

School of Applied Sciences  
Bachelor of Science in Computing

**COMP321-321 ISI Report**

Academic Year 2020/21   
2nd semester

|  |  |
| --- | --- |
| Online Shopping Mall Project | |
|  |  |
| Project number: | Group 5 |
| Team members: | Chi, Dend Yong Zhi (P-16-0498-1) |
|  | Ben, Leong Hang Pan (P-18-0187-5) |
|  | Tony, Tou Kin Hong (P-18-0191-5) |
| Supervisor: | Ms Calana Chan |
| Assessor: |  |
| Submission Date: | April 30, 2021 |

**Contents**

[1 Introduction 8](#_Toc70334360)

[1.1 Overview 8](#_Toc70334361)

[1.2 Objectives 9](#_Toc70334362)

[1.3 Risk Assessment 11](#_Toc70334363)

[1.3.1 Risk description, preventive measures and emergency response plan 12](#_Toc70334364)

[1.3.2 Reassess the risks after applying the emergency response plan 14](#_Toc70334365)

[2.Background and Related Work 16](#_Toc70334366)

[3.System Design 19](#_Toc70334367)

[3.1 Data Modelling 19](#_Toc70334368)

[3.2 Data Modelling 26](#_Toc70334369)

[*3.2.1* *Sequence Diagram* 26](#_Toc70334374)

[4. System Implementation 28](#_Toc70334375)

[4.1 Architecture 28](#_Toc70334376)

[4.1.1 Layered Architecture 28](#_Toc70334377)

[4.1.2 MVC Model 29](#_Toc70334378)

[4.2 Product Search 31](#_Toc70334379)

[4.3 Image Storage and Handling 33](#_Toc70334380)

[4.4 Password Security 35](#_Toc70334381)

[4.5 Purchase Order Processing 37](#_Toc70334382)

[4.6 Customers’ Ratings and Reviews 39](#_Toc70334383)

[4.7 Inventory checks 40](#_Toc70334384)

[4.8 Inventory checks 41](#_Toc70334385)

[5. Results and Discussion 43](#_Toc70334386)

[7.1 Project Outcome 43](#_Toc70334391)

[*7.1.1* *Product List* 43](#_Toc70334392)

[7.1.2 Product Detail Page 45](#_Toc70334393)

[7.1.3 Account Management 46](#_Toc70334394)

[7.1.4 Shopping Cart 49](#_Toc70334395)

[7.1.5 Purchase Tracking 51](#_Toc70334396)

[7.1.6 Product Catalog Maintenance 53](#_Toc70334397)

[7.1.7 Purchase Order Processing 58](#_Toc70334398)

[7.1.8 Customer Rating and Comment 59](#_Toc70334399)

[7.1.9 Notification 61](#_Toc70334400)

[7.1.10 Inventory Control 63](#_Toc70334401)

[7.2 Testing and System Evaluations 64](#_Toc70334402)

[6. Conclusion and Further Work 71](#_Toc70334403)

[Reference 72](#_Toc70334404)

[Appendix A. Project Management 73](#_Toc70334405)

[1 Product scope 73](#_Toc70334406)

[1.1 Product scope 73](#_Toc70334407)

[1.2 Process scope 75](#_Toc70334408)

[1.3 work-breakdown structure 82](#_Toc70334409)

[2 Schedule 83](#_Toc70334410)

[3 Job allocation 86](#_Toc70334411)

[Appendix B. Peer Assessment Form 89](#_Toc70334412)

[Appendix C. Program source code 90](#_Toc70334413)

**List of tables**

[Table 1: Table of Risk Ranking List 9](#_Toc70278934)

[Table 2: Table of risk ranking list after applying the emergency response plan 12](#_Toc70278935)

[Table 3: Testing of requirements A1, A2, A3, A4, A5, A6, A7, C1, C6, B4 61](#_Toc70278936)

[Table 4: Testing of requirements B1, B2, B3, B5 62](#_Toc70278937)

[Table 5: Testing of requirements C2, C3, C4, C5 62](#_Toc70278938)

[Table 6: Testing of requirements D1, D2, D3, D4, Z3, Z4 64](#_Toc70278939)

[Table 7: Testing of requirements E1, E2, E3, E4, E5, E6, Z5 65](#_Toc70278940)

[Table 8: Testing of requirements F1, F2, F3, F4, F5, F6, F7, Z6. 66](#_Toc70278941)

**List of figures**

[Figure 1: The impact and possibility of risk 11](#_Toc70278942)

[Figure 2: Impact and risk probability after reassessment 14](#_Toc70278943)

[Figure 3: Entity Relationship Diagram (After Normalization) 18](file:///C:\Users\BenLeong\Desktop\ISI\report\ISI%20report_A5.docx#_Toc70278944)

[Figure 4 Website architecture 27](file:///C:\Users\BenLeong\Desktop\ISI\report\ISI%20report_A5.docx#_Toc70278945)

[Figure 5 MVC model 28](#_Toc70278946)

[Figure 6: JavaScript Code Of Search Function 29](#_Toc70278947)

[Figure 7: PHP Code of Search Function at Customer Side 30](#_Toc70278948)

[Figure 8:PHP Code of Search Function at Vendor Side 30](#_Toc70278949)

[Figure 9: Code of Image Update 31](#_Toc70278950)

[Figure 10: Code of Image Storage 32](#_Toc70278951)

[Figure 11: Code of Image Removal 32](#_Toc70278952)

[Figure 12: Code of JavaScript Check Password Regular Expression 33](#_Toc70278953)

[Figure 13: Code of Hash Password 34](#_Toc70278954)

[Figure 14: Code of Change Password 34](#_Toc70278955)

[Figure 15: Purchase order status process 35](file:///C:\Users\BenLeong\Desktop\ISI\report\ISI%20report_A5.docx#_Toc70278956)

[Figure 16:Code of Order Hold 36](#_Toc70278957)

[Figure 17: Code of Order Shipped 36](#_Toc70278958)

[Figure 18: Code of Order Cancel 36](#_Toc70278959)

[Figure 19:PHP code of Rating and Reviews 37](#_Toc70278960)

[Figure 20: HTML code of Rating and Reviews 38](#_Toc70278961)

[Figure 21:Code of check out of stock 38](#_Toc70278962)

[Figure 22: Graphic User Interface of Notification 39](#_Toc70278963)

[Figure 23: Graphic User Interface of Notification 40](#_Toc70278964)

[Figure 24:Figure of Product list 41](#_Toc70278965)

[Figure 25:Figure of the product list page 42](#_Toc70278966)

[Figure 26: Figure of changing the product list paging 42](#_Toc70278967)

[Figure 27: Figure of comparison of product list filtering 42](file:///C:\Users\BenLeong\Desktop\ISI\report\ISI%20report_A5.docx#_Toc70278968)

[Figure 28: Figure of Product Detail Page 43](#_Toc70278969)

[Figure 29: Figure of Registration Page 44](#_Toc70278970)

[Figure 30: Figure of Error Message of Registration 44](#_Toc70278971)

[Figure 31:Figure of Login interface 45](#_Toc70278972)

[Figure 32:Figure of interface after login 45](#_Toc70278973)

[Figure 33:Figure of The Validation of Login 45](#_Toc70278974)

[Figure 34:Figure of changing password 46](#_Toc70278975)

[Figure 35: Figure of the database view 46](#_Toc70278976)

[Figure 36: Figure of Successful Message after adding product to shopping cart 47](#_Toc70278977)

[Figure 37: Figure that show the number in shopping cart item 48](#_Toc70278978)

[Figure 38: Figure of purchase tracking page interface 49](#_Toc70278979)

[Figure 39:Figure of purchase order detail page 50](#_Toc70278980)

[Figure 40:Figure of Product Catalog Maintenance 51](#_Toc70278981)

[Figure 41:Figure of Vendor Searching by Product Number Function 52](#_Toc70278982)

[Figure 42:Figure of Vendor Searching by Product Keyword Function 53](#_Toc70278983)

[Figure 43:Figure of Vendor Searching by Product Keyword and Brand fitter Function 53](#_Toc70278984)

[Figure 44:Figure of Add New Product button 54](#_Toc70278985)

[Figure 45:Figure of Add New Product page 54](#_Toc70278986)

[Figure 46:Figure of Edit Information button 55](#_Toc70278987)

[Figure 47:Figure of Edit information interface 55](#_Toc70278988)

[Figure 48:Figure of order processing 56](#_Toc70278989)

[Figure 49:Figure of Product Detailed Order Page 57](#_Toc70278990)

[Figure 50: Figure of Product Detailed Order Page of User 58](#_Toc70278991)

[Figure 51: Figure of comment page interface 58](#_Toc70278992)

[Figure 52:Figure of The Customer Rating and Review 59](#_Toc70278993)

[Figure 53:Figure of Notification Interface 59](#_Toc70278994)

[Figure 54:Figure of Notification Interface 60](#_Toc70278995)

# Introduction

This section covers an overview report of the current Macao market, and we will describe the objectives and plans of the project. Finally, the risk assessment and solution are proposed.

## Overview

With the continuous progress of science and technology, mobile phones have become an indispensable technology product in human life. According to The Internet Usage Trends in Macau conducted in 2020 by The Macao Association for Internet Research, the average time spent on The Internet in Macao is 4 hours per day, and the penetration rate of mobile phones is 94%, while the figure was only 46% in 2011 [1], which proves that mobile phones are becoming more and more popular. People tend to change their mobile phones more frequently. To follow technology development, users need a new phone to get the most convenient and latest functions.

Besides, shopping on the Internet has also become common because the Internet touches almost every corner around us. Customers can buy products anytime, anywhere through a browser on a PC or smartphone. The vendor can also manage products or process orders online. The perfect market environment combined with advanced technology provides excellent retail industry opportunities. Especially in Macao, online sales can significantly reduce the cost of the rent. Therefore, our project’s goal is to establish a shopping platform for selling mobile phones. We provide various functions that consumers need. For example, a consumer can add a product to a shopping cart before making a purchase and track the status of the order. Vendors can also manage orders and check fund transfer.

## Objectives

The main goal is to build an online shopping centre system, enabling customers to browse, select and purchase products in the mall, and vendors to manage, maintain and process the entire online shopping system.

The section below illustrates the 40 requirements which will be implemented from 8 aspects.

1. **[T1] Display product list**

Similar to most online shopping system, we design a list of products for customers to browse through. A thumbnail image of the product is displayed in the list. Customers can search a product by keywords, sort the product list by price and filter products by brand.

1. **[T2] User account Management function**

According to the requirements, users need to log in to an account when purchasing goods. Users without an account need to fill in the basic information for registration (user name, password, email, and address) and then log in to the system to purchase products. For security reasons, the customer's password is hashed and stored in the database.

1. **[T3] Product Page**

By clicking on a specific product in the product list, customers can be redirected to the product details page. Users can view more information about the product on the product details page. The website will display detailed pictures on this page to provide users with more details of the product. Customers can add products to their shopping cart by clicking a button.

1. **[T4] Shopping cart**

The shopping cart function is implemented in the system. Users can add multiple products to the shopping cart before making the purchase. In the shopping cart, the customer can modify each product's quantity, remove specific products, change or add a shipping address, and check out to make a purchase order.

1. **[T5] Purchase Tracking**

In this project, customers can view their past and current purchase orders. Customers can view these orders in the order list. Clicking on an order takes the customer to the order details page to see more information. The customer can cancel the order before shipping on the order details page.

1. **[T6] Management product features**

The system also provides some functions for vendors to manage their products. Vendors can create new products, browse each product's information, modify the information and pictures of existing products, and delete certain products.

1. **[T7] Order Processing**

The vendor can also process orders through the system. The order list provides all the order records of the customers. Vendors can filter these orders based on their status. By clicking on an order in the order list, the vendor can go to the details page for a particular order. The vendor can send the unshipped order, hold the order, and cancel the order before it is shipped on the details page.

1. **[T8] Advanced Function**

In order to improve the user experience, the project has some advanced features. For example, customers can rate and comment on products. They can also quickly know which products are hot-selling and out-of-stock products for vendors. The notification function allows customers to see the status of the order soon.

## Risk Assessment

A stable online mall system requires a comprehensive risk assessment and an effective emergency plan. The major threats to the project and their probabilities are evaluated by priority in Table 1. Figure 2 shows the impact and probability of risk.

Table 1: Table of Risk Ranking List

|  |  |
| --- | --- |
| Priority | Risk Identifier and Description |
| Risk 1 | Server-side data inconsistence |
| Risk 2 | Server overloads |
| Risk 3 | The server stops working (Hardware Failure) |
| Risk 4 | Packet injection |
| Risk 5 | SQL injection |
| Priority 1 is the highest priority | |

|  |  |  |  |
| --- | --- | --- | --- |
| Impact  Probability | Low | Medium | High |
| Low |  | * Server overloads | * Server hardware failures. * SQL injection |
| Medium |  |  | * Packet injection |
| High | * Server-side data inconsistency |  |  |

Figure 1: The impact and possibility of risk

### 1.3.1 Risk description, preventive measures and emergency response plan

**Risk 1: Server-side data inconsistence**

Data inconsistency happen when multiple tables within a database deal with the same data but may receive it from different inputs.

**Precaution:** Strictly set the database format, using dual data validation, that is, the user input with a front-end data validation, when sent back to the server, data validation is done again, to ensure the accuracy of the data.

**Solution:** Clarity is the best way to resolve data inconsistency, Besides, it is also crucial to avoid duplication of data as much as possible. For example, the database should use the foreign key to obtain data to avoid repeated data input.

**Risk 2: Server overloads**

Server overload means that the request exceeds the load range of the server. There are two kinds of situations leading to server overload. The first one is too many normal users, leading to server overload; the second one is malicious attacks such as DDoS.

**Precaution:** This web hosting centre has a traffic monitoring service. If the traffic exceeds a specific range, it will issue an alert.

**Solution:** If there is a need, we can always upgrade the service, requiring more space and network speed. The website can also intercept DDoS and other consumption of network resources of the attack.

**Risk 3: Server stops working**

Server hardware may go wrong after millions of calculations and high temperatures, and a small error can cause a machine to stop running.

**Solution:** It is described in detail in the web hosting contract that there is a guarantee that our servers work 99.9% of the time. All the servers are monitored 24/7 by an expert team of Admins and Engineers.

**Risk 4: Packet injection**

Packet injection in computer networking is the process of interfering with an established network connection using constructed packets to appear as part of the standard communication stream.

**Solution:** Our website has applied for SSL credentials, ensuring the packet's encryption and preventing the package from being intercepted and modified.

**Risk 5: SQL injection**

SQL injection is a security vulnerability in the application program and database layer. SQL instructions are included in the string input by the user. If the character check is ignored in the poorly designed program, the system will execute the user's SQL code input.

**Precaution:** To prevent SQL injection, we should check the user input data at the front end, and if the user enters SQL code, we forbid it from being sent to the server.

**Solution:** We can solve SQL injection by verifying input data and managing database permissions. For example, some pages only need the function of select, so we only allow select permissions to avoid using full privileges.

### 1.3.2 Reassess the risks after applying the emergency response plan

When an emergency response plan is applied, two risks reduce the probability of occurrence. Accordingly, the new risk prioritization table is shown below (see Table 2), and the impact and probability of risk are shown in Figure 2.

Table 2: Table of risk ranking list after applying the emergency response plan

|  |  |
| --- | --- |
| Priority | Risk Identifier and Description |
| Risk 1 | Server-side data inconsistence |
| Risk 2 | Server overloads |
| Risk 3 | The server stops working (Hardware Failure) |
| Risk 4 | Packet injection |
| Risk 5 | SQL injection |
| Priority 1 is the highest priority | |

|  |  |  |  |
| --- | --- | --- | --- |
| Impact  Probability | Low | Medium | High |
| Low | * Server-side data inconsistency | * Server overloads | * Server hardware failures. * SQL injection * Packet injection |
| Medium |  |  |  |
| High |  |  |  |

Figure 2: Impact and risk probability after reassessment

The report is about our online shopping mall. Chapter 1 gives a brief introduction to the whole project. In Chapter 2 of this report, we will cover the background of our project and some of the rival research. Chapter 3, system design, namely data modelling and dynamic modelling. In chapter 4, we will mention the implementation of the system. Chapter 5, the end of the report will include the results and discussion by presenting some project results and testing the system. Chapter 6, we will have a conclusion and further work on our project.

# 2.Background and Related Work

Electronic commerce (e-commerce) is the activity of electronically buying or selling products on online services or over the Internet. The common features of e-commerce websites and the background of our project are introduced in the background section. Besides, the comparison between Taobao, Amazon and our works are discussed in the related work section.

**2.1 Background**

With the development of science and technology, e-commerce has become a part of daily life, many people like to use e-commerce websites to do shopping online now. People just need to place an order on some e-commerce platforms or websites, and then they can wait for the goods to be delivered. This is a very convenient way of shopping, especially at special times. For example, during the period of COVID-19, people are going out less often, shopping outside becomes difficult and dangerous. Using e-commerce platforms allows people to shop at home, which reduces the risk of COVID-19 infection, and to shop normally.

There are some common features of e-commerce web sites:

1. Customer login system: The user should create an account and login into the system.
2. Shopping cart: User can add their favourite items to the shopping cart.
3. Search bar: The user can use the search bar to enter the keyword to search for the items they want.
4. Online payment system: The user can complete transactions through an online payment.
5. Classification of goods: On the homepage of the website, there are some product categories for users to choose from.
6. Consumer reviews on products: This part let the customers read, rate and give reviews to products.

Every year new smartphones are being launched and smartphones have become important things in people's lives. Some people may want to change smartphones frequently with the latest version of the phone, and some people may change smartphones after a long period, so we want to build an e-commerce platform for smartphone transactions.

**2.2 Related work**

Taobao [2] is a Chinese online shopping website and is mainly used by users in Asia. It is one of the world’s biggest e-commerce websites. Amazon [3] is an American multinational online shopping website that is used by users all over the world. They are both world-renowned e-commerce platforms, and they have all the common features of an e-commerce website, so we will compare them in several different ways.

Customer service:

On Taobao, buyers can contact the store through Aliwangwang, and they will get feedback very quickly, which is very convenient for buyers. On the contrary, in Amazon, the buyer can only ask the seller by leaving a message at the bottom of the product, but the seller may not reply in time, which is inconvenient for the buyer.

Goods recommendation:

On Amazon, when we look at a good, the bottom of the page will have a bar to tell us what related items other customers also browsed/bought. However, in Taobao, the system will only show system recommendations at the bottom of the page.

Language options:

Since Amazon's customer base is people from all over the world, it has a language switching function, making it easy for people from different regions to browse. Taobao's main customer base is Chinese, so it only has a Chinese-language interface but no other language interface, which is very inconvenient for foreigners living in Asia.

Category filter:

At Amazon, when a user has finished searching for a product, the filter is on the left-hand side of the screen. However, in Taobao, after the user has searched the product, the filter will be at the top of the interface.

In conclusion, although Taobao and Amazon are popular e-commerce sites, they also have several differences in some functions. The reason for these differences may be that the two e-commerce websites have targeted different customers. To make our work better, we think they have their advantages and we can learn something from their features.

In our system, we will put the filter on the left-hand side, we think the filter is placed on the left-hand side can make the interface look cleaner and better organized.

