澳 門 理 工 學 院
Instituto Politécnico de Macau
Macao Polytechnic Institute

School of Public Administration
Bachelor of Science in Computing

COMP321 Information System Implementation
Final Report

Academic Year 2017/18
2nd semester

Online Shopping Mall

Project number:	1
Team members:	Demi(P1507862)
	Tony(P1507921)
	Chris(P1507881)
	Giulio(P1507805)
Supervisor:	Calana
Assessor:	Andrew Siu
Submission Date:	Apr 23, 2018

Table of Contents

1 Introduction

1.1 Overview

The developed network environment enables retail businesses to solve some traditional problems. Online shopping malls are websites that enable retailers to sell in one portal and allow customers to purchase. Customers may do shopping at home and no longer need go outside. Moreover, customers may view over a wide range of products without geographic restrictions. Vendors can save huge expense by not running a brick-and-mortar store but pay a cheaper cost for running the online shop. This report is about our project, Tonymall, which is an online shopping mall selling various kinds of smart phones.

1.2 Objectives

The main objective of the project is to provide an online smart phone shop, including the following functions.

1. Customers may search products by keyword or brand name.
2. Registered customers can add products into shopping cart after login to Tonymall.
3. Registered customers can check their carts and place orders.
4. After bought a product, customers can write reviews and rate for the product.
5. The vendor can manage customers' orders which customers can trace.
6. The vendor may change products' price, which will not affect the price in existing purchase orders.
7. The vendor can analyze the sales of products and find out the best selling product.

This report is organized as follows: Chapter 2 introduces the background of our work. Chapter 3 presents our design approach. Chapter 4 explains the implementation details. Chapter 5 shows our results and discussion. Finally, chapter 6 makes conclusion and further work about this project.

2 Background and Related Work

2.1 Background

We are living in an information age, electronic devices are becoming more and more prevalent. With more people addicted to using computers and mobile phones, e-commerce has developed its own market. E-commerce allows customers to purchase many kinds of commodities online with no barriers of time or distance. At the same time, it provides a platform for retailers to sell goods on the internet easily. E-commerce has expanded rapidly over the past several years and is predicted to continue developing with a high speed. While, with the development of e-commerce, there are some challenges for it and people are trying to solve them. First of all, the trust problem. Nowadays almost all e-commerce platforms support customers to communicate with sellers online to minimize the trust problem. Besides, customers have the right to review products that they have purchased successfully and all reviews are transparent so that other customers are able to understand the corresponding products and sellers more according to those reviews. Secondly, order cancellation problem. Customers may be worried about that they purchase something hastily but then they find that they do not need those products. E-commerce platforms allow customers to cancel an order that they have verified. This operation provides customers a sense of safety so that they don't need to worry about regret shopping.

E-commerce has several common features. Firstly, it must be easy to use. The e-commerce website has to provide simple and unified navigation bar for customers in order that users have the ability of operating the website immediately. Compared to a complex website which customers need to spend lots of time to learn how to use, a simple website must be the best choice for customers because the website could bring lots of convenience to them. Secondly, quick loading times for every page. Customers may lose patience easily when they face to a website that they need to wait for a long time to change pages within. Quick loading times could bring customers better using experiences and thus they will probably enjoy using the website. Thirdly, high-resolution photos. E-commerce websites should display multiple photos per product. Those photos must be high-resolution and display lots of angles of products so that customers could know products more deeply. Fourthly, user-generated reviews. The star rating on a product is really popular among customers. Besides, e-commerce websites allow customers to write some reviews about products so that customers are able to evaluate products objectively. Our e-commerce website offers clear and concise navigation bar and high-

resolution photos of each product to customers. Besides, our website provides customers a review platform so that they could give reviews to products that they have purchased.

2.2 Related Work

We design our e-commerce platform according to Taobao. Firstly, customers can only add products into the shopping cart when he or she has logged in successfully. Just like Taobao, if the customer hasn't logged in, the website displays the login page to customers automatically to allow them to login. Secondly, customers can see several pictures of one product. Taobao provides products' images in lots of angles and we also offer both thumbnail and detail images per product to customers.

However, there are some differences between our website and existing e-commerce platforms. The first difference is that we display all products in the product page in a table and each product takes up one row space. We only have small number of products in each page and thus it is really convenient for customers to see different products in a table and compare the prices through the price column themselves. Besides, we only sell smart phones in our e-commerce website and thus we do not need to provide customers the function of searching by products type in the product page. We only allow customers search by product name or brand. The final point is that we provide RAM and processor information in the product detail page because these two characteristics are really crucial for a smart phone and we think customers will pay more attention to these two fields when they want to purchase a smart phone.

