clothes	36	
	Shoes Nike	Clothes	250	
	Socks	Clothes	500	
	Shirt	Clothes	50	
	T-Shirt L	Clothes	10	
	Suite	Clothes	1800	

### 2.2.6 Seller Use Case

In case of new seller he needs to register as a seller in the **MASKM**-Store.

Use case: Register a Seller

**Diagram:**



### **Brief Description**

The Seller register as seller/supplier.

### **Initial Step-By-Step Description**

Before accessing the functionality of the system seller has to register as a Seller in the **MASKM-Store**.

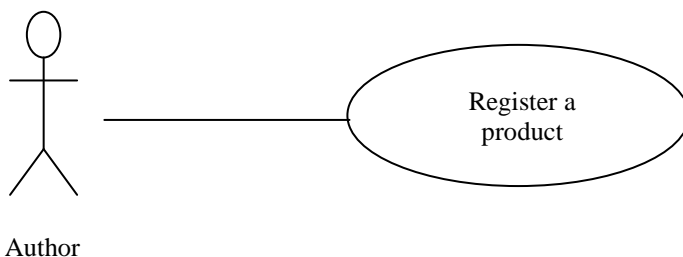
1. The Seller register as seller to be able to submit his products

### **2.2.7 Seller Use Case**

Before this use case can be initiated, the Seller has already connected to the **MASKM-Store**.

Use case: register a product for approval

**Diagram:**



### **Brief Description**

The Seller registers a new product.

### **Initial Step-By-Step Description**

Before this use case can be initiated, the Seller has already connected to the **MASKM-Store**.

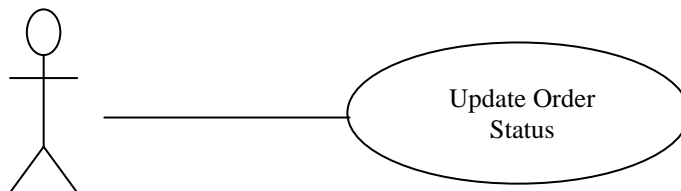
1. The Seller register a new product and it will be pending Admin approval

## 2.2.8 Seller Use Case

Before this use case can be initiated, the Seller has already connected to the **MASKM-Store**.

Use case: Update order status Shipped / Cancelled


**Diagram:**



Author

SELLER DASHBOARD











HOME / SELLER DASHBOARD



Fashion Store

- Dashboard
- Products
- Order
- Logout

Orders

Order Id	Creation Date	Price	Payment	Status	Action
102	2021-08-13	536	No	Cancelled	
107	2021-08-13	394	No	Pending	 
110	2021-08-13	536	No	Cancelled	
113	2021-08-13	86	No	Pending	 
122	2021-08-13	536	No	Pending	 
125	2021-08-13	536	No	Pending	 
128	2021-08-13	536	No	Cancelled	
141	2021-08-13	536	No	Pending	 
100	2021-08-13	500	No	Shipped	

### Brief Description

The Seller can update his the status of the orders belongs to him

### Initial Step-By-Step Description

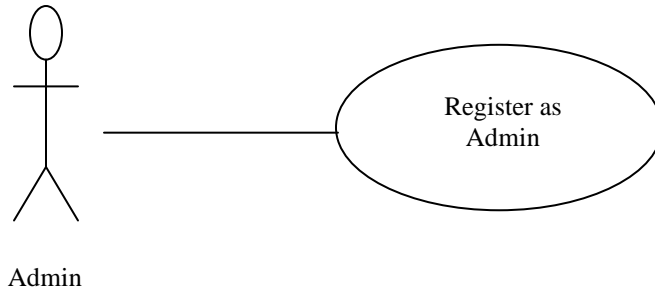
Before this use case can be initiated, the Seller has already connected to the **MASKM-Store**.

- The Seller register a new product and it will be pending Admin approval

## 2.2.9 Admin Use Case

Use case: Register as admin

**Diagram:**



### **Brief Description**

The Admin register in the system as admin to be able to access the admin functionalities.