# 3.System Design

To implement an e-business system, a good database structure and some special considerations must be implemented. The two phases of the system design, an overview of the overall database structure, details and special considerations will be described below, and a detailed description of the system's dynamic data exchange methods between the different parts of the system will be covered.

## 3.1 Data Modelling

The following figure is a standardized ER diagram of the system. There are nine entities in the database modelling.

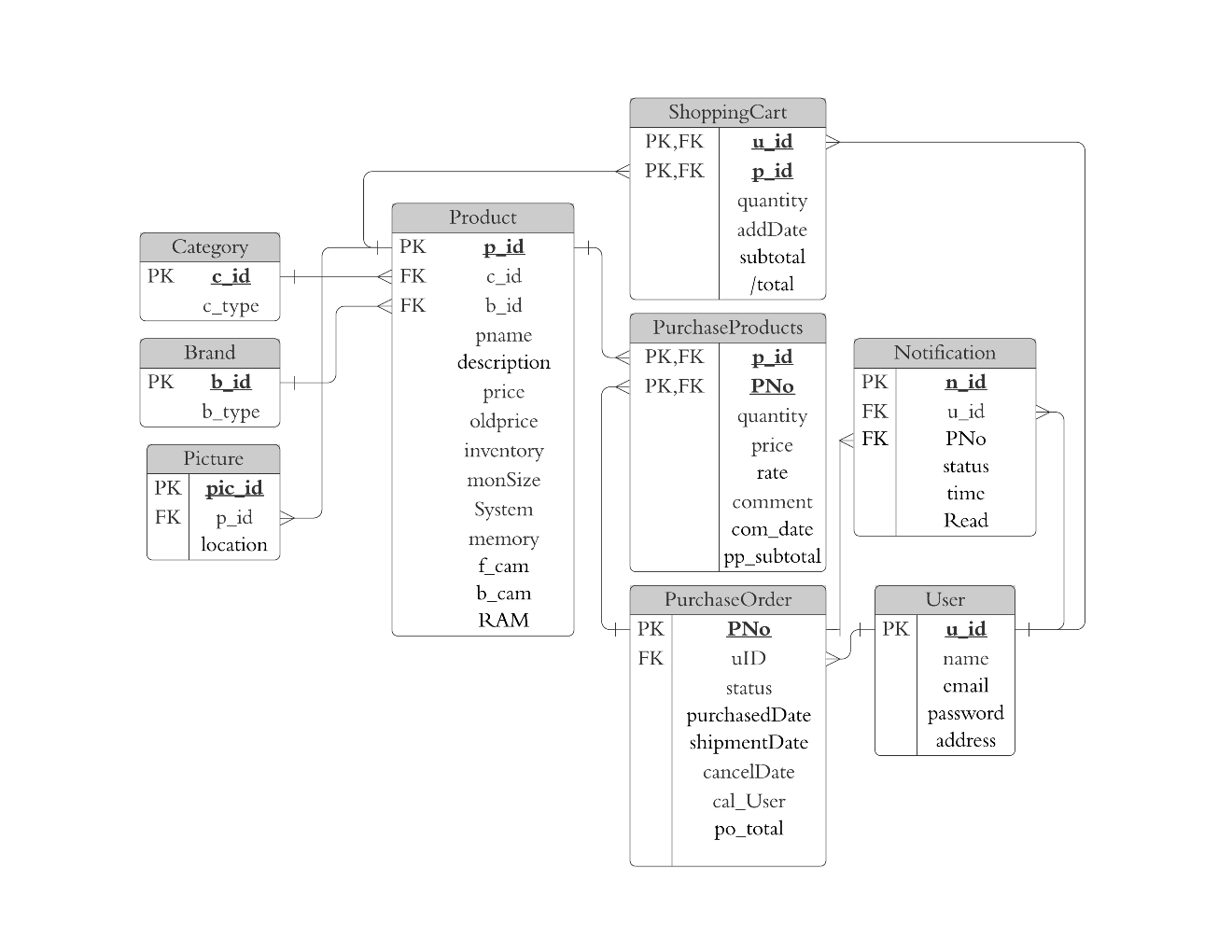


Figure 3: Entity Relationship Diagram (After Normalization)

Entity Product represents the products (smart phones) with a unique product id and some detailed attributes( smartphone name, description, price, inventory, monitor size, system and images) which are required in the purchase process. Since a product may have many detailed pictures, there is a named Picture entity to store information about these detailed pictures. It contains foreign keys, indicating which product it belongs to, and indexes, which indicate the order of the images. The database we used in this system is MYSQL, it cannot store any format of image files, so the path of the image is stored in the database rather than the image file. Entity Category is created to save different smartphones categories. Entity Brand is created to save different smartphones brands.

To implement the purchase order tracking functions, the PurchaseOrder entity is designed to store the data of every purchase order. It includes the user's id, status, date of purchase, cancel date and user (if the order is cancelled), purchase amount total. Since an order may have multiple products, so a PurchaseProducts entity is created to store information about each product in a particular purchase order. The purchased products entity contains a PNo as a foreign key for a particular order and a pid as a foreign key for a particular product. It also contains the price, review, comment date, price, and subtotal amount of the product. Since the price of the product might be changed later, the price is also an attribute in the table, which is used to record the fees paid when the customer checks out.

The online mall system has a shopping cart function, therefore the entity ShoppingCart is created to store product information in a specific user's shopping cart. The entity stores a uid and pid to indicate the customer to which it belongs, the number of products, the date the product was added to the cart, the price of the product, and the foreign key for that particular product. The reason for storing only the product ID is that the price of each product may change before the user checks out, so only one reference is stored in the ShoppingCart entity to maintain overall system consistency.

Customers are represented by the Users entity. It contains basic information about customers, for example, user name, email, password and address.

To notify a customer about the status of the item purchased, entity Notification is used to create and save notification with nid. This entity contains uid, PNo, status, time of the notification was created, and whether the notification is read or unread.

Presented below is our database data dictionary which will give further details on the domain of the attribute:

“Product” is an entity that contains all product’s information. p\_id of “Product” is referenced by two entities called “ShoppingCart”, “PurchaseProducts”.

|  |  |  |  |
| --- | --- | --- | --- |
| **Product** | | | |
| Attributes | Description | Data Type(Remarles) | Remind |
| p\_id  c\_id  b\_id  pname  description  price  oldprice  inventory  monSize  System  memory  f\_cam  b\_cam  RAM | Uniquely identifies the product  To identify the category it belongs  To identify the brand it belongs  Name of product  Short description of product  Current product price  Original price of product  Current product quantity of product  Inches of the product screen  The system of product  The memory size of the product(Gigabyte)  Product front camera pixel (Million)  Product back camera pixel (Million)  The capacity of product random access memory(Gigabyte) | Integer(8)  Integer(8)  Integer(8)  Text(255)  Text(255)  Double  Double  Integer(5)  Double(8,2)  Text  Integer(3)  Integer(5)  Integer(5)  Integer(3) | Primary Key  Foreign Key  Foreign Key |

“User” is an entity that contains all user data. u\_id of “User” is referenced by three entities, namely “ShoppingCart”, “Notification” and “PurchaseOrder”.

|  |  |  |  |
| --- | --- | --- | --- |
| **User** | | | |
| Attributes | Description | Data Type(Length, Decimal Point) | Remind |
| u\_id  name  email  password  address | Uniquely identifies the customer  The name of the user  The email address of the user  The password of the user’s account  The shipping address of the user | Integer(8)  Text(255)  Text(255)  Text(255)  Text | Primary Key |

“Category” is an entity that contains all the categories of products. c\_id of “Category” is referenced by the entity called “Product”.

|  |  |  |  |
| --- | --- | --- | --- |
| **Category** | | | |
| Attributes | Description | Data Type(Length, Decimal Point) | Remind |
| c\_id  c\_type | Uniquely identifies the category  The name of a type | Integer(8)  Text | Primary Key |

“Brand” is an entity that contains all the brands of products. b\_id of “Brand” is referenced by the entity called “Product”.

|  |  |  |  |
| --- | --- | --- | --- |
| **Brand** | | | |
| Attributes | Description | Data Type(Length, Decimal Point) | Remind |
| b\_id  b\_type | Uniquely identifies the brand  The name of brand | Integer(8)  Text | Primary Key |

“Picture” is an entity that contain the path of all product pictures.

|  |  |  |  |
| --- | --- | --- | --- |
| **Picture** | | | |
| Attributes | Description | Data Type(Length, Decimal Point) | Remind |
| pic\_id  p\_id  location | Uniquely identifies the picture  To identify the product  The location of picture | Integer(8)  Integer(8)  Text | Primary Key  Foreign Key |

“PurchaseOrder” is an entity that records all the orders of the system. PNo of “PurchaseOrder” is referenced by two entities called “PurchaseProducts” and “Notification”.

|  |  |  |  |
| --- | --- | --- | --- |
| **PurchaseOrder** | | | |
| Attributes | Description | Data Type(Length, Decimal Point) | Remind |
| PNo  uID  status  purchasedDate  shipmentDate  cancelDate  cal\_User  po\_total | Uniquely identifies the purchase order  To identify the user it belongs  The product’s delivery status  The date of purchase  The date of shipped.  The person who deleted the Order.  The total price of purchase order | Integer(8)  Integer(8)  Tinyint(1)  Date  Date  Date  Text  Double(8,2) | Primary Key  Foreign Key  [1:Pending  2:Shipped  3:Canceled  4:Hold]  [Can be null]  [Can be null]  [Can be null]  [Can be null] |

“p\_id” and “PNO” are the composite primary key in “PurchaseProducts”, that records the corresponding quantity of purchased products and other details.

|  |  |  |  |
| --- | --- | --- | --- |
| **PurchaseProducts** | | | |
| Attributes | Description | Data Type(Length, Decimal Point) | Remind |
| p\_id  PNo  quantity  price  rate  comment  com\_date  pp\_subtotal | To identify the product it belong  To identify the purchase order it belong  The number of products purchased  The purchase price of the product  The rating of the product  The comment of Customer reviews of the product  The date of comment  The subtotal price of purchase products | Integer(8)  Integer(8)  Integer(6)  Double  Double  Text  Date  Double(8,2) | Primary Key  Foreign Key  Primary Key  Foreign Key  [Can be null] |

"ShoppingCart" is an entity that records the products each user wishes to purchase in the ShoppingCart.

|  |  |  |  |
| --- | --- | --- | --- |
| **ShoppingCart** | | | |
| Attributes | Description | Data Type(Length, Decimal Point) | Remind |
| u\_id  p\_id  quantity  addDate  subtotal  total | To identifies the user it belong  Uniquely identifies the product  The number of the purchased product in Shopping Cart  The date the user added the product to the shopping cart  The subtotal of shopping cart  The total of shopping cart | Integer(8)  Integer(8)  Integer(6)  Date  Double  Double | Primary Key Foreign Key  Primary Key Foreign Key  C:\Users\BenLeong\Downloads\DBMS ER diagram (UML notation) (7).png  Derived Attribute  Derived Attribute |

"Notification" is an entity that records order status and alerts the user.

|  |  |  |  |
| --- | --- | --- | --- |
| **Notification** | | | |
| Attributes | Description | Data Type(Length, Decimal Point) | Remind |
| n\_id  u\_id  PNo  status  time  Read | Uniquely identifies the notification  To identify the user it belongs  To identify the purchase order  The record of corresponding order status  (Pending or Shipped or Cancelled by the vendor or cancelled by the user or Hold)  The time of order status changed  Determine the notification is read by the user or not | Integer(8)  Integer(8)  Integer(8)  Integer(1)  Date  Boolean | [Primary Key]  [Foreign Key]  [Foreign Key]  [1:Pending  2:Shipped  3:Canceled  4:Hold]  [0 or 1] |

## 3.2 Data Modelling

In this section, the dynamic modelling is specified with the use of a sequence diagram.



### *Sequence Diagram*

The following diagram depicts how a customer completes the entire purchase process in our system.

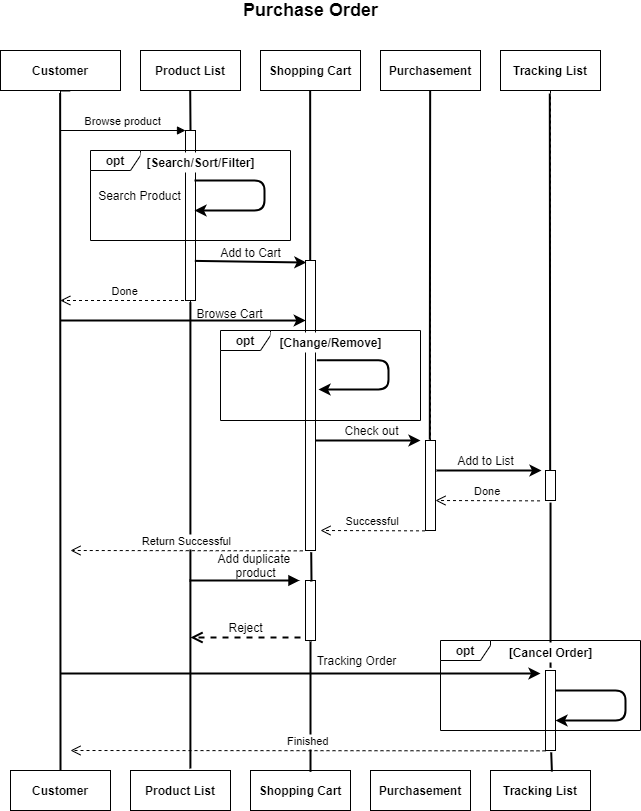


Figure 4: Sequence Diagram for User Purchasing

First, customers can browse the various products on the product list page. Alternatively, they can filter by product brand and price and search for product names using one or more keywords. Before the customer chooses the products they want, he/she can add them to the shopping cart. However, the customer cannot add duplicate products to the shopping cart. The customer can then check out, change the product quantity or delete the product. When the customer checks out, our system saves all the products in the shopping cart. The purchase order information is displayed to the customer. At the same time, the server will generate a tracking order list of orders to the customer. Therefore, customers can track their orders by checking their status. The customer can wait for the order to ship or cancel the order.

# 4. System Implementation

In this section, we introduce the architecture and functions of the online mall.

## Architecture

The online mall consists of two architectures: layered architecture and the MVC model. The architecture is divided into the following two sections.

To achieve efficient, robust, and developer-friendly development, we chose some popular technologies as development tools. The database used in this system is MYSQL. The entire system is written in PHP with no framework. The whole system adopts MVC (Model - View - Controller) design mode.

## 4.1.1 Layered Architecture

Our project uses the layered architectural pattern to divide code with similar responsibilities into layers. JavaScript handles the presentation of content and interaction with users. It gets the data from Apache, use PHP and Bootstrap to receive data from the upper layer and convert it. At the back-end is the MySQL server which is used to retrieve data from the data source.

Figure 5 Website architecture

Web Browser  
HTML and JavaScript

Web Server  
Apache

Application Server  
PHP

Database  
MySQL Server

## MVC Model

Our project also uses the MVC model to separate the model, view, and controller. Speeds up our development process and provides an easy way to divide work. The architecture is shown below.

**Controller**

Handling HTTP Requests

Processing PHP Requests

Data Validations

**View**

Generating Page

Using Templates

**Model**

Database Operation

Verification

Business Processes

HTTP Request

Return Display

Send Request

Return Response

Get Data

Send

Figure 6 MVC model

## Product Search

We implement three types of search capabilities for our customers. Each user can search for products by the Product keyword. If the user uses a brand or category filter, the search function will filter the product by brand or category.

The search function consists of JavaScript and PHP. JavaScript is responsible for checking whether the user has entered a keyword and clicked the search button. If it detects that the user has entered any character and clicked the search button, it will make the keyword a GET parameter. PHP is responsible for receiving Get Parameters (such as search keywords, brands, category). PHP converts these conditions into SQL queries. After filtering, users can get the products they want.

As shown in the following code, the customer and vendor search functions are very similar. The only difference is that vendors can search for products by product ID.

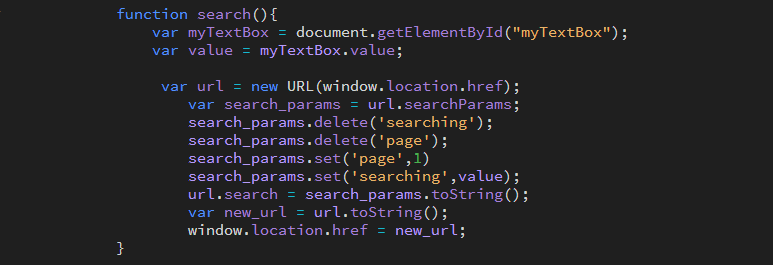


Figure 7: JavaScript Code Of Search Function

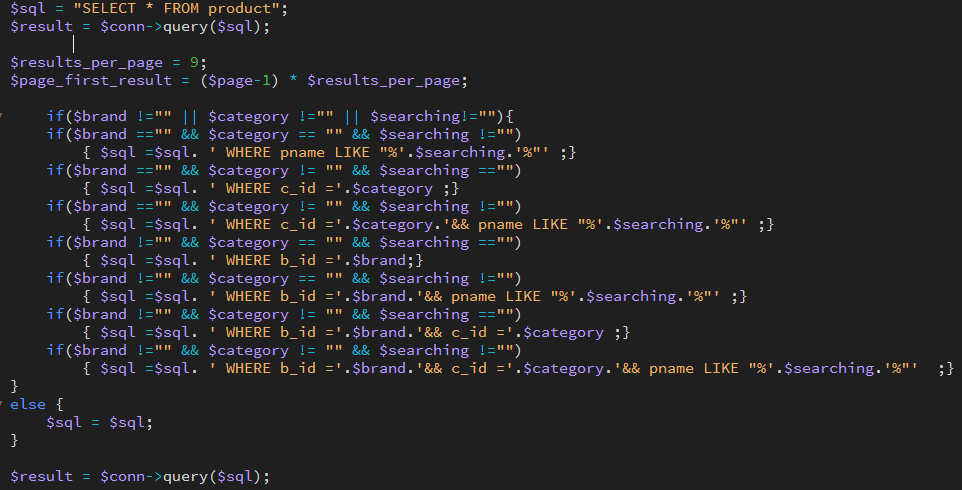


Figure 8: PHP Code of Search Function at Customer Side

The customer's search function can search for products by product keywords. Besides, it also supports brands and category of filters.



Figure 9:PHP Code of Search Function at Vendor Side

The customer's search function can search for products by product keywords and product ID. Besides, it also supports brands and category of filters.

## Image Storage and Handling

When the vendor uploads the image to the server, it assigns the image a unique id before storing it. After that, it will be stored in the root \img folder specified in photos\_management.php (Figure 10). Then, the website will insert the file path into the database.

For images displayed on the client, the server transfers all image paths by retrieving the image in the database. The server provides two types of users, the vendor and the customer, to view the product image and allows the vendor to edit it. Both of them can view product thumbnails on the product list page, and all of the detailed images on the product details page. Besides, vendors can delete and re-upload images on the photos management page.

This part is composed of PHP. PHP is responsible for uploading the picture back to the root directory and updating the path of the picture in the database.