3 System Design

In systems design, our process is defining the architecture, modules, interfaces, and data for Tonymall to satisfy the requirements.

3.1 Data Modelling

Basically, we have 6 entities in our database design. “SmartPhone” is the most important entity, which stores our products. “User” stores the user details. Every transaction is based on these two entities.

Figure 1 shows the entity relationship diagram of the database of this project.

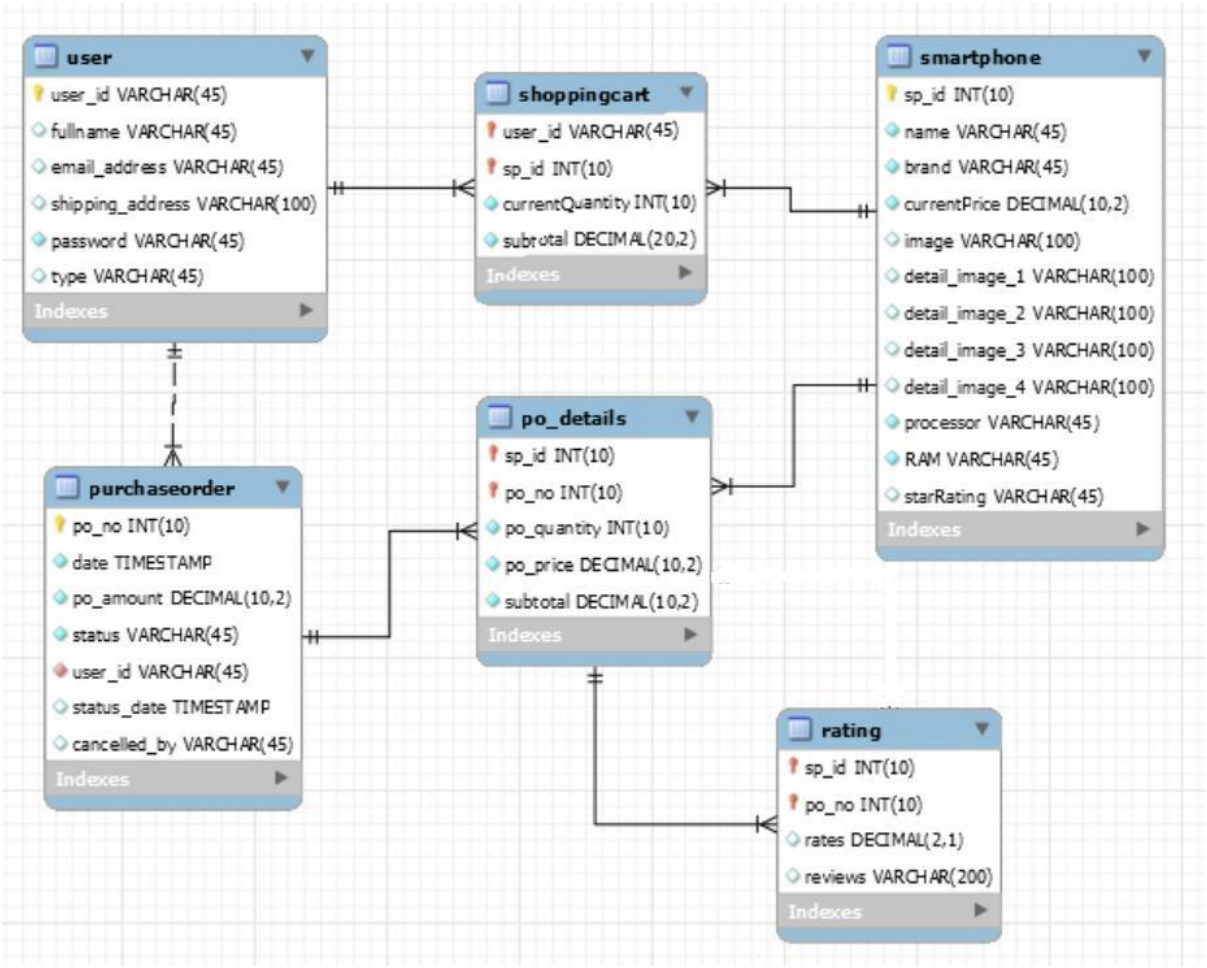


Figure 1: Entity Relationship Diagram

"User" entity consist of a vendor and lots of customers. Each user is uniquely identified by user_id, which is generated by the system. There is an attribute named "type", which is for distinguishing the vendor's account and customers' account. Vendor is able to change the information about products while customers cannot.