### **Initial Step-By-Step Description**

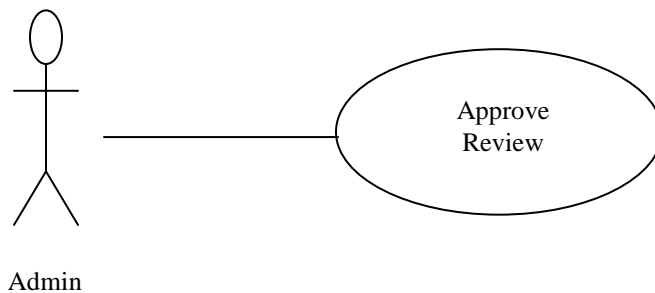
Before using the application as admin he need to register first and add admin role.

1. The Admin need to register as admin and get the admin roles.

#### **2.2.10 Admin Use Case**

Use case: Approve Review

### **Diagram:**



### **Brief Description**

The admin needs to login as admin and then he can approve a review of a product.

### **Initial Step-By-Step Description**

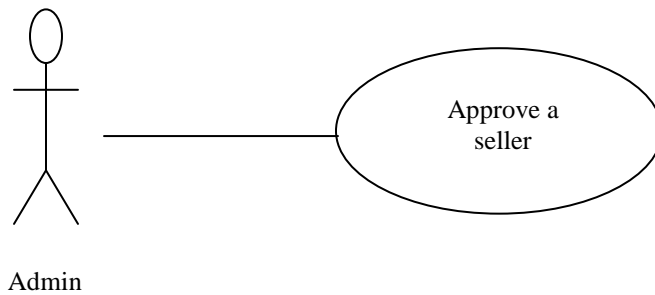
Before this use case can be initiated, the Admin has already connected to the Online Journal Website.

1. The Admin login to the system
2. The Admin will get all the un approved review
3. The Admin will approve/reject a product review

### 2.2.11 Admin Use Case

Use case: Approve Seller

**Diagram:**



**Brief Description**

The admin needs to login as admin and then he can approve a seller.

**Initial Step-By-Step Description**

Before this use case can be initiated, the Admin has already connected to the Online Journal Website.

1. The Admin login to the system
2. The Admin will get all the un approved seller
3. The Admin will approve/reject a seller

## 2.3 *User Characteristics*

The Buyer is expected to be able to browse all product's categories and drill down to see the products per category”

The Seller is expected to list his products add a new product and can list all the orders that is related to him and can cancel the order if it is not shipped or update the order status to be shipped

The Admin are expected to be able to approve reviews, approve sellers and add product category.

## 2.4 Non-Functional Requirements

- The system should use Spring boot as backend and react for the frontend and the GUI
- The backend code should be structured. It will consist of model(domain)layer , service (business) layer and Persistence (repository Layer)
- All backend API will be displayed in swagger
- The frontend is going to be developed by React

## 3.0. Requirements Specification

### 3.1 features

<b>Admin</b>
a. If the seller registers to the web site, he/she need to get approval from Admin in order to post products.
b. The admin should approve the reviews that are made from the Buyers.
<b>Seller</b>
a. Register as Seller
b. Product (CRUD). If a product has already been purchased, it cannot be deleted.
c. Seller cannot buy products from the website
d. Maintain orders <ul style="list-style-type: none"> <li>a) Cancel Order, the status of order on buyer's part should also changed</li> <li>b) Change Order status (Shipped-On the way-Delivered)</li> </ul>
<b>Buyer</b>
a. Register as Buyer
b. Follow and Unfollow Seller
c. Cannot sell items on this website
d. Can place an order <ul style="list-style-type: none"> <li>a) Maintain Shopping Cart (CRUD)</li> </ul>



<ul style="list-style-type: none"> <li>b) Maintain Shipping and Billing Address</li> <li>c) Maintain Payment</li> <li>d) Place order</li> <li>e) Every successful purchase (not returned), gain points from the website. You can use points to buy products (something like coupons)</li> </ul>
<ul style="list-style-type: none"> <li>e. Maintain Orders <ul style="list-style-type: none"> <li>a) Check Order History</li> <li>b) Can cancel order before shipping, after shipping cannot</li> <li>c) Download/Print receipt as PDF or Excel</li> <li>d) Write Product Review. Review must be approved by Admin before live.</li> </ul> </li> </ul>
<b>General</b>
Login/Logout
Security with JWT (Users should not be able to access other pages links)
Process verifications etc. (user get email of purchase, gets a message)
Validation is required for all form submission. Otherwise, you'll not get full credit on different feature part.
<b>Technical aspects</b>
Neat code and organization
Managed packages, folders, and files