We divide the upload photos into three states, state one replaces the photos, the vendor can replace the old photos with the new photos. State two, adds photos, there are no photos in the field, and the Vendor is allowed to upload photos. State three, delete the photos, if Vendor feels the photos are not satisfied, you can delete the photos.

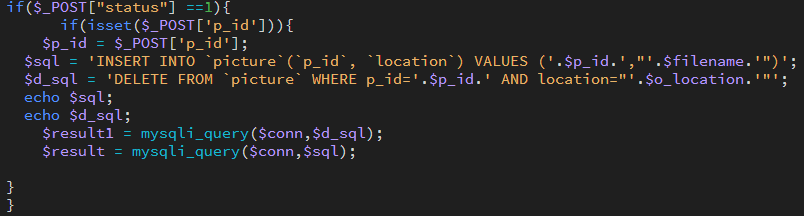


Figure 10: Code of Image Update

Since this system only allows 1-5 pictures to be uploaded. If the vendor replaces a new picture, then the system needs to delete the previous picture.

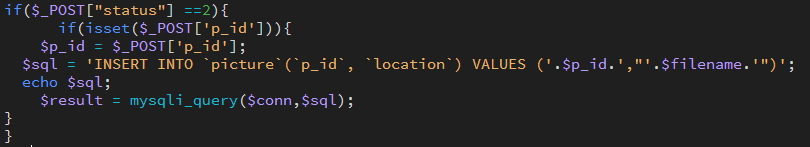


Figure 11: Code of Image Storage

Figure 11 shows how to store the image in the server. The condition indicates whether the image is empty (you can upload it if it is empty, otherwise you can only replace the image). If the conditions pass, the image will be uploaded to the server and the path will be inserted into the database. The image has an auto-incrementing number and the value of the reference product ID. The image name depends on the image name and path.



Figure 12: Code of Image Removal

The Vendor can delete the picture by pressing the delete button. When the Vendor deletes the detail image, the server supports the automatic redistribution of the order of the detail image. For example, if the Vendor deletes the product detailed information image 1 and it has 4 detailed information images, the detailed information images 2 and 3 will be automatically changed to the order of 1 and 2.

## Password Security

This part is composed of JavaScript and PHP. JavaScript is responsible for checking whether the password entered by the user meets the conditions. If it meets the conditions, it will use the post method to send it to PHP. PHP is responsible for encrypting and uploading the user's password to the database.



Figure 13: Code of JavaScript Check Password Regular Expression

The figure (Figure 13) shows that the password must enter at least one uppercase letter and the entire letter to satisfy the regular expression and the new password and the confirmed password must be the same.

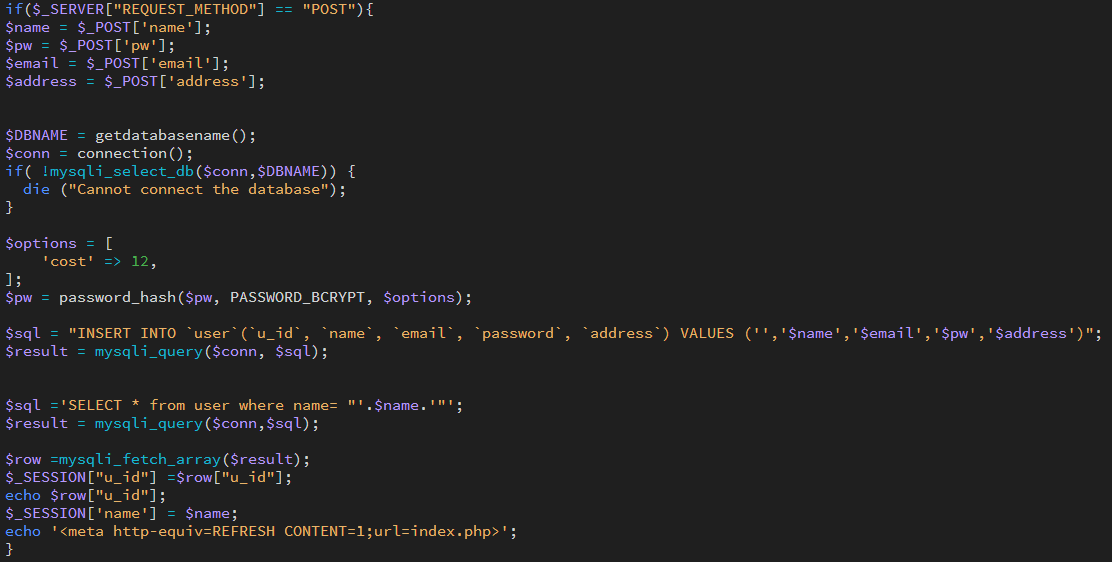


Figure 14: Code of Hash Password

because the database administrator or hacker can access the database. Therefore, the server must hash the password before storing it in the database. PHP provides password encryption crypt() for hashing passwords, so we implement the hash function by using it.

PHP provides a hash function to verify user passwords. This check method will verify that a given plain text string corresponds to a given hash. If the check passes, the user will successfully log in or change the password.

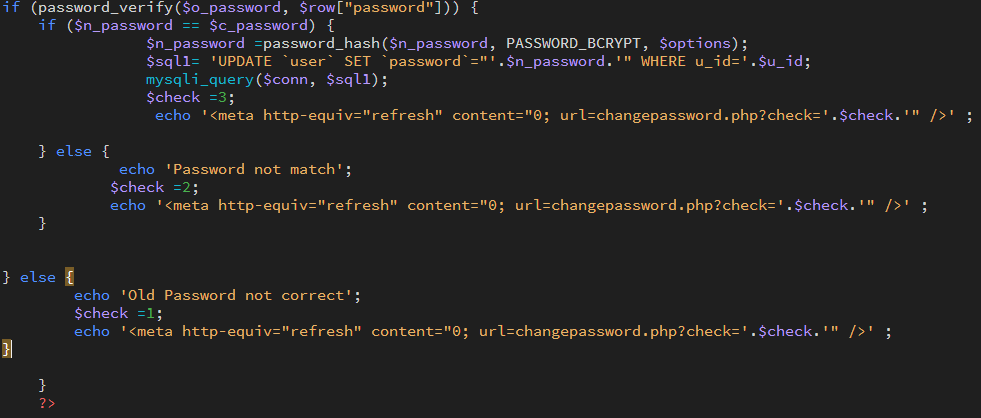


Figure 15: Code of Change Password

In the change password function, the server will execute a validator to verify all substituted data. After that, pass the data to the hash module and check whether the old password matches. Besides, you can check whether the new password and the repeated new password match. If all the above conditions are passed, the password will be changed.

## Purchase Order Processing

The order in the shopping mall system has 4 states: pending, shipped, cancelled and Hold. To facilitate management and save database storage, 4 numbers are used to represent these 4 states. 1 to 4 respectively represent pending, Shipped, cancelled and Hold. The following sections will use numbers in brackets after the status term to indicate status. (For example, pending review (1))

According to the requirements of the project specification, the order must start with the status "pending" (1). Then, the order can be cancelled(3). by the customer or vendor before shipment, otherwise, it can be shipped(4). by the vendor. If the vendor cancels the order, the order status should be changed to cancelled(3). Figure 13 below shows the status change in the system. If the product is out of stock, the vendor can choose to hold(4) and keep the order, but not ship it temporarily.

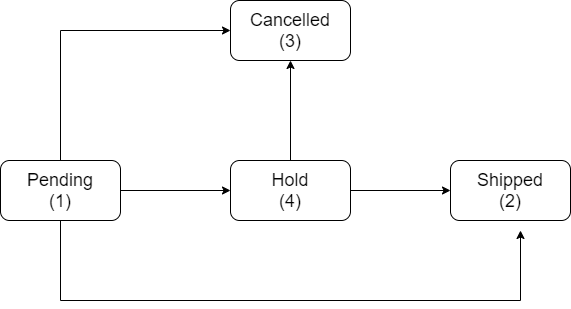


Figure 16: Purchase order status process

Vendor have 3 implemented functions to process order status: Cancel(), Hold() and Shipping(). There is also a function called Cancel() for customers to cancel orders. In this system, only 3 functions are the way to modify the order status. Before any state of the order. The system should confirm the product status, these three functions only need to check whether the current status of the order is one of the possible previous statuses of the next status.

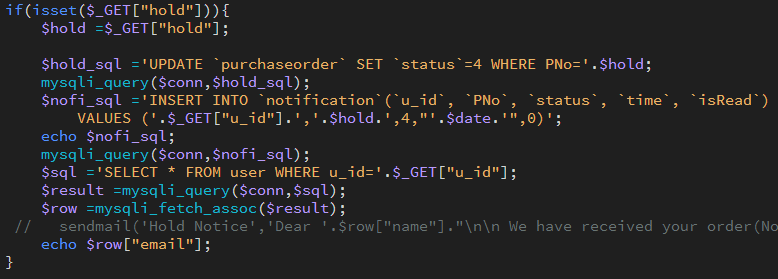


Figure 17:Code of Order Hold

The hold function transfers the state from pending (1) to hold (4). Turn to the check to check whether the order status in the code is pending (1).

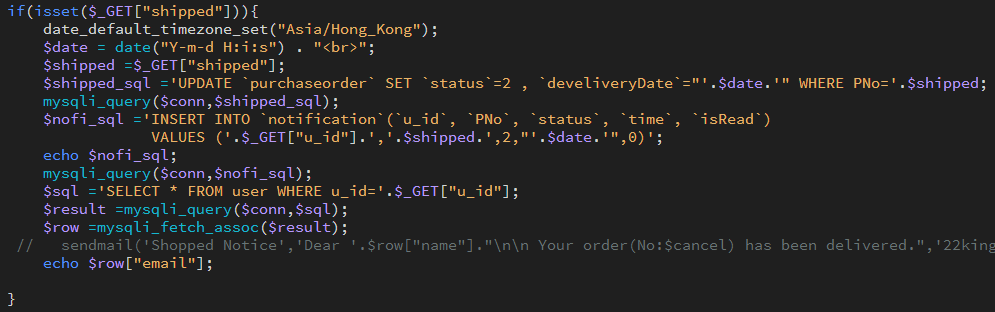


Figure 18: Code of Order Shipped

The Shipped function transfers the order from hold (4) to shipped (2). Therefore, the code checks whether the status of the order is hold (4) before shipping.

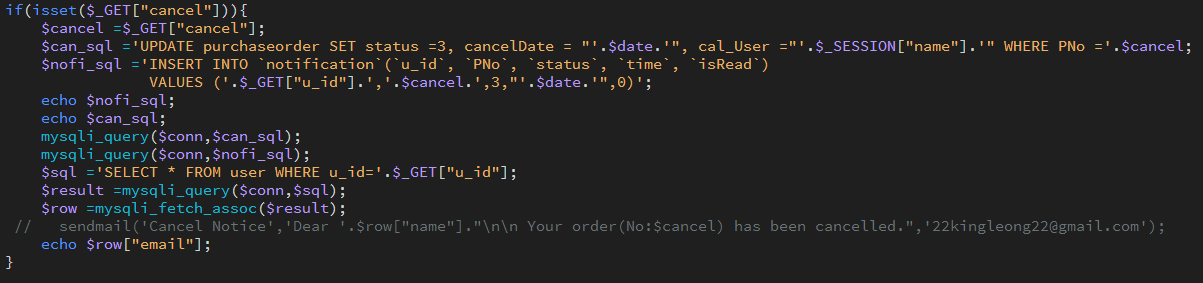


Figure 19: Code of Order Cancel

The Cancel function order shifts the order status from pending (1) or on hold (4) to cancelled (3). Therefore, the function checks whether the status of the order is pending (1) or held (4) before cancelling the order.

## Customers’ Ratings and Reviews

The shopping mall system provides customers with scoring and comment functions. After the seller ships the goods, customers can submit new ratings and reviews for each product of their order. The ratings and reviews of each customer will be stored in the PurchasedProducts database. The server will display the average rating and reviews for each product. The average rating will be stored in the product in the database. Comments will be stored in PurchasedProducts in the database. The average product rating is the sum of the ratings of each product divided by the number of customers that have been rated.

On the client-side, the customer must rate the ordered products. The viewing function is optional. Customers can view their order products according to their intentions.

The following code describes how the server processes submitted ratings and comments.

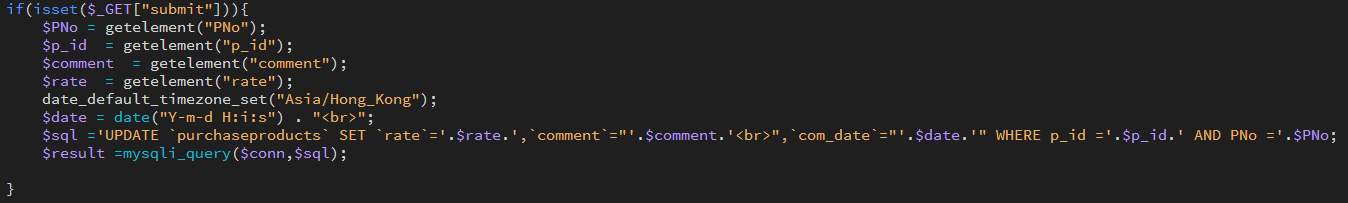


Figure 20:PHP code of Rating and Reviews



Figure 21: HTML code of Rating and Reviews

## Inventory checks

The shopping mall system has inventory checks. Before placing the customer's order, the system will check whether the product has a sufficient amount of inventory to satisfy the customer's order of product quantity. If not, the system will prohibit the supplier’s shipping order. For goods, the product inventory in the database product will use the number of products ordered by the customer in the purchased products in the database. It also updates the product inventory in the database product.

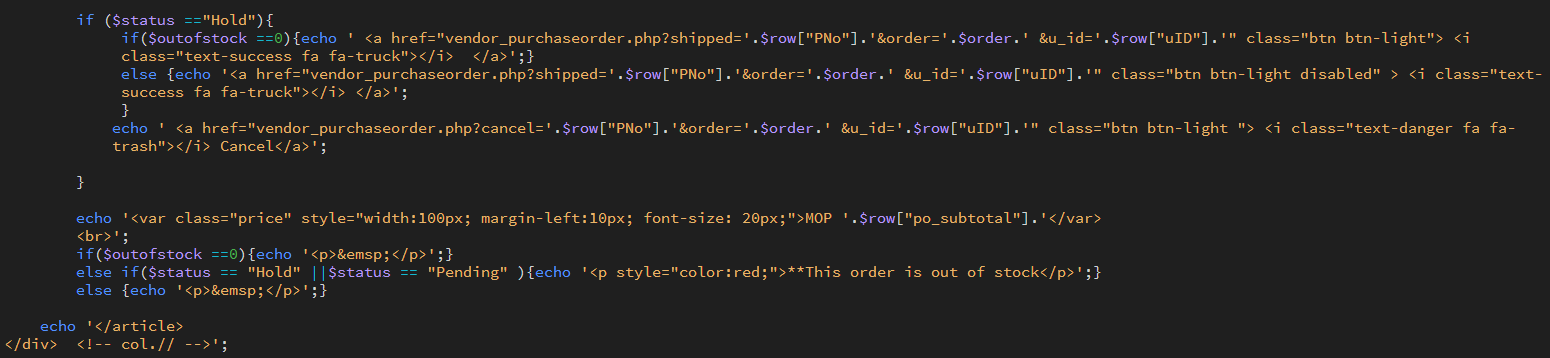


Figure 22:Code of check out of stock

## Inventory checks

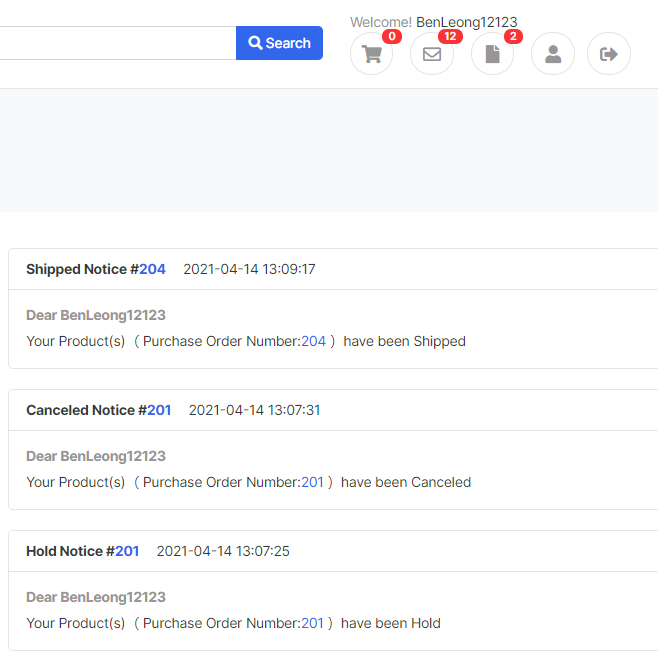
The notification function is a very user-friendly function for customers. This function has been implemented in this system to support customers to track changes in the status of purchase orders. In this system, the function is real-time notification, which means that when any status of the purchase order changes, the customer will be notified immediately. The GUI of the notification function is shown in the figure below. 

Figure 23: Graphic User Interface of Notification

This function is divided into two parts: client and server. The basic principle of this function is polling, which means that every time the client refreshes the screen, it sends a request to the server to check whether there is a new notification during this period.

The following figure informs the updated code. The main purpose of this code is to request the server to check whether there is a new notification every time the page is refreshed. If there is a new notification, the data will be updated on the client-side. There are two reasons for each refresh request. First, if the request frequency is too frequent, this will affect the user experience of real-time notification. Besides, there is no need to repeatedly request the server, because the improvement of the user-friendly experience is small, and subsequent operations will put more pressure on the server.



Figure 24: Graphic User Interface of Notification

# 5. Results and Discussion

In this section, we will discuss the results of some screen shots of online shopping centres.



## Project Outcome

This project has realized an online shopping system. In this chapter, the results of the entire system are shown in screen captures.

This project covers 5 advanced requirement-based functions. The functions of the 40 are divided into ten regions, and the number of the following is shown in brackets: product list (5), the product details page (2), account management (5), a shopping cart (6), procurement tracking (4), product catalogue maintenance (6), purchase order processing (7), customer rating and comment(2), notification (1), Price and inventory control (2). Based on these 10 areas, the results of the project are as follows.