In "SmartPhone" entity, "sp_id" is the primary key of each smartphone. "image" and "detailed_image" are stored as a path in the database. The website displays the images by the path of the image. "processor" and "RAM" are the two properties of a smartphone.

In shopping cart, "sp_id" and "user_id" are the composite primary key, and they are foreign key which reference to sp_id of the SmartPhone table and user_id of the User table respectively. The subtotal is the total amount of each product. This is calculated by the price of the single product multiplying the quantity of the product. The total price is the sum of all subtotals. However, the total price is not stored in our database, it's calculated in real-time in JavaScript, and then shown in the website immediately. If the "currentPrice" in "SmartPhone" changes, the price shown in shopping cart will be changed at the same time because customer haven't purchased it.

In "PurchaseOrder" table, each purchase order has a unique "po_no", and has a foreign key "user_id" which references to "user_id" of the User table. "po_amount" is the total of the order, and is stored rather than derived. "date" is created once user confirms the purchase order. "status" describes the various stages of the purchase order processing, possible values include "pending", "shipped", "hold" and "canceled".

In "PO_details", there is an attribute called "po_price" which is the price of that product when transaction occurred. The "po_price" will not be changed any more once written down, even when the "currentPrice" in "SmartPhone" changes.

In rating system, a user can only make a comment and rate a product within an order only once. In "smartphone" table, "starRating" will record average rate for each product. Every time customer reviews the shipped product, "starRating" will be updated.

3.2 Dynamic Modelling

In Tonymall, customers are able to browse products without login. But they have to login first if they want to buy something, check shopping cart, or track the purchase order.

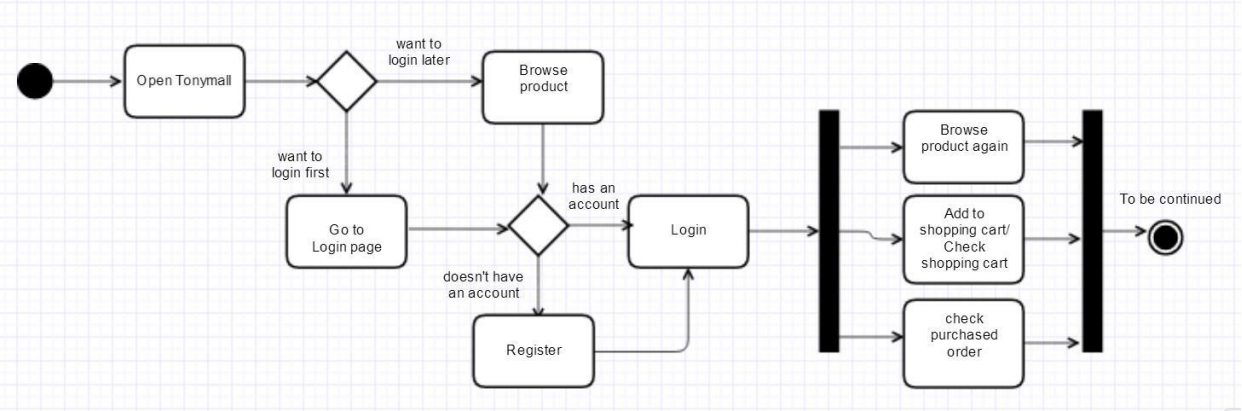


Figure 2: Activity Diagram for user account system

If a user can't find anything he wants, he may just leave Tonymall directly. But if he finds a product, he can add this product to shopping cart first, then decides whether to buy it right now or not. If he purchases it immediately, he just needs to wait until Tonymall ships the product for him. However, he has another option which is not to buy it right now, and leaves it in shopping cart.