### *Product List*

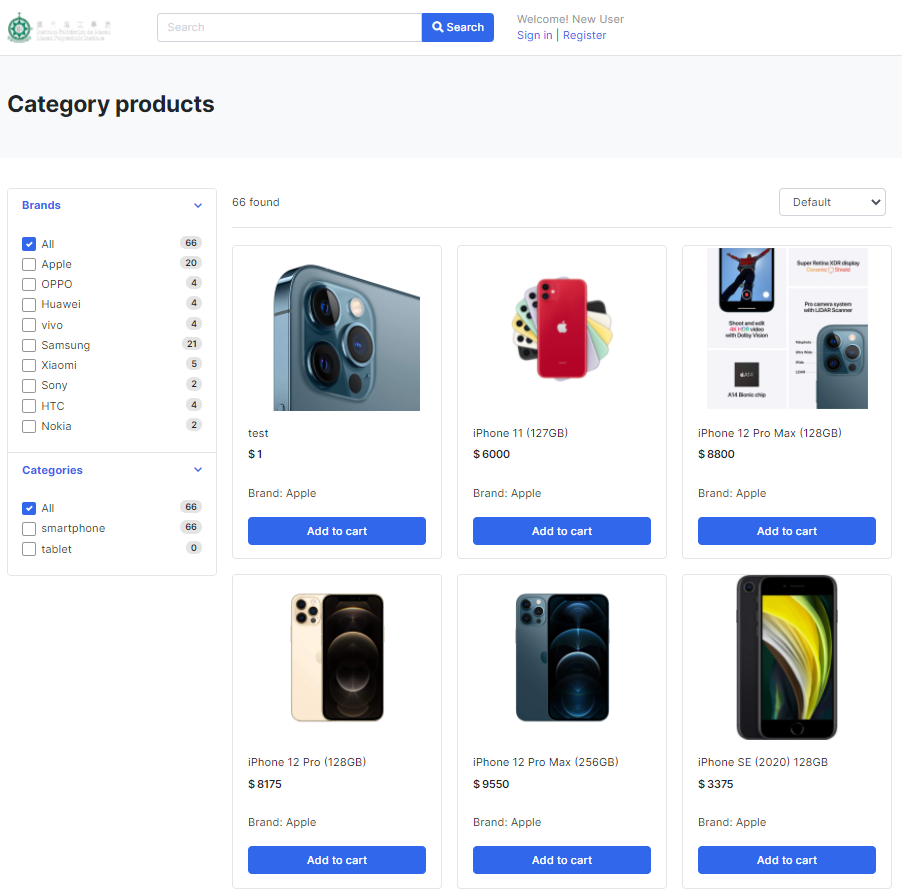


Figure 25:Figure of Product list

On the product list page, users can view the product list and part of their information, and achieve filtering and searching functions. Users can find what they are looking for through brand filtering or keyword search. Besides, it supports automatic paging at the bottom of the list, and sorting is not lost until the page changes.

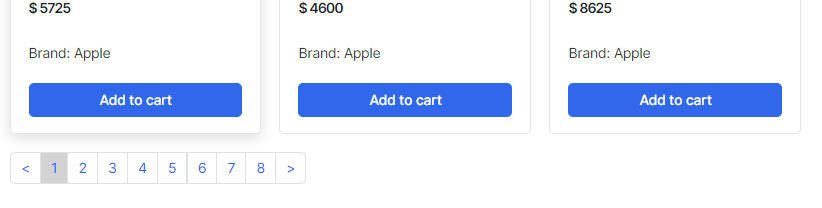


Figure 26:Figure of the product list paging

When the user clicks a specific page number, the page goes to that page. It also supports changing pages by clicking the page-up (<) or page-down (>) buttons.

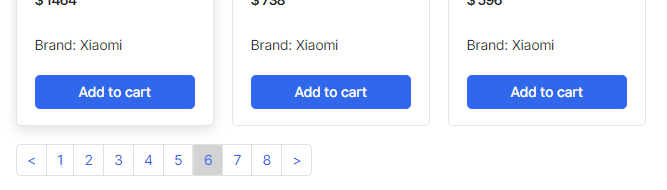
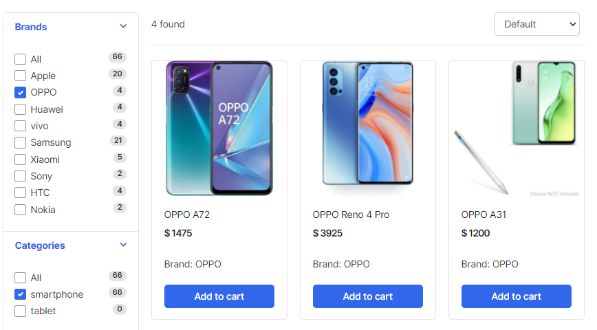


Figure 27: Figure of changing the product list paging

When a user clicks a brand name in the brand and category filter list, the product list content only displays the products that match the brand and category filter.

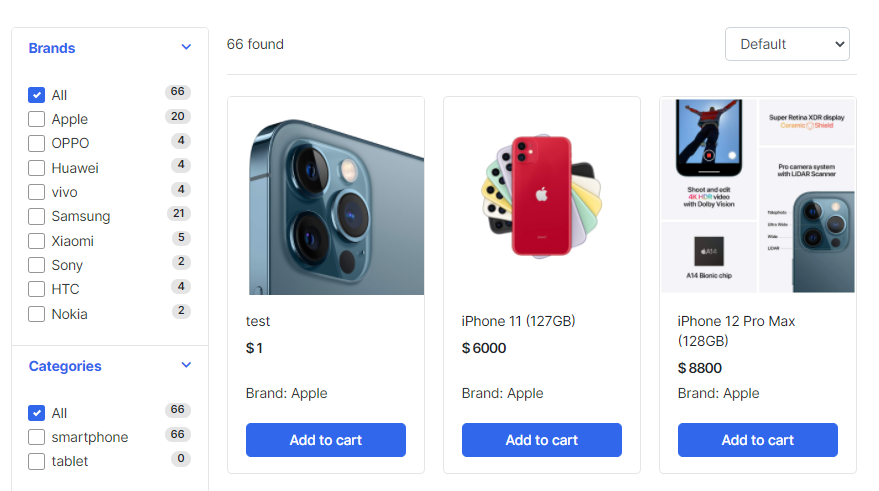
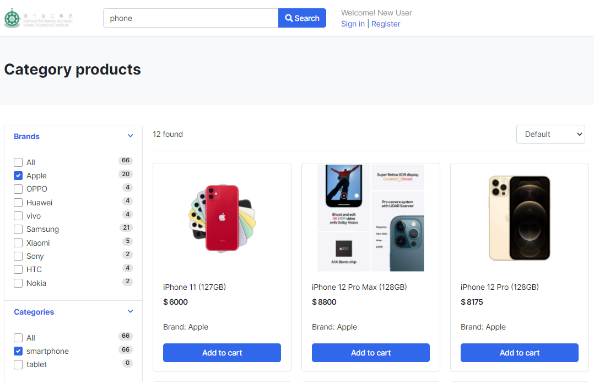
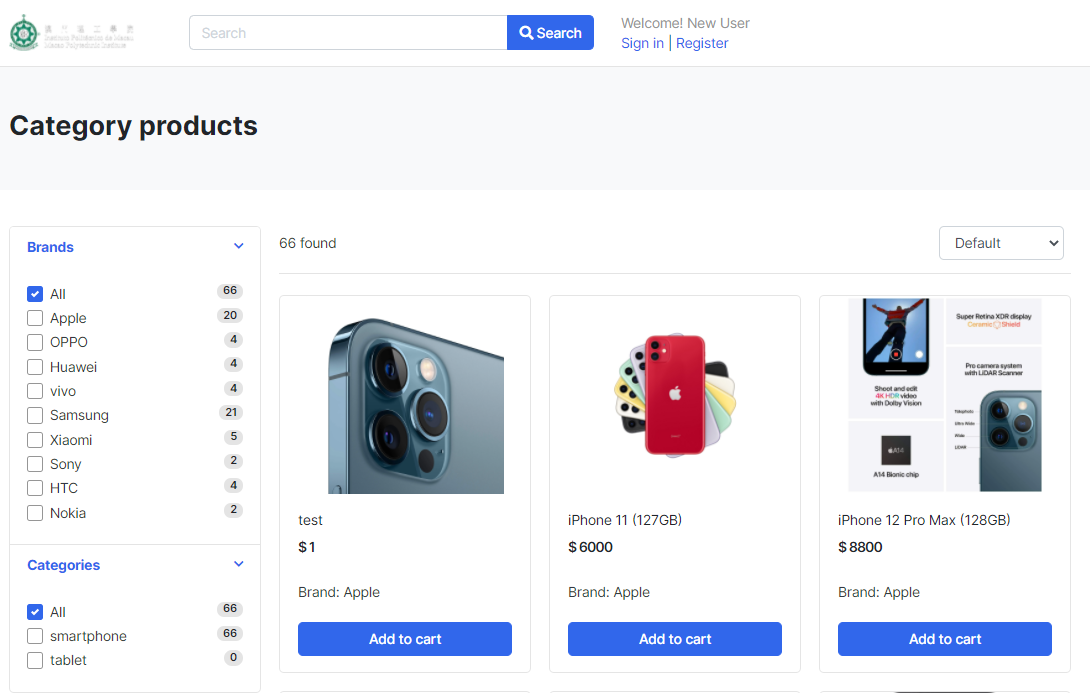


Figure 28: Figure of comparison of product list filtering

Users can search for products by entering a keyword of product names.

### Product Detail Page

When the user clicks on a specific product name, the page is redirected to the product details page. This page displays information about the products and product average ratings and comment, and the date of the comment.

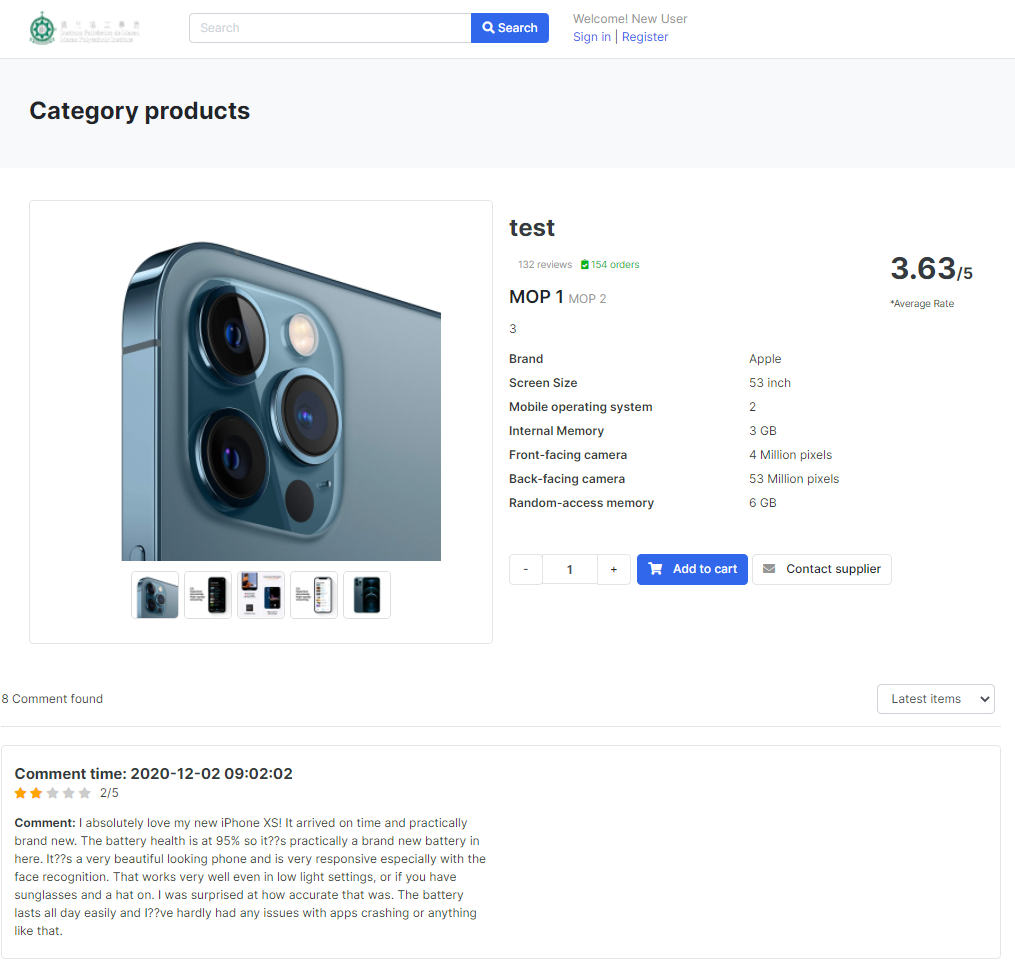


Figure 29: Figure of Product Detail Page

### Account Management

Visitors can register a user account by clicking on the "Register" button to the right of the top bar, and then enter some personal information and a password to register.

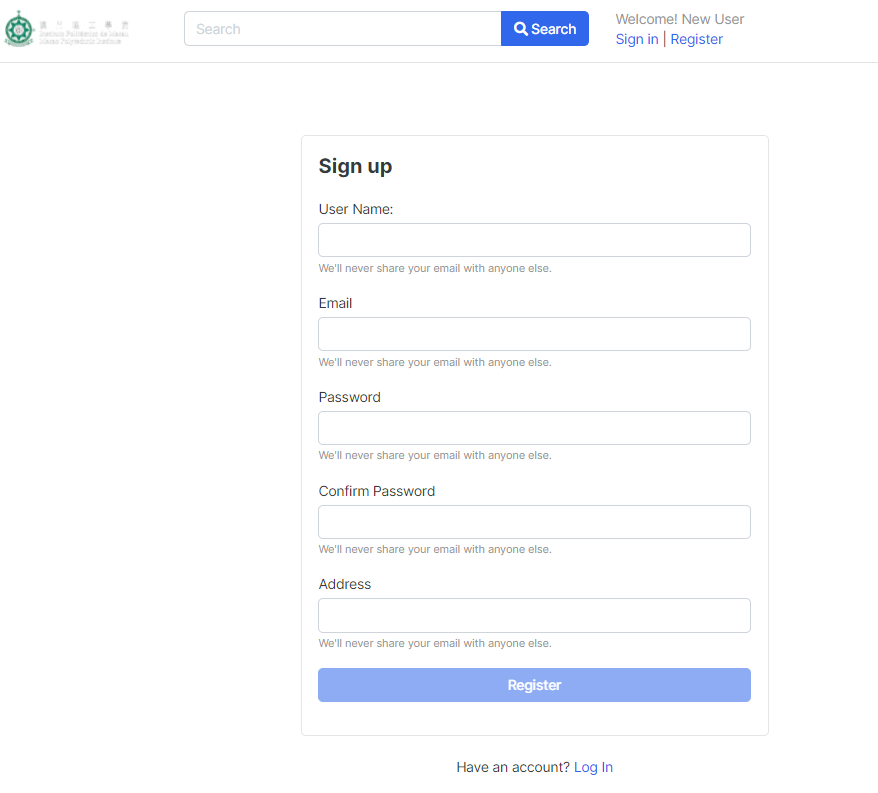


Figure 30: Figure of Registration Page

Each text box has a different validation to ensure that visitors are typing in the correct format.

If the format is wrong, the server returns an error message to the visitor.

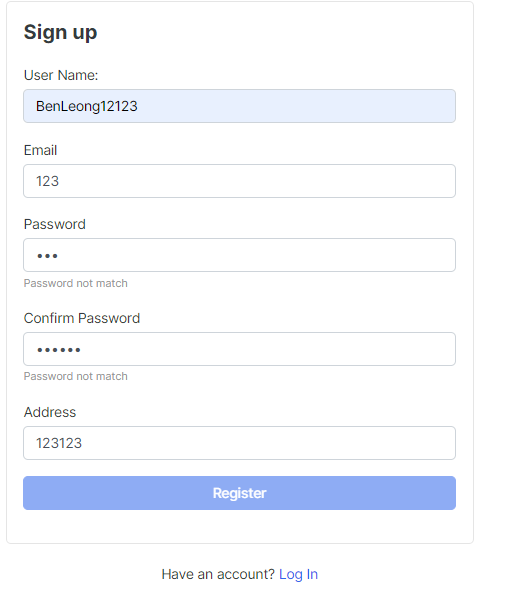


Figure 31: Figure of Error Message of Registration

Registered users can log in by clicking the "Login" button on the right side of the top bar, and then they can log in by filling in their user name and password, as well as login verification. After successful login, five buttons will appear on the right side of the top bar, including Notification, Order, Shopping Cart, Account Management, and Logout.