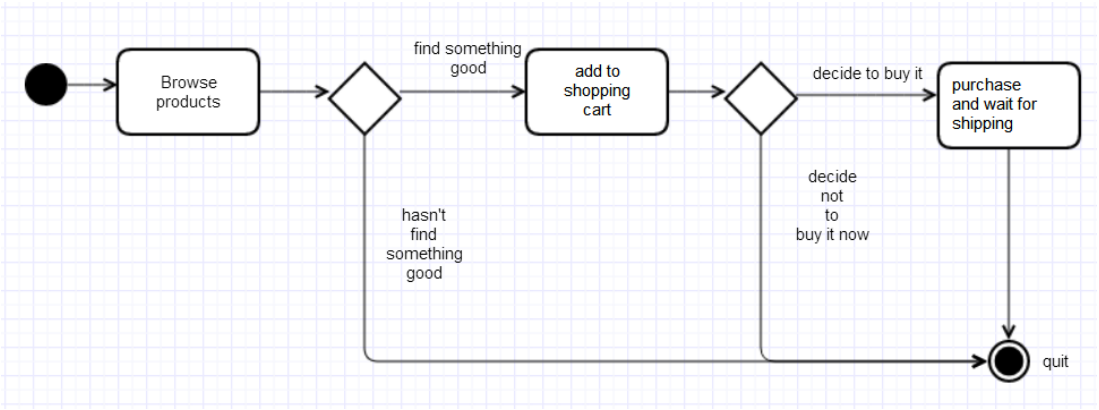


Figure 3: Activity Diagram for purchasing product

After the user receives the product, he is able to make a rating and write down some comment for that product. Also, if he doesn't want to review or he already reviewed once, he may just quit.

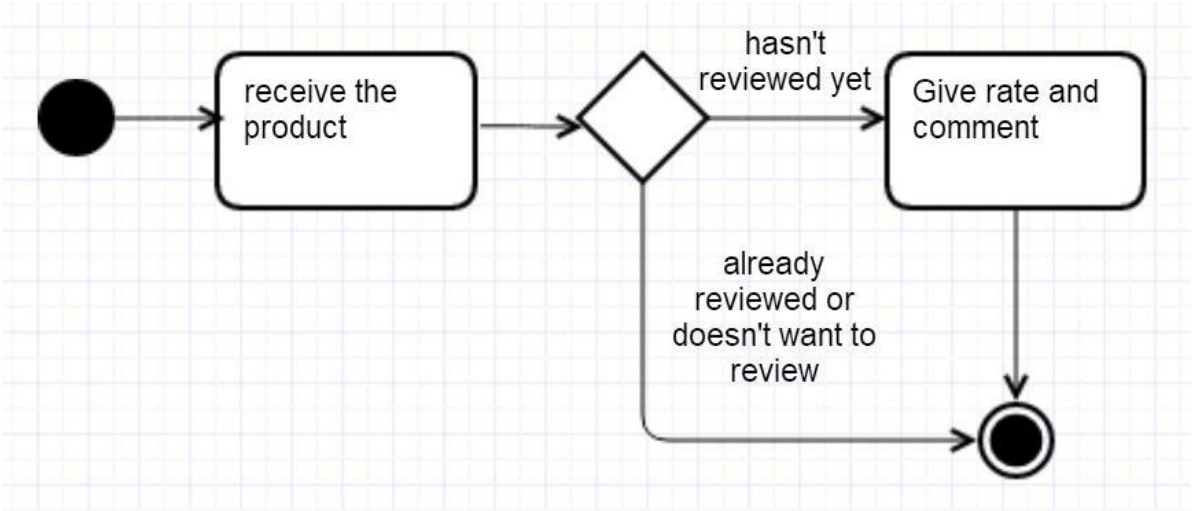
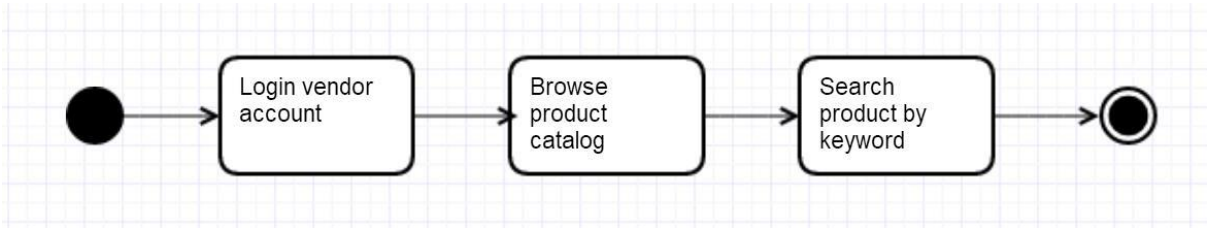


Figure 4: Activity Diagram for review system

Vendor is able to login his unique vendor account. He can also browse the product catalog as customers can, and search product by keyword even the product ID.



4 System Implementation

The following paragraphs explain how we implement our system. Section 4.1 introduces the main architecture of system. From section 4.2 to 4.7, we will talk about each component in more details.

4.1 Architecture

TonyMall online shopping website is implemented based on Model-view-controller (MVC) architectural pattern. MySQL database represents the model to manage all data of TonyMall. 'views' folder includes all the representation of webpage and app.js & 'routes' folder plays the role of controller.

We are taking advantages of Node.js to develop server-side of TonyMall because of the following reasons:

- Node.js is based on V8 engine which is high-performance JavaScript engine developed by Google.
- Built-in package management. NPM tool in Node.js provides convenient way for installation.
- Increasing scale of community. It is much easier to solve problems and debug.

Moreover, in order to store the data of users, vendors, products and purchase orders, we rely on MySQL. The convictive reasons are following:

- large size of community. It can help us to save developmental time.
- Easy to get started. We have learned relative courses of MySQL.
- Provide MySQL Workbench, which is efficient to use because of visual database design tool.

The whole TonyMall website is running JavaScript language both on browser-side and server-side. For the purpose of reducing development time and providing greater web design, TonyMall is using handy and practical Web UI framework, the Bootstrap, serving from tiny button to form table. Below are several reasons to choose this framework:

- Easy to use. One of the advantages of Bootstrap is that it only needs to add class to elements where you want to implement.
- Provide great grid system. Layouts are the one of the most important parts in website, messy format can degrade the user experience. Bootstrap is based on responsive grids which have much better layout.
- Documentation is excellent. We can implement components more effectively.

Font Awesome is also used in TonyMall to provide icons.

In the following sections, we will elaborate how we implement and develop TonyMall in more details.

4.2 Product Search and Detail Display

Product can be searched, or filtered, by name, brand or id. The search result can be sorted by price.

```
                con.query('SELECT sp_id, name, brand, currentPrice, image,
detail_image_1, ' +
                    'detail_image_2, processor, RAM, starRating, starRating*20
as starRatingInPercent' +
                    ' FROM smartphone WHERE ?? LIKE ? LIMIT ?, ?;', [search, '%'
+ keyword + '%',
                    (current_page - 1) * itemNo, (current_page - 1) * itemNo +
itemNo], function (err, row) {
```

4.3 Image Storage and Handling

Product images are stored in local folder. These images' paths are stored in the database. These images can not be uploaded, changed or deleted. Related images will be displayed in the product detail page.

4.4 Add Product to Shopping Cart

After users logs in, product can be added to shopping cart if user click the button "Add to shopping cart".

```
                    <button type="button" onclick="location
.href='/products/AddToCart/{{this.sp_id}}'" id="detail-page-addToShoppingCart"
class="btn btn-primary btn-lg btn-block">Add  to shopping cart</button>
```

```
        con.query("INSERT INTO shoppingcart(user_id, sp_id, currentQuantity,
subtotal) Values(?, ?, 1, (SELECT currentPrice from smartphone where sp_id=?))",
[user, spId, spId], function (err, result) {
```

In the shopping cart page, all products' information is displayed in table format.

```
            <table class="table">
                <thead class="thead-light">
                <tr>
                    <th>Purchase Name</th>
                    <th>Price</th>
                    <th>Quantity</th>
                    <th>Remove</th>
                </tr>
                </thead>
                <tbody>
                {{# if CartProduct}}
                    {{# each CartProduct}}
                        <tr>
                            <th scope="row"  onclick="location.href='/products/detail/{{this.sp_id}}'">{{this.name}}</th>
                            <td>{{this.currentPrice}}</td>
                            <td><button id="Subtract" type="button" class="btn btn-primary" onclick="location
.href='/products/Subtract/{{this.sp_id}}'">-</button>
                                <span id="quantity">{{currentQuantity}}</span>
                                <button id="Add" type="button" class="btn btn-primary" onclick="location.href='/products/Add/{{this
.sp_id}}'">+</button>
                            </td>
                            <td><button id="Remove" type="button" class="btn btn-primary" onclick="location
.href='/products/remove/{{this.sp_id}}'">Remove</button></td>
                        </tr>
                    {{/each}}
                {{/if}}
                </tbody>
            </table>
```

4.5 Checking Out All Items

In the shopping cart page, users are able to check out all products in the shopping cart by clicking the "Check Out All Products" button. This action creates a purchase order with a unique P.O number and clears the content of the cart.

```
        var queries = "INSERT INTO purchaseorder(date, po_amount, status, user_id)
Values(NOW(), (SELECT SUM(subtotal) FROM shoppingcart WHERE user_id=?), ?, ?);
SELECT LAST_INSERT_ID(); INSERT INTO po_details (sp_id, po_no, po_quantity,
po_price, subtotal) SELECT sp_id, LAST_INSERT_ID(), currentQuantity, subtotal DIV
currentQuantity, subtotal FROM shoppingcart WHERE user_id=?; DELETE FROM shoppingcart
 WHERE user_id =?;";
        con.query(queries, [user, 'pending', user, user, user], function (err,
result) {
```

4.6 Purchase Order Processing

The core part of shopping website is purchase order processing. Some changes of state can only be done in certain situations. Customer type will be checked when user logs in to TonyMall. More specifically, the following code examines how we implement purchase order processing.