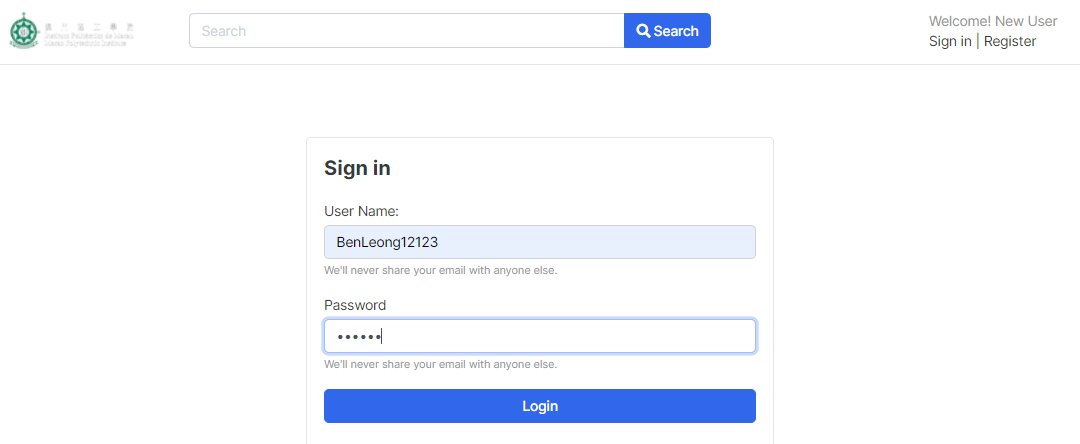


Figure 32:Figure of the Login interface

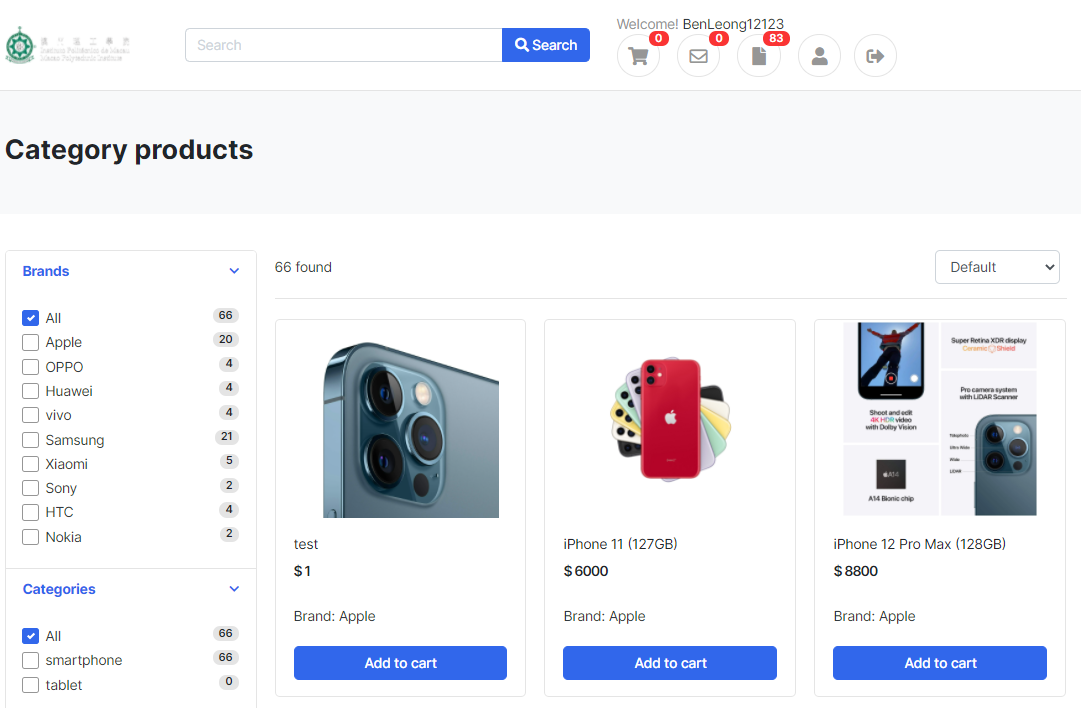


Figure 33:Figure of the interface after login

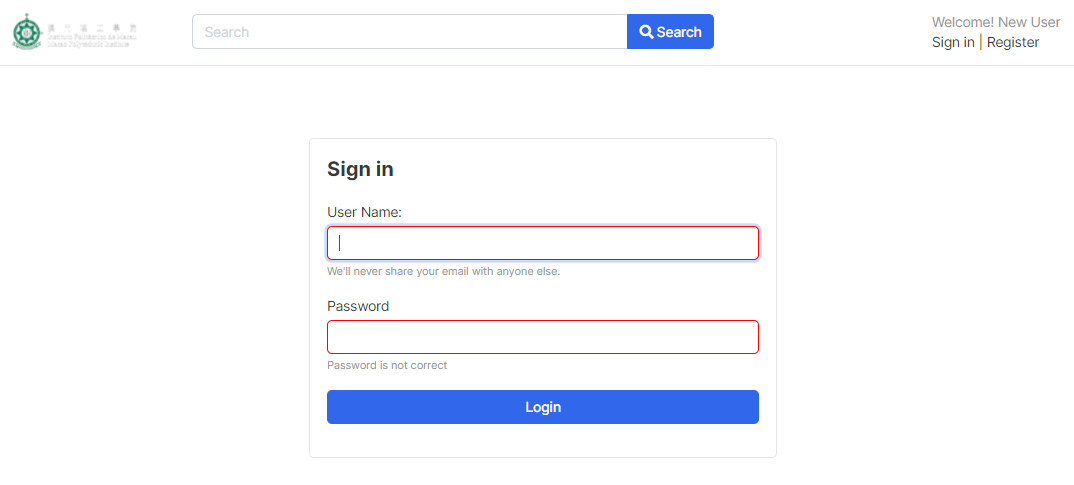


Figure 34:Figure of The Validation of Login

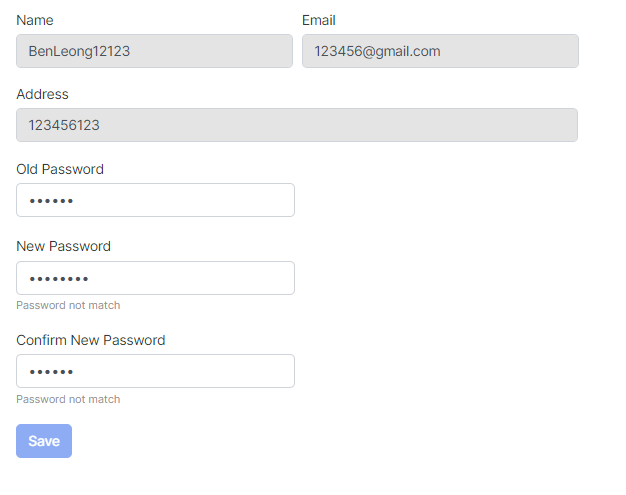


Figure 35:Figure of changing password

The change password system verifies that the user has entered the correct format.

After successful login, you will be redirected to the original page. If you log in from the product details page, you will be redirected to the product details page.

To protect password security, all passwords of users would hash before saved in the database.

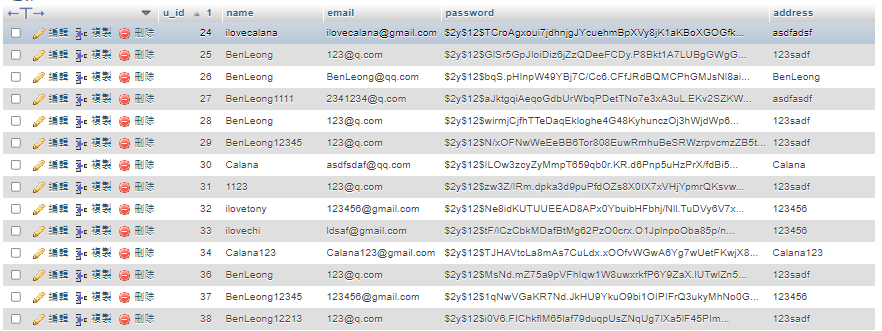


Figure 36: Figure of the database view

### Shopping Cart

Users can click the "Add to Cart" button on the product details page to add the product they want to the cart and the system will display a successful message.

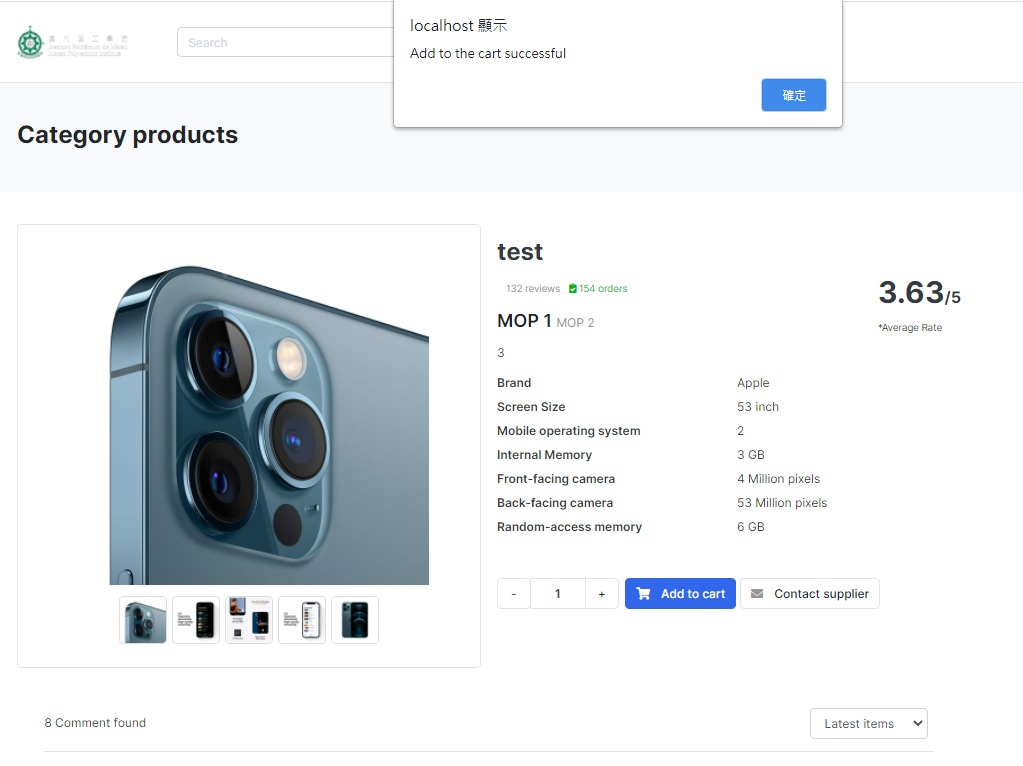


Figure 37: Figure of Successful Message after adding product to shopping cart

When the user successfully adds the product to the shopping cart, the "Shopping Cart" in the upper right corner will show the number of products in the shopping cart. Click to quickly view the contents of the shopping cart.

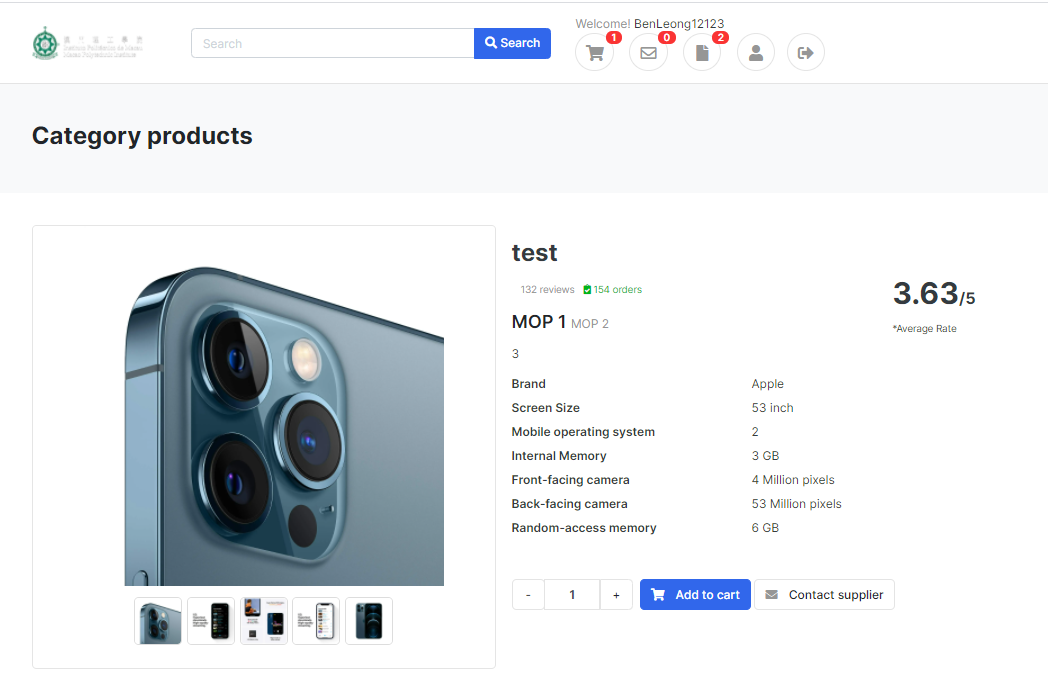


Figure 38: Figure that shows the number in shopping cart item

### Purchase Tracking

The purchase tracking page (Figure 39) allows users to view their orders. This information includes the order number, total amount, purchase date, and status. Users can view All order, pending and hold the order and shopped and cancelled using the filtering function. User can cancel orders through this page before they are shipped.

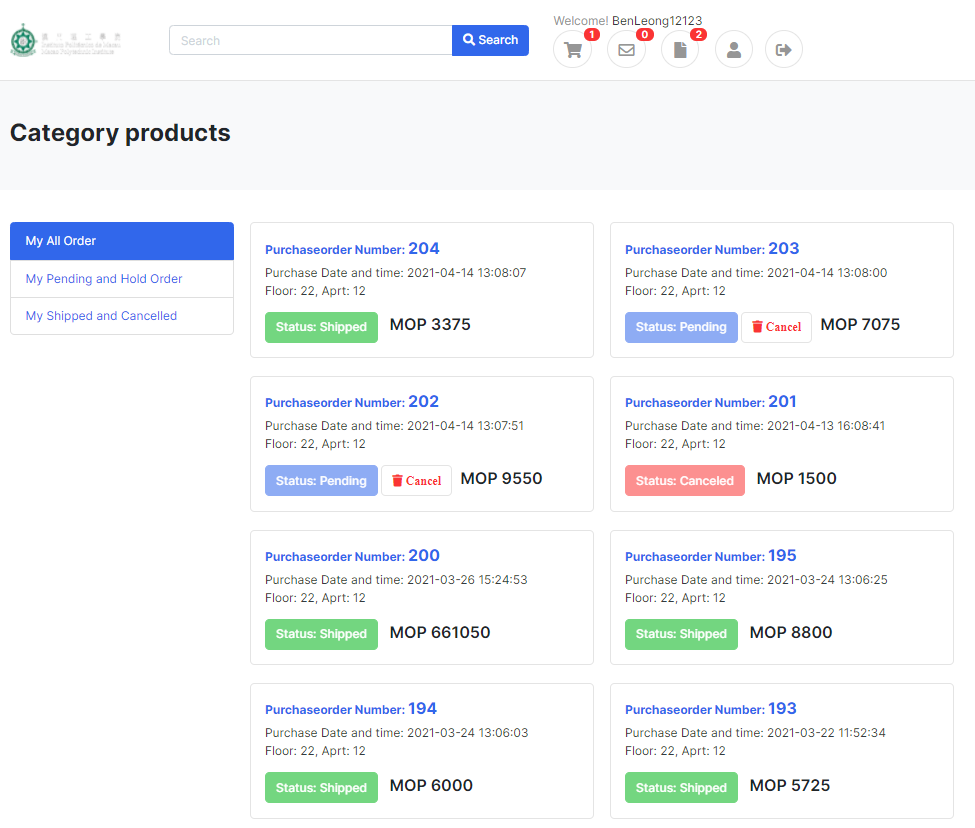


Figure 39: Figure of purchase tracking page interface

On the purchase order details page (Figure 40), the user can view the details of a particular order. The details include the information, subtotal, quantity, user email, user name, address, and unit price of each product in the order. Besides, the status, total amount, and date of the order are also included on the page. If the order has been shipped or cancelled, the date of shipment or cancellation is displayed.

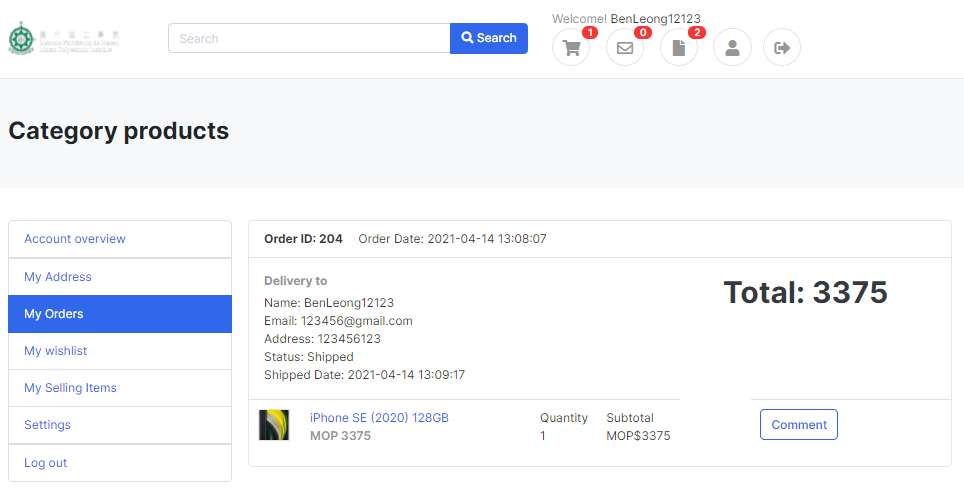


Figure 40:Figure of purchase order detail page

### Product Catalog Maintenance

Product catalogue maintenance provides vendors intending to add and edit products. The interface is similar to User, products can be filtered through search or filters, but has an extra function to add products. The interface is shown in the figure below.

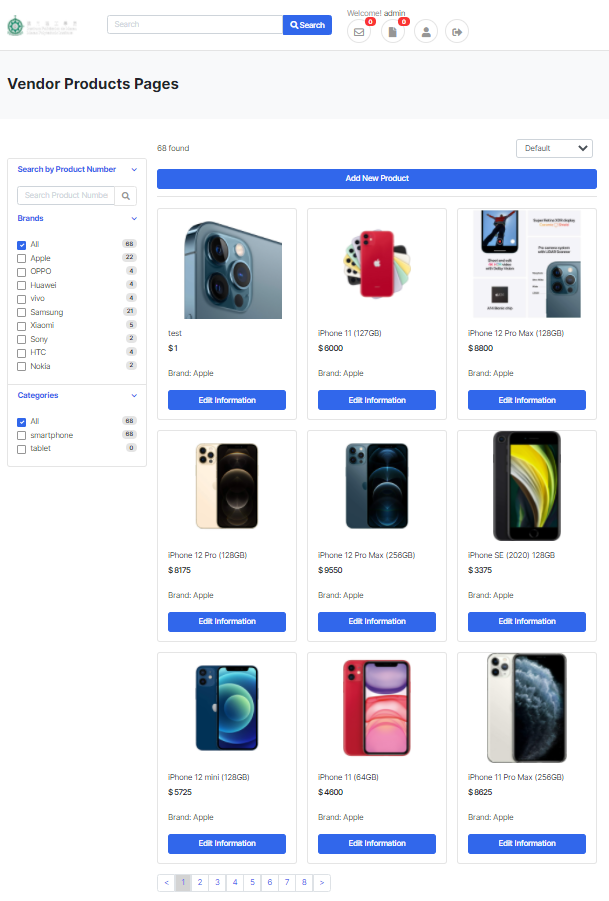


Figure 41:Figure of Product Catalog Maintenance

Each page can display up to 9 products, and filtering and search functions are still available on the vendor side. The screen capture below describes the functionality in more detail.

Vendors can search for products by product ID, title, brand, category.

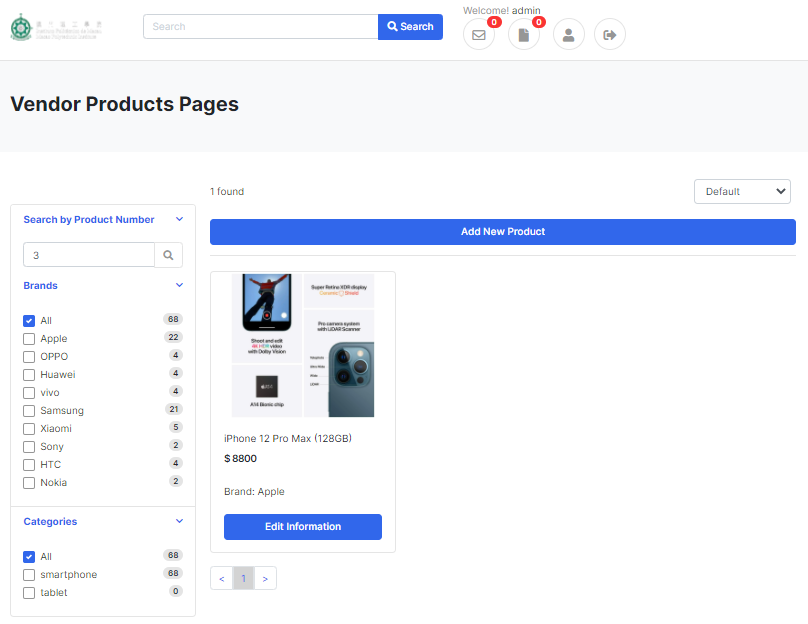


Figure 42:Figure of Vendor Searching by Product Number Function

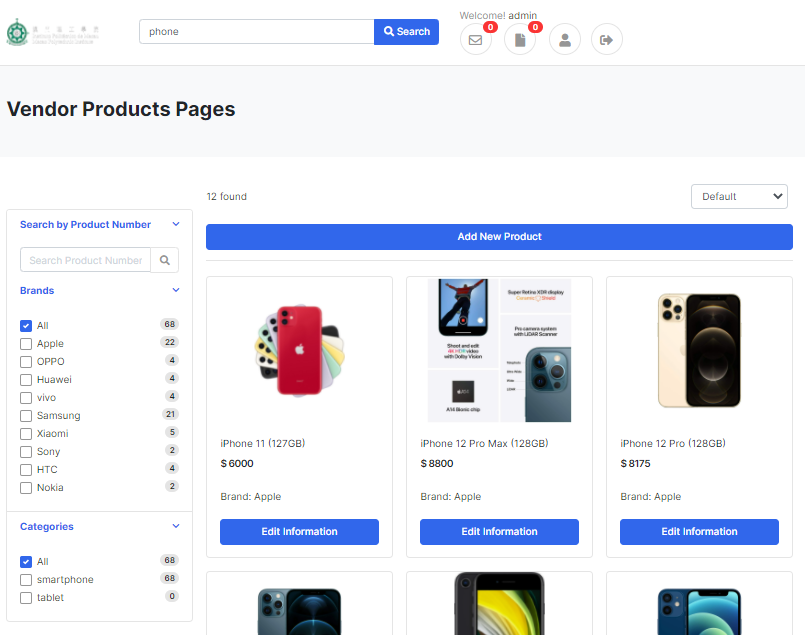


Figure 43:Figure of Vendor Searching by Product Keyword Function

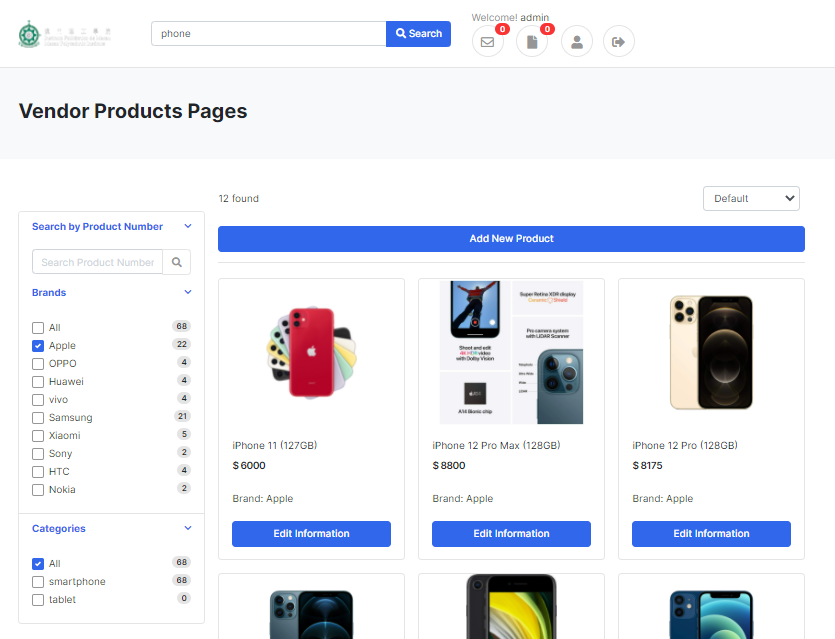


Figure 44:Figure of Vendor Searching by Product Keyword and Brand fitter Function

**Add new Products**

The vendor can click on new products through this page.

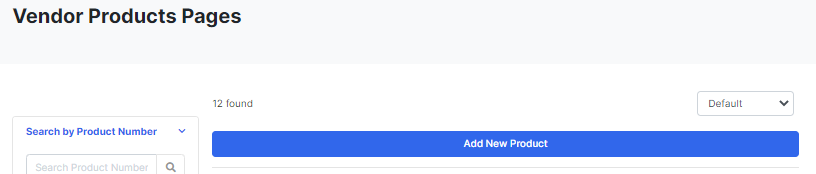


Figure 45:Figure of Add New Product button

After clicking the "Add New Product" button, the Vendor can input product information and upload thumbnails and detailed pictures of products.

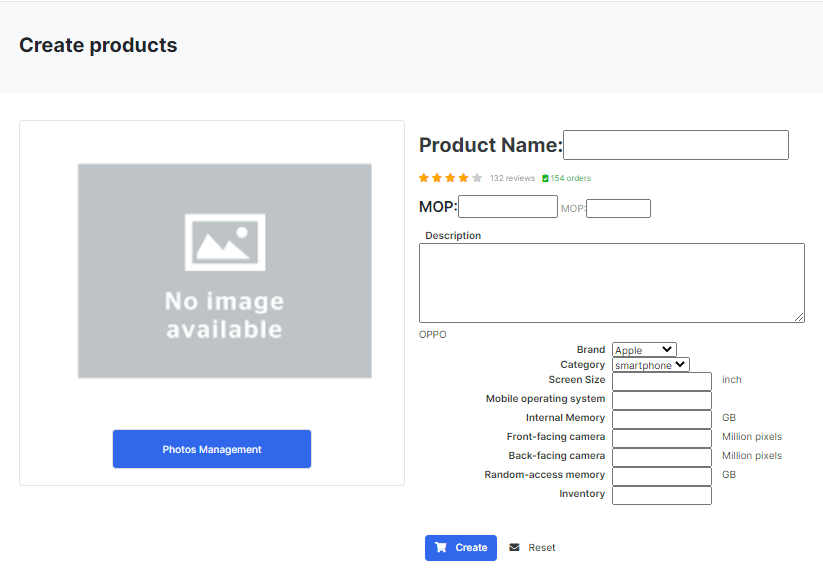


Figure 46:Figure of Add New Product page

**Edit Products**

Vendors can modify existing products through “Edit Information” button. Vendors can modify their information and pictures.

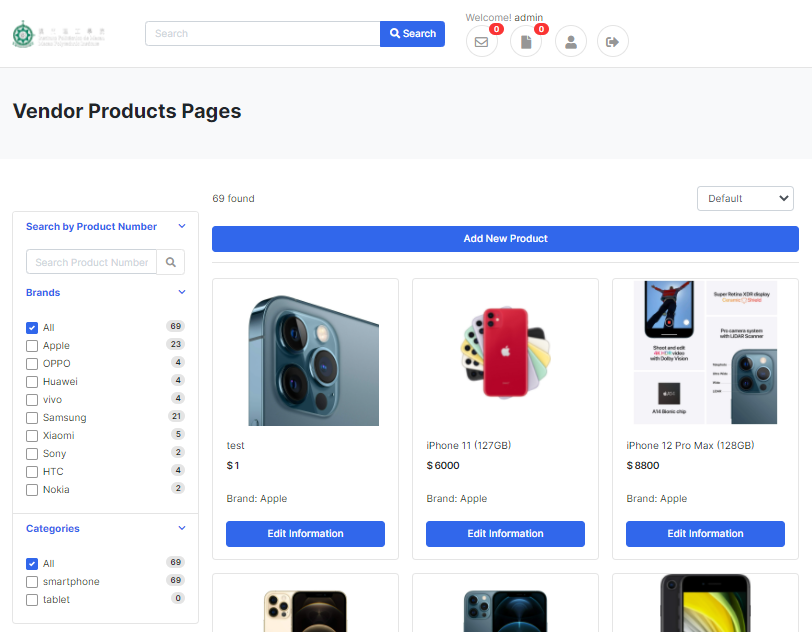


Figure 47:Figure of Edit Information button

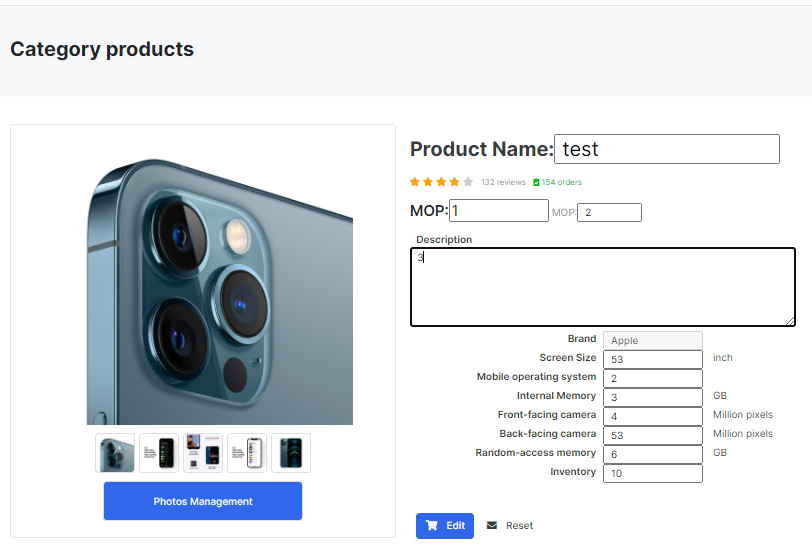
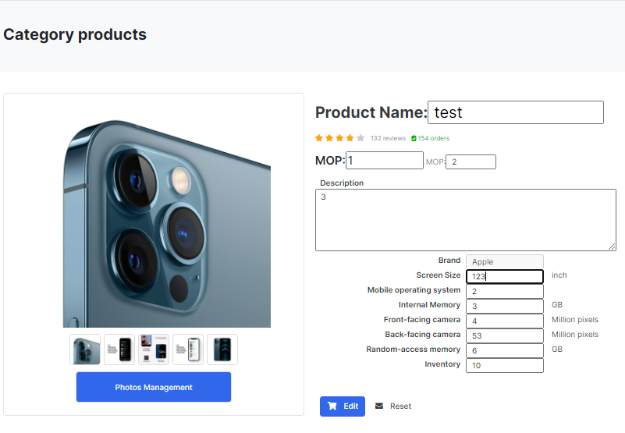


Figure 48:Figure of Edit information interface

### Purchase Order Processing

Purchase order processing helps vendor view and manage orders. In the purchase order list page (Figure 49), the Vendor can view the entire order list and filtering to view orders in various states. The vendor uses this page to hold, cancel, ship or cancel a particular order.

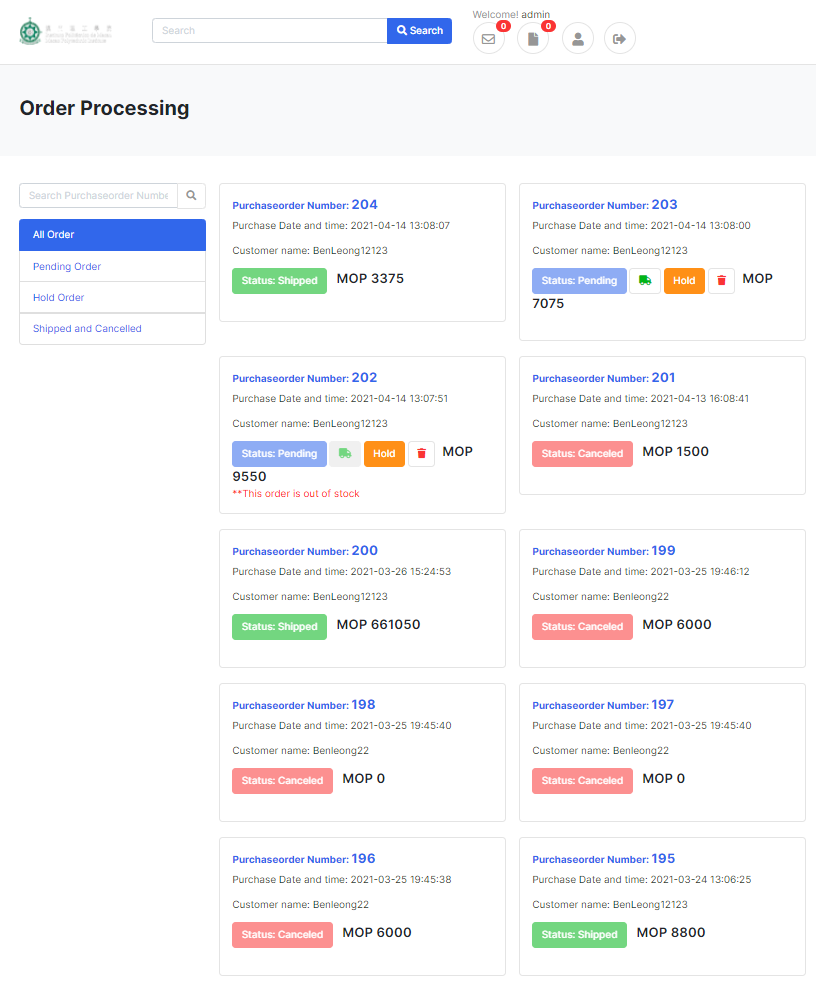


Figure 49:Figure of order processing

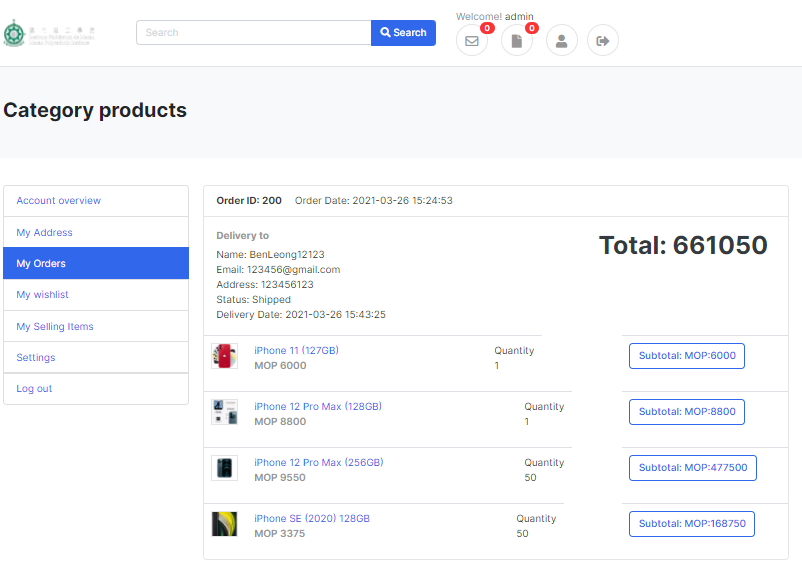
The screen capture of the purchase order processing page (Figure 50) is shown below. By using this page, vendors can view the details of a particular order. All products in this order and order information are listed on this page. 

Figure 50:Figure of Product Detailed Order Page

### Customer Rating and Comment

Before the customer gives a review and comment, he/she should order at least one of the products. When the order was shipped, the customer can go to the corresponding product to rate and write a comment.

When an order has shipped, the system will send a notification to remind the customer. Customer will be redirected to a comment page by pressing on the product link.

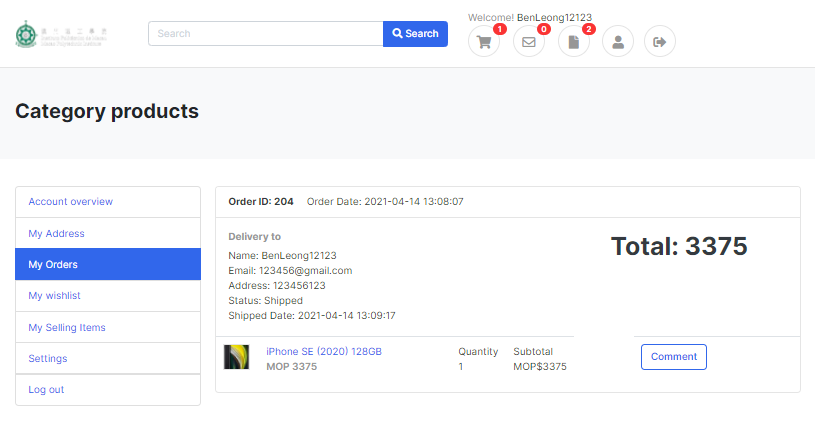


Figure 51: Figure of Product Detailed Order Page of User

After clicking the “comment” button, customers can give ratings and comments on this page (Figure 52).

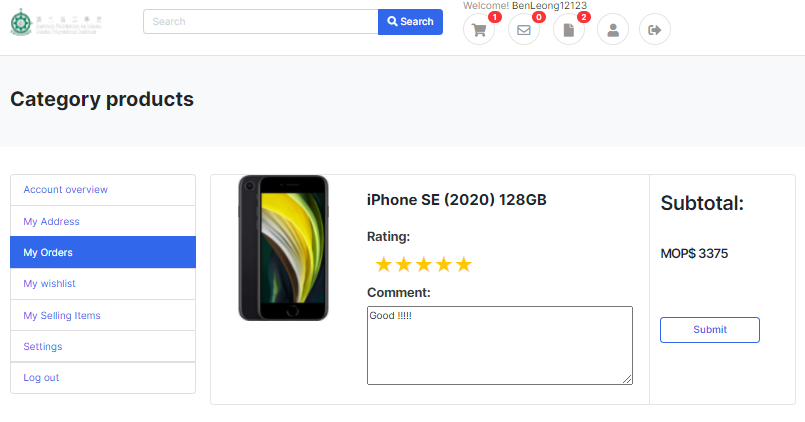


Figure 52: Figure of comment page interface

After submitting the comment, the data updates automatically.

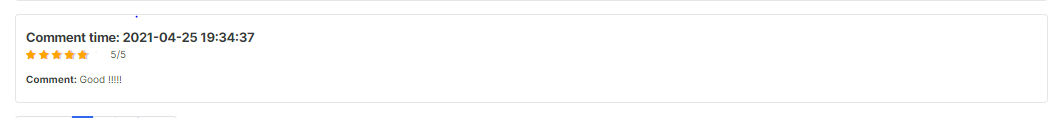


Figure 53:Figure of The Customer Rating and Review

### Notification

Notification allows users to track status changes of orders in real-time. After logging in, the user can see the message identifier on the screen, and if there is a new message, a red number will appear on the right side of the letter. Users can also click on the message flag to view unread messages. Users can click on the message to read the message (jump to the notification page) to change the status of the message to read. The GUI is shown in Figure 54.

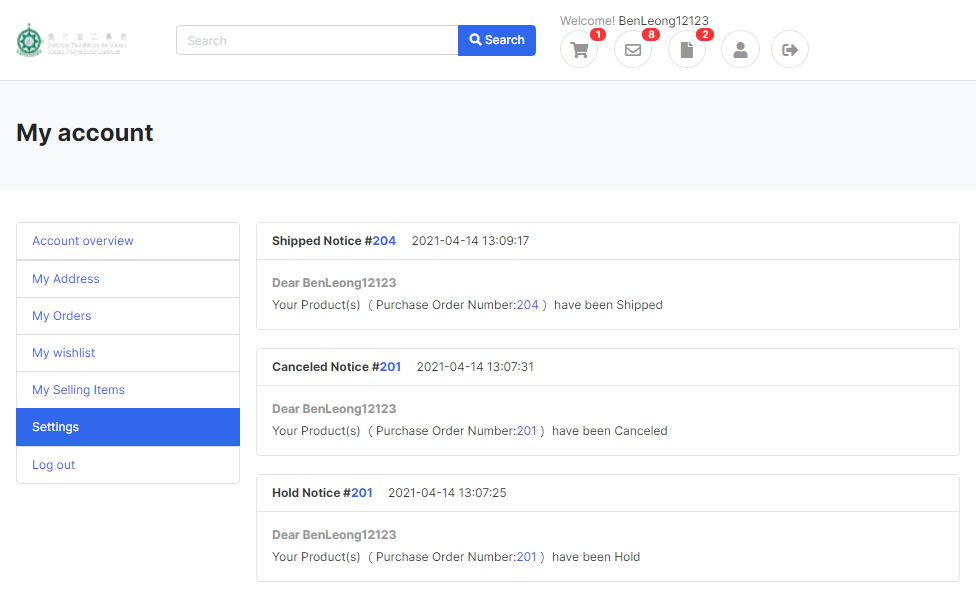


Figure 54:Figure of Notification Interface

By clicking the “Notifications” button, all notifications become read and the user can jump to the Notification List page to view all historical notifications. Besides, by clicking on a specific notification, the user can jump to a specific order. The screen capture is shown in Figure 54.