Firstly, we separate customer and vendor in different routes. Therefore, there is no conflict between customer and vendor.

```
// Vendor
app.use('localhost/vendor/order/past', vendorRoutes);
app.use('localhost/vendor/order/hold/:num', vendorRoutes);
app.use('localhost/vendor/order/cancel/:num', vendorRoutes);
app.use('localhost/vendor/order/hold', vendorRoutes);
app.use('localhost/vendor/order/pending', vendorRoutes);
app.use('localhost/vendor/order/ship/:num', vendorRoutes);
app.use('localhost/vendor/order/:num', vendorRoutes);
app.use('/vendor/order', vendorRoutes);

//order
app.use('localhost/order/current', orderRoutes);
app.use('localhost/order/past', orderRoutes);
app.use('localhost/order/cancel/:num', orderRoutes);
app.use('localhost/order/:num', orderRoutes);
app.use('/order', orderRoutes);
```

Secondly, we create two handlebars helper functions when we setup view engine. ‘cancel’ helper function has three arguments, while ‘check_type’ helper function has two arguments.

```
app.engine('.hbs', expressHbs({
    defaultLayout: 'layout',
    extname: '.hbs',
    helpers: {
        cancel: function (arg1, arg2, arg3) {
            if (arg1 === arg2 || arg1 === arg3) {
              return true
            }else {
              return false
            }
      }, check_type: function (arg1, arg2) {
            if (arg1 === arg2) {
                return true
            }else {
                return false
            }
        }
    }
}));
```

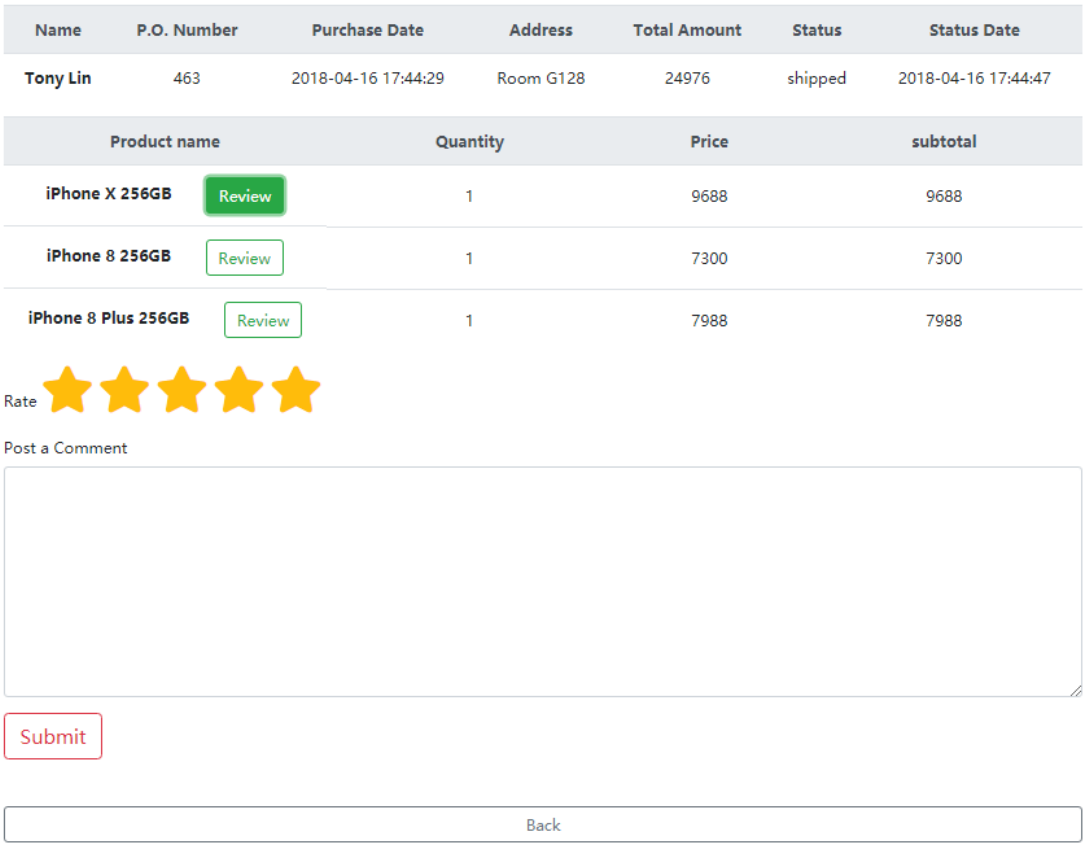
Thirdly, when we render the page, TonyMall uses these two functions to determine what kinds of buttons we should provide.

```
{{#if (check_type this.status 'pending')}}
```