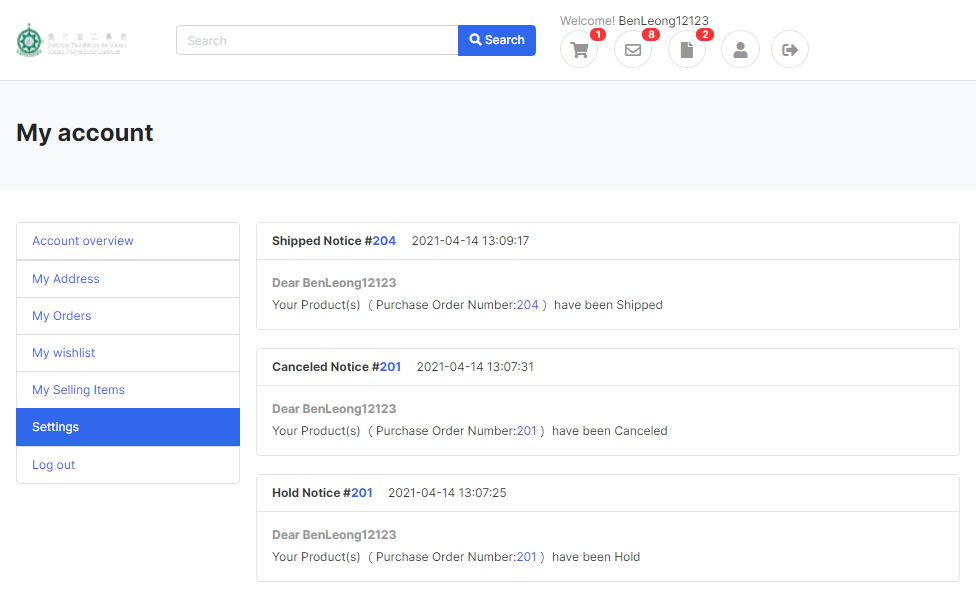
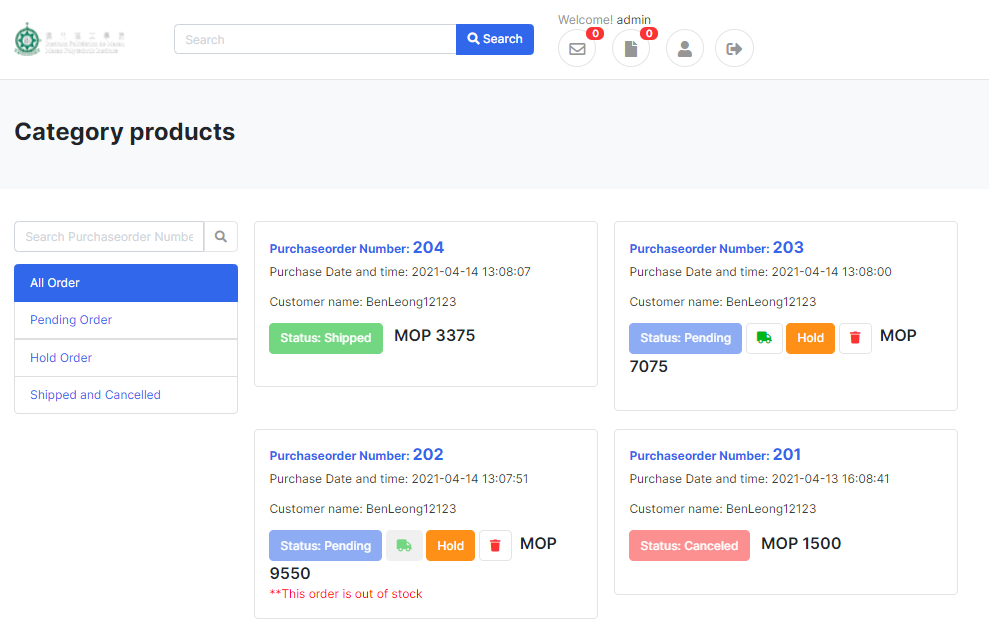


Figure 55:Figure of Notification Interface

### Inventory Control

When the stock falls below the quantity of a product, the supplier will not be able to ship. 

## Testing and System Evaluations

Table 3: Testing of requirements A1, A2, A3, A4, A5, A6, A7, C1, C6, B4

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case Number: | 1 | | |
| Test Case Name: | Product list | | |
| Requirements: | A1, A2, A3, A4, A5, A6, A7, C1, C6, B4 | | |
| Short Description: | The customer browses the product list. By clicking the product name, the customer can jump to the detailed page of the clicked product. On the product list page, customers can sort products by price, and filter products by brand and search criteria. On the product detail page, customers can view more information about the product, such as product description, one or more pictures, etc. Also, customers can add products to the shopping cart after logging in with their account. However, customers cannot add duplicate products to the shopping cart. | | |
| Pre-condition: |  | | |
| Function | Action | Expected System Response | Pass/Fail |
| 1 | Browsing the product | All products will appear on the list of products on the work page. | Pass |
| 2 | Filtering by brand | All products filtered by the brand are displayed in the product list. | Pass |
| 3 | Filtering by searching keywords | All products filtered by the search keyword will be displayed in the product list. | Pass |
| 4 | Sorting by price | All products sorted by price will be shown on the product list | Pass |
| 5 | Clicking the product name | The system directs the user to the product details page. | Pass |
| 6 | Viewing the detail of the product | The product details are shown here. | Pass |
| 7 | Adding product into the cart | The system checks whether the user is logged in. If the user is logged into the system, the user can add the product to the shopping cart. | Pass |
| 8 | Adding duplicate product | The system will reject products that are already stored in the purchase. | Pass |
| 9 | Redirect the login page | If a user without an account wants to add a product to the cart, the system will jump to the login page. | Pass |

Table 4: Testing of requirements B1, B2, B3, B5

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case Number: | 2 | | |
| Test Case Name: | Account management | | |
| Requirements: | B1, B2, B3, B5 | | |
| Short Description: | Customers should have their accounts before purchasing products. Customers can register a new account on the registration page. The customer logs in to the system using the account on his login page. On the customer's homepage, he/she can change his/her password, which contains at least 6 characters. | | |
| Pre-condition: |  | | |
| Function | Action | Expected System Response | Pass/Fail |
| 1 | User sign in | Use the registered account to successfully log in to the system. | Pass |
| 2 | User sign up | Users register with the appropriate name and password | Pass |
| 3 | Change password | The original password for the user is the same as the password in the database. The new password must contain at least 6 characters (at least 1 digit and 1 capital letter). | Pass |
| 4 | Storing hash value in database | The user's password is stored in the database by the server using a hash value. | Pass |

Table 5: Testing of requirements C2, C3, C4, C5

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case Number: | 3 | | |
| Test Case Name: | Shopping cart | | |
| Requirements: | C2, C3, C4, C5 | | |
| Short Description: | After the customer adds the product to the shopping cart, he/she can go to the shopping cart to browse the products to be added. On the shopping cart page, customers can modify the products they have added, such as changing the number of products and deleting products. Besides, customers can check out all items in the shopping cart. | | |
| Pre-condition: |  | | |
| Function | Action | Expected System Response | Pass/Fail |
| 1 | Browsing the shopping cart | All added products will be displayed in the shopping cart. | Pass |
| 2 | Check out | The shopping cart will remove all items. The system will generate a purchase order message to the user. | Pass |
| 3 | Changing the quantity of an item | The database successfully changed the number of items. | Pass |
| 4 | Removing an item | The database successfully deleted an item | Pass |

Table 6: Testing of requirements D1, D2, D3, D4, Z3, Z4

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case Number: | 4 | | |
| Test Case Name: | Purchase tracking list | | |
| Requirements: | D1, D2, D3, D4, Z3, Z4 | | |
| Short Description: | Customers can view all their orders on the purchase tracking page. Customers can click the button to filter by current and past. By clicking on the order number, the page will jump to the purchase order details page. The customer can click the "Cancel Order" button to cancel the order and then send the purchase order. If the order has been shipped, the customer can comment and rate each product in this purchase order. | | |
| Pre-condition: |  | | |
| Function | Action | Expected System Response | Pass/Fail |
| 1 | Browse purchase tracking list | All purchase orders of the user will be displayed in the user's purchase tracking list. | Pass |
| 2 | Clicking the button of the current order | The purchase orders with the status "Pending" and "Hold" will be displayed on the user's purchase tracking list. | Pass |
| 3 | Clicking the button of the past order | The purchase orders with the status of "shipped" and "cancelled" will be displayed on the user's purchase tracking list. | Pass |
| 4 | Clicking order number | The system will direct the user to the purchase order detail page. | Pass |
| 5 | Viewing the detail of the purchase order | The details of the purchase order will be displayed here. | Pass |
| 6 | Cancel order | The user cancels the order before it is shipped. | Pass |
| 7 | Commenting the product | The user comments on the product after the order is shipped. | Pass |
| 8 | Rating the product | Users rate the product after shipping. | Pass |

Table 7: Testing of requirements E1, E2, E3, E4, E5, E6, Z5

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case Number: | 5 | | |
| Test Case Name: | Product catalogue | | |
| Requirements: | E1, E2, E3, E4, E5, E6, Z5 | | |
| Short Description: | On the supplier product list page, the supplier can browse the product catalogue. Suppliers can find products by searching some keywords in the product name in the search bar. By clicking the delete button, the supplier can delete the selected product. By clicking the Create button, the supplier can add a new product. By clicking the edit button, the supplier can edit the selected product. When the supplier edits the product information, the original price will remain in the database. The product will be sold at the new price. By clicking on the product name, the supplier can view the details of the selected product. On the detail page, the supplier can also click the "Edit and Delete" button. | | |
| Pre-condition: |  | | |
| Function | Action | Expected System Response | Pass/Fail |
| 1 | Browsing the product list | All products will be displayed on the product list that can be paged. | Pass |
| 2 | Searching product | All products filtered by search keywords will be displayed on the product list. | Pass |
| 3 | Adding new product | The vendor can add a new product. | Pass |
| 4 | Editing information of the product | The supplier successfully updated the product. | Pass |
| 5 | Removing existing product | The supplier successfully deleted the product. | Pass |
| 6 | Viewing detail information of product | The vendor view product details | Pass |
| 7 | Uploading photos | Suppliers can upload 1-4 photos in each product. | Pass |
| 8 | Change photos | You can add and delete photos in each product. | Pass |
| 9 | Changing price | The original price will remain in the database. The system will use the new price for sales | Pass |

Table 8: Testing of requirements F1, F2, F3, F4, F5, F6, F7, Z6.

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case Number: | 6 | | |
| Test Case Name: | Purchase order list | | |
| Requirements: | F1, F2, F3, F4, F5, F6, F7, Z6 | | |
| Short Description: | Suppliers can browse the list on the purchase order list page. The order list can be filtered by each status, such as pending, reserved, shipped and cancelled. Suppliers can enter an existing order number in the search bar. By clicking on the order entry, the system can jump to the purchase order processing page. On the purchase order processing page, the supplier can view all products of each order. Suppliers can cancel, reserve, unreserve and ship purchase orders. If the product quantity is greater than the product inventory, the seller cannot ship and holds the order. | | |
| Pre-condition: |  | | |
| Function | Action | Expected System Response | Pass/Fail |
| 1 | Browse the purchase order list | All purchase orders will be displayed on the purchase order list. | Pass |
| 2 | Filtering by status | All purchase orders filtered by the brand will be displayed on the product list. | Pass |
| 3 | Enter the order number | After entering the order number, the system resets the supplier and is directed to the purchase order processing page. | Pass |
| 4 | By clicking the shipped order button | The status of the purchase order is changed from "pending" to "shipped". | Pass |
| 5 | By clicking the hold order button | The status of the purchase order is changed from "Pending" to "Hold". | Pass |
| 6 | By clicking the cancel order button | The status of the purchase order was changed from "Cancelled" to "Delivered". | Pass |
| 7 | By clicking the ship order button | The status of the purchase order is changed from "hold" to "shipped". | Pass |
| 8 | Marking ‘out-of-stock’ | If the product quantity is greater than the product inventory in the purchase order, the system will add a "shipped" button. | Pass |

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case Number: | 7 | | |
| Test Case Name: | Notification | | |
| Requirements: | Z2 | | |
| Short Description: | If the supplier processes the customer's purchase order, the system will notify the user within a few minutes. Customers can click the notification to jump to the purchase order details page. By clicking the "View All" button, the customer can see all notifications in which the seller changed the status of the purchase order. The notification distinguishes between reading and unread status. | | |
| Module: | Notification | | |
| Pre-condition: |  | | |
| Function | Action | Expected System Response | Pass/Fail |
| 1 | Sending the message | After the supplier changes the status of the user's purchase order, the system will notify the user. | Pass |
| 2 | Viewing all messages | Users can view all notifications notified by the system | Pass |
| 3 | Reading message | The system will guide the user to the purchase order details page. | Pass |

# 6. Conclusion and Further Work

In conclusion, we have successfully implemented 35 essential functions and five advanced functions in our project. First, our scheme provides a user-friendly interface for users. Second, customers can view the product on the marketplace page and search for the product by entering the product name's keyword. They can then view specific product details, including product details, promotions, customer ratings, and reviews, and add the product to the shopping cart. Besides, customers can check out on the shopping cart page. When the vendor ships the product, customers will receive a notification, and they may give some reviews and ratings to the product at the end of the purchase process.

Our project has a contract with a web hosting center to ensure that the web page is secure and reliable. The first reason is that our web pages support HTTPS, a secure protocol. Besides, our website is backed up once a day to ensure security. Our project can go a long way towards ensuring the site's safety, but we also have much space for improvement, such as our project only stores data in one database at the moment. This is very dangerous in the real world because there is no data validation. If a hacker breaks into our database and maliciously changes the data, we cannot restore the database's correct data.

In the future, we look forward to enhancing the user experience, HCI, and implementing more features for the vendor and customers. First, we want to implement a new feature that allows vendors to change the order of detail images on the product edit page. Besides, we want customers to categorize product tables through more attributes to meet customer needs. For example, if they're going to know our project's latest product, they can sort by the product's release date.

On the other hand, our project is aimed at the Macao market. Macao's official languages are Chinese and Portuguese, but our project only has the English version. In the future, we will develop the Chinese and Portuguese versions, collect Macao people's data, and redesign a shopping platform more catered to Macao market. [4]

# Reference

|  |  |
| --- | --- |
| [1] | T. M. A. f. I. Research, “Internet Usage Trends in Macao,” The Macao Association for Internet Research, Macao, 2020. |
| [2] | “Taobao,” Alibaba, [Online]. Available: https://world.taobao.com/. [Accessed 23 4 2021]. |
| [3] | J. Bezos, “Amazon,” Amazon, [Online]. Available: https://www.amazon.com/ref=nav\_logo. [Accessed 23 4 2021]. |
| [4] | B. Harwani, Make an E-commerce Site in a Weekend: Using PHP, 2015. |
| [5] | M. Isherwood, Designing Ecommerce Websites Paperback, 2017. |
| [6] | R. Earp, Database Design Using Entity-Relationship Diagrams (Foundations of Database Design Book 1), 2007. |
| [7] | Song, Il-Yeol & Whang, Kyu-Young & Korea, “Database Design for Real-World E-Commerce Systems,” *DBLP,* 4 2000. |
| [8] | R. Stephens, Beginning Database Design Solutions, 2008. |

# Appendix A. Project Management

# Product scope

The section below illustrates the 35 requirements distributed among 5 main function groups.

## Product scope

1. **[T1] Display product list**

We looked at different online shopping sites and found that they all share many of the same features, such as designing a list of products for customers to browse, searching, the ability to filter products by brand or value, and the ability to display product images. Therefore, we also refer to them and their functions to design this website

1. **[T2] Login/register function**

According to requirements, users need to log in to an account when purchasing goods. Users without an account need to fill in the basic information for registration (user name, password, email, and address) and then buy goods after registration.

1. **[T3] Product Page**

When users click on a specific product, they can enter the detailed web page of the product to view more information about the product, such as pictures, inventory, functions, etc. Users can also purchase products here or add them to their shopping cart.

1. **[T4] Purchase Tracking**

Users can find these orders in the order list. By clicking on a charge, Users can enter the order details page to view more status. For example, if an order has been shipped or canceled, both the supplier and the user have the right to cancel the order.

1. **[T5] Order Processing**

Suppliers can also process orders through the system. The order list provides a record of each order for each customer. Suppliers can filter these orders by order status. By clicking on a paper in the order list, the supplier can enter the detailed information page of a particular order. The supplier can ship the canceled order on the detail page, hold the order, and then cancel the order before shipping.

1. **[T6] Add/delict product features**

The system also provides some functions for suppliers to maintain their products. Suppliers can browse each product's information, create new products, modify the information and pictures of existing products, and delete certain products.

1. **[T7] Shopping cart**

Users can add favorite products to the shopping cart. In the shopping cart, customers can modify the quantity of each product, delete specific products, and then make a purchase.

1. **[T8] Advanced Function**

To improve the user experience, the project has some advanced features. For example, customers can rate and comment on products. For suppliers, they can also quickly know which products are hot-selling and out-of-stock products. The notification function allows customers to see the status of the order quickly.

## Process scope

Each main function includes one or more user requirements, which will be discussed below. The details more detailed of the requirements of each main function

|  |  |  |
| --- | --- | --- |
| Main functions | **[T1] Display product list** | |
| Requirements: | A1, A2, A3, A4, A5, A6, A7, C1, C6, B4 | |
| **Function** | **Action Name** | **Description** |
| A1 | Browse products | A customer may browse products in a list of products. The list shows the basic information of products, including product name, brand, price, and a thumbnail image. Each product belongs to one of the pre-defined brands. |
| A2 | Paging | The product list supports paging. The customer can navigate the product list by ‘page up,’ ‘page down’, and jumping to a specific page. Paging works appropriately after applying a filter or sorting, as listed below. |
| A3 | Filter | All products filtering by searching keywords will show on the product list. |
| A4 | Searching | The customer may filter the product list by searching keywords in the product name. This function works correctly with the brand filter. |
| A5 | Sort by price | The customer may sort the product list by price. |
| A6 | Product detail page | The customer may select a product in the product list to go to the product detail page. The product detail page shows information for one product, including the product name, brand, price, and a thumbnail image. The product detail page shows a detailed description as a list of at least two properties. For example, the product detail page for a book may show authors, ISBN, publisher, release date, and the number of pages. |
| A7 | One or more detailed photographs | The product detail page supports the display of one or more detailed photographs of the selected product. |
| C1 | Adds a product | The customer adds a product to his/her shopping cart by clicking a button on the product detail page. The quantity to buy is assumed to be 1. The items in the shopping cart are persisted across user sessions. Next time the customer logs in, they can still see the items in the shopping cart. |
| C6 | Duplicate product | If the customer adds a duplicate product to the shopping cart, the application will give a warning message and does not change the shopping cart's content. |
| B4 | Redirects | If a customer tries to add a product to the shopping cart on the product detail page without first logging in, the system redirects the user to the login page. After successful login, the system redirects the user back to the original product detail page. |

|  |  |  |
| --- | --- | --- |
| Main function | **[T2]Login/register function** | |
| Requirements: | B1, B2, B3, B5 | |
| **Function** | **Action Name** | **Description** |
| B1 | Register a new account | A customer may register a new account. They have to provide full name, email address, password, and shipping address. After registration, the user has logged in automatically. |
| B2 | Log in and log out | A customer may log in and log out, and the interface shows the current user's name. The product list and product detail page are accessible to customers without login. On the other hand, the shopping cart and purchase tracking are only accessible after login. |
| B3 | Change password | The customer can change the password. There is a vital requirement for a password. The password should contain at least six characters, in which there must be at least one digit and one capital letter. |
| B5 | Hash values | The server only saves hash values of customers’ passwords. Passwords are never saved in plain text on the server. |

|  |  |  |
| --- | --- | --- |
| Main functions | **[T3] Product Page and [T6] Add/Delite product features** | |
| Requirements: | E1, E2, E3, E4, E5, E6, Z5 | |
| **Function** | **Action Name** | **Description** |
| E1 | Browse the product catalog | The vendor may browse the product catalog in an interface similar to the product list for customers. The vendor is not a customer, and no shopping cart or ‘add to cart’ button should be shown. |
| E2 | Vendor searching | The vendor can find products by searching keywords in product names. They can also find a specific product by entering a unique product ID. |
| E3 | Add a new product | The vendor may add a new product to the catalogue. The vendor enters basic information of the product, including product name, brand, price, and a thumbnail image. They can enter detailed information about the new product as a list of properties. |
| E4 | Upload photographs | In addition to the thumbnail image, the vendor can upload 1 to 4 detailed photographs for a product. These photos are usually of higher resolution and are displayed on the product detail page in a user-friendly interface. (Refer to requirement A7) |
| E5 | edit information | The vendor can edit information about a product on a product detail page. They can change the product name and product brand. They can also change detailed information as a list of properties. (Refer to requirement E3). |
| E6 | Add or remove photos | The vendor can change the thumbnail and detail photos for a product. They can add or remove photos |
| Z5 | Price change | Design how to implement price change of products. This is useful, e.g., for a promotional price reduction or regular price adjustment. Price change should not affect the price in existing purchase order and other historical records. |

|  |  |  |
| --- | --- | --- |
| Main function | **[T4] Purchase Tracking** | |
| Requirements: | D1, D2, D3, D4, Z3, Z4 | |
| **Function** | **Action Name** | **Description** |
| D1 | Purchase tracking page | The purchase tracking page lists the purchase orders that the customer has placed. This page shows the following for each purchase order: the P.O. number, the purchase date, the total order amount, and the purchase order status. The purchase orders are displayed in reverse chronological order of the purchase date. When the customer clicks an entry in the list, they can see the detail in a purchase order detail page |
| D2 | Vendor filter | The customer can filter the list of purchase orders in two ways. First, the page only shows ‘current purchases’ with status ‘pending’ and ‘hold.’ Second, the page only shows ‘past purchases’ with the status ‘shipped’ and ‘canceled.’ |
| D3 | Purchase order detail page | The purchase order detail page shows the P.O. number, the purchase date, the customer name, the shipping address, the total order amount, and the purchase order status. If the order is shipped, this page shows the shipment date. If the order is canceled, the page shows the order cancel date and who (customer or vendor) canceled the order. The page also includes the product name, the quantity, the unit price, and the subtotal for each product in the purchase order. |
| D4 | Cancel the order | Before a purchase order is shipped, the customer can cancel the order. This can be done by clicking a button on the purchase order detail page. This action will change the status of the purchase order to ‘canceled’. Note that this action is only available for purchase orders in the status ‘pending’ or ‘hold’. |
| Z3 | Customer’s rating | A customer can express their satisfaction with a product with a customer’s rating. A customer who has purchased a product successfully can rate it on a scale of 1 to 5 stars after the purchase is shipped. The customer can only score the product once, and the product score is obtained by the average customer rating. |
| Z4 | Reviews for products | In addition to star ratings, customers also want to write short reviews for products in the shopping mall. |
| Main function | **[T5] Order Processing** | |
| Requirements: | F1, F2, F3, F4, F5, F6, F7, Z6 | |
| **Function** | **Action Name** | **Description** |
| F1 | Purchase order list page | The purchase order list page lists purchase orders received by the application. It shows the P.O. numbers, purchase dates, customer names, total order amounts, and purchase order status. The purchase orders are sorted in descending order of purchase date (i.e., newest first). The vendor can click an entry to open a purchase order processing page. |
| F2 | Status | The vendor can filter the purchase order list in three ways. They can show only the ‘pending orders’ (with status ‘pending’). They can offer only the ‘orders on hold’ (with status ‘hold’). Finally, the vendor can select to show ‘past orders’ (with status ‘shipped’ or ‘cancelled’). |
| F3 | Purchase order processing page | The purchase order processing page shows similar information as the purchase order detail page (refer to requirement D3). Besides, the vendor can click a button to ship a purchase order. This action changes the status of the purchase order from ‘pending’ to ‘shipped’ and starts the shipping process. |
| F4 | Purchase Order | The vendor can enter a P.O. number to view and process a specific purchase order. |
| F5 | Hold a purchase order | On the purchase order processing page, the vendor can click a button to hold a purchase order. This is useful, for example, if some product in the purchase order is temporarily out of stock. This action is only available when the status of the purchase order is ‘pending’, and this action changes the status to ‘hold’. |
| F6 | Hold and ship a purchase order | On the purchase order processing page, the vendor can click a button to unhold and ship a purchase order. This action changes the status of the purchase order from ‘hold’ to ‘shipped’ and starts the shipping process. |
| F7 | Cancel a purchase order | On the purchase order processing page, the vendor can click a button to cancel a purchase order. This is useful, for example, to inform the customer that the ordered products are no longer available. This action is only available for purchase orders in the status ‘pending’ or ‘hold’. This action changes the status of the purchase order to ‘cancelled’. |
| Z6 | Out-of-stock | Sometimes, a hot-selling product is ‘out-of-stock. The vendor can mark a product as temporarily out-of-stock and customers can still place orders for such products, but the system should only be able to ship them when the vendor marks the products as in-stock later. |

|  |  |  |
| --- | --- | --- |
| Main function | **[T7] Shopping cart** | |
| Requirements: | C2, C3, C4, C5 | |
| **Function** | **Action Name** | **Description** |
| C2 | List the products | The customer can list the products in his/her shopping cart on a shopping cart page. On this page, the entry for each product shows the product name, price and the quantity to buy. The page also shows the total order amount (i.e. how much the customer has to pay in total) in the shopping cart. The customer can click an item in the shopping cart to go to the product detail page of the entry. |
| C3 | Check out | The customer can press a button on the shopping cart page to check out all items in the shopping cart. This action creates a purchase order with a newly allocated unique P.O. number and clears the content of the cart. After checkout, the system shows the purchase order detail page of the newly created purchase order. (refer to requirement D3). |
| C4 | Change the quantity | The shopping cart page allows the customer to change the quantity of an item. This allows the customer to order more than one piece of a product (e.g. buy two copies of a book). |
| C5 | Remove an item | The customer can remove an item from the shopping cart. |

|  |  |  |
| --- | --- | --- |
| Main functions | **[T8] Advanced Function** | |
| Requirements: | Z2 | |
| **Function** | **Action Name** | **Description** |
| Z2 | Notification | Design a notification feature to make it easier for a customer to track the change of status of purchase orders. For example, when the vendor ships a purchase order, the customer will receive a notification message. The interface should distinguish between reading and unread notifications. |

## work-breakdown structure

|  |  |
| --- | --- |
| Analysis | Background study on the online shopping mall implementation of similar projects. |
| Collect detailed requirements |
| Design | Design the overall architecture. |
| Design database. |
| Design interfaces for web browsers. |
| Implementation | Develop interface templates for the supervisor to review |
| Create a site map or hierarchy chart showing the flow of web pages. |
| Create the individual web pages for the site. |
| Create a database for the server-site modules. |
| Sprint for [T1]\* |
| Sprint for [T2]\* |
| Sprint for [T3]\* |
| Sprint for [T4]\* |
| Sprint for [T5]\* |
| Sprint for [T6]\* |
| Sprint for [T7]\* |
| Sprint for [T8]\* |
| Deployment | Install the web site on the internet. |

\* The detailed steps are described in Schedule

# Schedule

Since the date of our meeting is Wednesday, therefore the report that the start date of the week is Wednesday and the end date is Tuesday.

\*\* This table is the basic schedule, and the Gantt chart will submit it before the next meeting.

|  |  |  |
| --- | --- | --- |
| Date | Task | Sub Task |
| Week 1  [before 2/10-2/16] | Background study on the online shopping mall implementation of similar projects.[2/10] |  |
|  | Collect detailed requirements.[2/10] |  |
|  | Design the overall architecture.[2/11] |  |
|  | Design database. [2/12-2/14] |  |
|  | Design interfaces for web browsers. [2/15-2/16] |  |
| Week2  [2/17-2/23] | Develop interface templates for the supervisor to review[2/7] |  |
|  | Create a site map or hierarchy chart showing the flow of web pages.[2/19] |  |
|  | Create the individual web pages for the site.[2/20] |  |
|  | Create a database for the server-site modules. [2/21] |  |
| Week3  [2/24-3/2] | Sprint for [T1]Display product list\* | A1(Browse products) [2/24] |
|  |  | A2(Paging)[2/24] |
|  |  | A3(Filter)[2/24] |
|  |  | A4(Searching)[2/26] |
|  |  | A5(Sort by price)[2/26] |
|  |  | A6(Product detail page)[2/27] |
|  |  | A7(One or more photographs)[2/27] |
|  |  | C1(Adds a product)[2/28] |
|  |  | C6(Duplicate product)[2/28] |
|  |  | B4(Redirects)[3/1] |
|  |  | Flex time[3/2] |
| Week4  [3/3-3/9] | Sprint for [T2]Login/register function\* | B1(register a new account)[3/3] |
|  |  | B2(login and log out)[3/4] |
|  |  | B3(change password)[3/5] |
|  |  | B5(hash values)[3/5] |
|  |  | Flex time[3/5-3/9] |
| Week5  [3/10-3/16] | Sprint for [T3]Product Page\* and Sprint for [T6]Add/delict product features\* | E1(browse the product catalogue)[3/10] |
|  |  | E2(Vendor searching)[3/10] |
|  |  | E3(Add a new product)[3/12] |
|  |  | E4(hash values)[3/12] |
|  |  | E5(Upload photographs)[3/13] |
|  |  | E6(add or remove photos)[3/13] |
|  |  | Z5(price change)[3/14] |
|  |  | Flex time[3/15-3/16] |
| Week6  [3/17-3/23] | Sprint for [T4]Purchase Tracking\* | D1(Purchase tracking page)[3/17] |
|  |  | D2(Vendor filter)[3/18] |
|  |  | D3(Purchase order detail page)[3/19] |
|  |  | D4(Cancel the order)[3/20] |
|  |  | Z3(Customer’s rating)[3/21] |
|  |  | Z4(reviews for products)[3/22] |
|  |  | Flex time[3/23] |
| Week7  [3/24-3/30] | Sprint for [T5]\* | F1(purchase order list page)[3/24] |
|  |  | F2(Status)[3/24] |
|  |  | F3(purchase order processing page)[3/25] |
|  |  | F4(Purchase Order)[3/26] |
|  |  | F5(hold a purchase order)[3/27] |
|  |  | F6(hold and ship a purchase order)[3/27] |
|  |  | F7(cancel purchase order)[3/28] |
|  |  | Z6(out-of-stock)[3/28] |
|  |  | Flex time[3/29-3/30] |
| Week8  [3/31-4/6] | Sprint for [T7]\* | C2(List the products)[3/31] |
|  |  | C3(Check out)[3/31] |
|  |  | C4(Change the quantity)[4/2] |
|  |  | C5(Remove an item)[4/2] |
| Week9  [4/7-4/13] | Sprint for [T8]\* | Z1(best selling products)[4/7] |
|  |  | Z2(notification)[4/7] |
|  |  | Z7(concurrency control to )[4/9] |
| Week10 [4/14-4/20] | Writing report[4/14-4/20] |  |
|  | Debuggin[4/14-4/20] |  |
| Week11 [4/21-4/27] | Presentation preparation[4/21-/27] |  |
| Week12 [4/28-5/4] | Flex time[4/28-5/4] |  |
| Week13 [5/4-5/7] | Final report (hard copy), Presentation slides (hard copy) |  |

# Job allocation

Job allocation is expressed by the RACI matrix, A RACI chart is a simple matrix used to assign roles and responsibilities for each task, [milestone](https://www.teamgantt.com/blog/the-how-and-why-of-using-milestones-in-your-project-plan), or decision on a project. In this, RACI will express the mission role.

R = Responsible

A = Accountable

C = Consulted

I = Informed

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Task | Sub Task | Ben | Chi | Tony | Responsible |
| Background study on the online shopping mall implementation of similar projects. |  | A R | I | I | Ben |
| Collect detailed requirements. |  | C | A,R | I | Chi |
| Design the overall architecture. |  | A,C | I | R | Tony |
| Design database. |  | A,R | I | I | Ben |
| Design interfaces for web browsers. |  | R | C | C | Ben |
| Develop interface templates for the supervisor to review |  | R | A | C | Ben |
| Create a site map or hierarchy chart showing the flow of web pages. |  | C | R | A | Chi |
| Create the individual web pages for the site. |  | C,R | C | I | Ben |
| Create database for the server-site modules. |  | C,R | C | I | Ben |
| Sprint for [T1]Display product list |  | C,A | I | R | Tony |
|  | A1(Browse products) |  |  |  |  |
|  | A2(Paging) |  |  |  |  |
|  | A3(Filter) |  |  |  |  |
|  | A4(Searching) |  |  |  |  |
|  | A5(Sort by price) |  |  |  |  |
|  | A6(Product detail page) |  |  |  |  |
|  | A7(One or more photographs) |  |  |  |  |
|  | C1(Adds a product) |  |  |  |  |
|  | C6(Duplicate product) |  |  |  |  |
|  | B4(Redirects) |  |  |  |  |
|  | Flextime |  |  |  |  |
| Sprint for [T2]Login/register function\* |  | A,R | I | I | Ben |
|  | B1(register a new account) |  |  |  |  |
|  | B2(login and log out) |  |  |  |  |
|  | B3(change password) |  |  |  |  |
|  | B5(hash values) |  |  |  |  |
|  | Flextime |  |  |  |  |
| Sprint for [T3]Product Page\* and Sprint for [T6]Add/delict product features\* |  | A,C | I | R | Tony |
|  | E1(browse the product catalogue) |  |  |  |  |
|  | E2(Vendor searching) |  |  |  |  |
|  | E3(Add a new product) |  |  |  |  |
|  | E4(hash values) |  |  |  |  |
|  | E5(Upload photographs) |  |  |  |  |
|  | E6(add or remove photos) |  |  |  |  |
|  | Z5(price change) |  |  |  |  |
|  | Flextime |  |  |  |  |
| Sprint for [T4]Purchase Tracking\* |  | C | I | A,R | Tony |
|  | D1(Purchase tracking page) |  |  |  |  |
|  | D2(Vendor filter) |  |  |  |  |
|  | D3(Purchase order detail page) |  |  |  |  |
|  | D4(Cancel the order) |  |  |  |  |
|  | Z3(Customer’s rating) |  |  |  |  |
|  | Z4(reviews for products) |  |  |  |  |
|  | Flextime |  |  |  |  |
| Sprint for [T5]\* |  | A,R | R | I | Ben and Chi |
|  | F1(purchase order list page) |  |  |  |  |
|  | F2(Status) |  |  |  |  |
|  | F3(purchase order processing page) |  |  |  |  |
|  | F4(Purchase Order) |  |  |  |  |
|  | F5(hold a purchase order) |  |  |  |  |
|  | F6(hold and ship a purchase order) |  |  |  |  |
|  | F7(cancel purchase order) |  |  |  |  |
|  | Z6(out-of-stock) |  |  |  |  |
|  | Flextime | A,C | R | R | Chi and Tony |
| Sprint for [T7]\* |  | C | A,R | R | Chi and Tony |
|  | C2(List the products) |  |  |  |  |
|  | C3(Check out) |  |  |  |  |
|  | C4(Change the quantity) |  |  |  |  |
|  | C5(Remove an item) |  |  |  |  |
| Sprint for [T8]\* |  | R | A,R | I | Ben |
|  | Z1(best selling products) |  |  |  |  |
|  | Z2(notification) |  |  |  |  |
|  | Z7(concurrency control to ) |  |  |  |  |
| Writing report[4/14-4/20] |  | A,C | R | R | Chi and Tony |
| Debuggin[4/14-4/20] |  | A,R | I | R | Ben and Tony |
| Presentation preparation[4/21-/27] |  | A,R | R | R | Chi and Tony |
| Flex time[4/28-5/4] |  |  |  |  |  |
| Final report (hard copy), Presentation slides (hard copy) |  | A,R | R | R | Chi and Tony |

# Appendix B. Peer Assessment Form

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| S:\3rd ITC\2nd ITC\MPI_logos\MPI logo09_C349 CPE.tif  BSc. in Computing 2020/21 COMP321 Information System Implementation  Peer Assessment Form | | | | | | |
| Group number | A5 | | | | | |
| Group members | |  |  |  | | --- | --- | --- | |  | Student ID | Student name | | 1. | P-18-0187-5 | Ben Leong | | 2. | P-16-0498-1 | Chi Deng | | 3. | P-18-0191-5 | Tony Tou | | 4. |  |  | | 5. |  |  | | | | | | |
| Contribution **(**Each row must total 100%) | | | | | | |
|  | | Member 1 | Member 2 | Member 3 | Member 4 | Member 5 |
| 1. Project leadership | | 50% | 20% | 30% | % | % |
| 2. Data modeling | | 40% | 35% | 25% | % | % |
| 3. User interface design | | 50% | 20% | 30% | % | % |
| 4. Program development | | 30% | 50% | 20% | % | % |
| 5. Solving technical problems | | 50% | 30% | 20% | % | % |
| 6. Testing and sample data | | 30% | 30% | 40% | % | % |
| 7. Report writing | | 35% | 35% | 30% | % | % |
| 8. Preparing / giving presentation | | 50% | 25% | 25% | % | % |

# Appendix C. Program source code

All source code is already uploaded at this link.

https://drive.google.com/drive/folders/1rCII8vAgb7IjD5GHy\_IqkCzZFQ02qmta?usp=sharing

ACKNOWLEDGEMENTS

We would like to thank our supervisor Calana Chan for her guidance through each stage of the process.

We would like to thank ourselves and our team members for cooperating throughout the process. Whenever we have difficulties, we will encourage and support each other, and we also appreciate our persistence.

This project includes many courses such as database design, project management, and programming, etc. We thank the teachers who have taught us this knowledge.