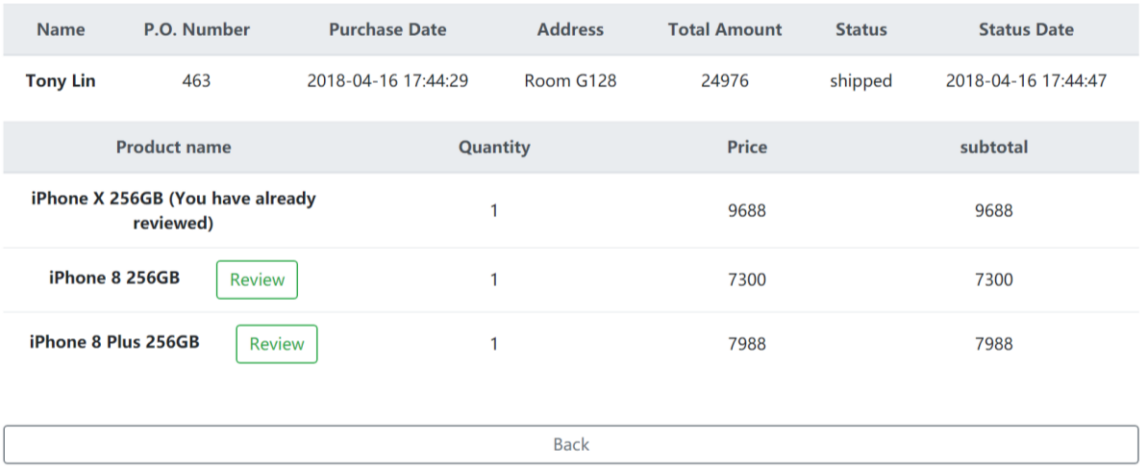
```
{{#if (cancel this.status 'pending' 'hold')}}
```

4.7 Customers' Ratings and Reviews

Rating and review function are provided in TonyMall. User can click review button to rate and post a comment to a product which user bought.



In order to reduce the burden on the server, changing their ratings is not allowed in TonyMall. Therefore, after reviewing a product, the button will be no longer be displayed.



In MySQL, we have a table called ‘rating’ to store all the reviews from user.

Table Name: rating

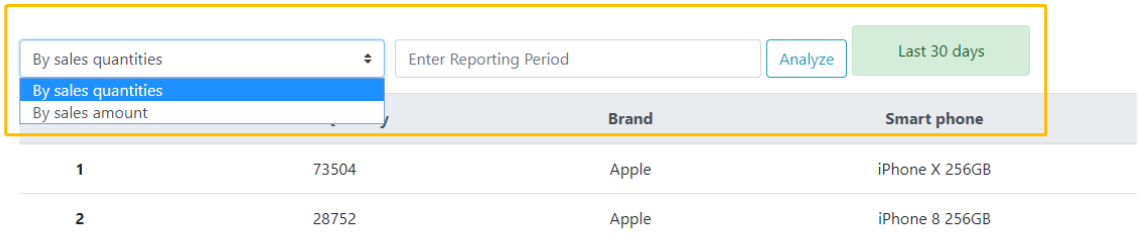
Collation: utf8 - default collation

Comments:

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
sp_id	INT(10)	☑	☑	☐	☐	☐	☐	☐	☐	
po_no	INT(10)	☑	☑	☐	☐	☐	☐	☐	☐	
rates	DECIMAL(2,1)	☐	☐	☐	☐	☐	☐	☐	☐	NULL
reviews	VARCHAR(200)	☐	☐	☐	☐	☐	☐	☐	☐	NULL
		☐	☐	☐	☐	☐	☐	☐	☐	

4.8 Vendor's Analysis Function

TonyMall provide vendor a very handy function, which is a report about best selling products with giving period and the default period is the last 30 days. As you can see, there is search bar for vendor searches by sales quantities or sales amount. There is also a green box to show the period.



In order to implement this function, we have three different queries. 'default', 'by sales quantities' and 'by sales amount' respectively.

This is the query for 'default' whose period is 30 days:

```
con.query('SELECT s.sp_id, s.brand, SUM(p.subtotal) AS total, s.name\n' +
    'FROM po_details p, smartphone s, purchaseorder o\n' +
    'WHERE p.sp_id = s.sp_id and p.po_no = o.po_no and ' +
    'datediff(NOW(), o.date) < 30 and o.status = \'shipped\'\n' +
    'group by s.sp_id\n' +
    'ORDER BY TOTAL desc;', (err, result) => {
```

This is the query for 'by sales quantities':

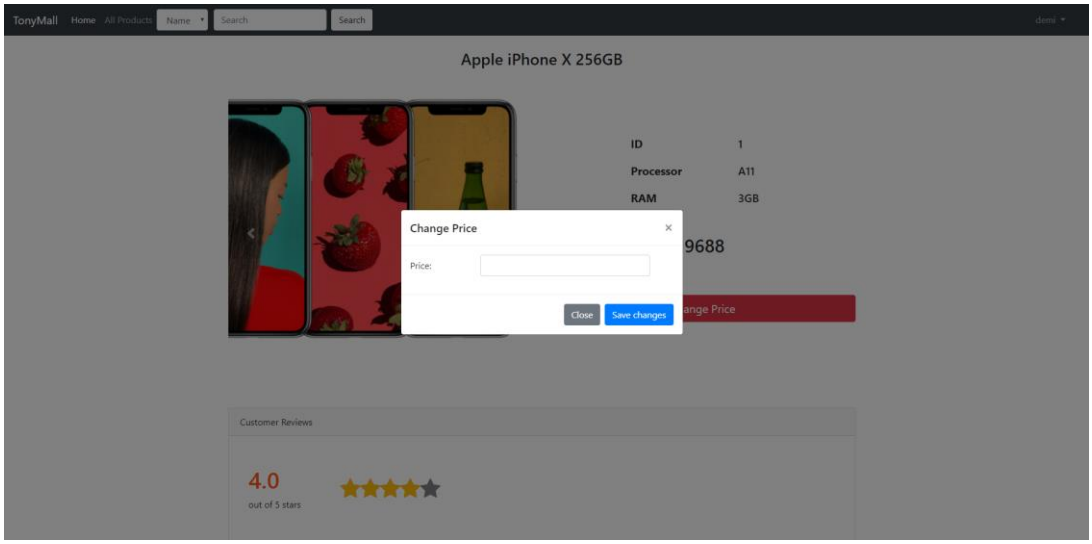
```
} else if (measure !== undefined && period !== undefined && measure === 'quantity') {
    con.query('SELECT s.sp_id, s.brand, SUM(p.po_quantity) AS total, s.name\n' +
        'FROM po_details p, smartphone s, purchaseorder o\n' +
        'WHERE p.sp_id = s.sp_id and p.po_no = o.po_no and ' +
        'datediff(NOW(), o.date) < ? and o.status = \'shipped\'\n' +
        'group by s.sp_id\n' +
        'ORDER BY TOTAL desc;', [period], (err, result) => {
```

This is the query for ‘by sales amount’:

```
if (measure !== undefined && period !== undefined && measure === 'amount') {
    con.query('SELECT s.sp_id, s.brand, SUM(p.subtotal) AS total, s.name\n' +
        'FROM po_details p, smartphone s, purchaseorder o\n' +
        'WHERE p.sp_id = s.sp_id and p.po_no = o.po_no and ' +
        'datediff(NOW(), o.date) < ? and o.status = \'shipped\'\n' +
        'group by s.sp_id\n' +
        'ORDER BY TOTAL desc;', [period], (err, result) => {
```

4.9 Vendor’s Change Price Function

Vendor can change the price of products by clicking ‘change price’ button.



This is done by this query with passing product id and price which vendor inputs:

```
    var id = req.query.id;
    var price = req.query.price;
    var con = db.connect((err) => {
        if (err) throw err;
    });
    con.query('UPDATE tonymall.smartphone\n' +
        'SET currentPrice = ?' +
        'WHERE sp_id=?;', [price, id], (err, result) => {
        if (err) throw err;
        con.end();
    });
    res.redirect(`/products/detail/${id}`);
```

5 Results and Discussion

In this chapter, we mainly talk about the result and some discussion about this project. Section 5.1 shows project outcome and section 5.2 discusses testing and system evaluations.

5.1 Project Outcome

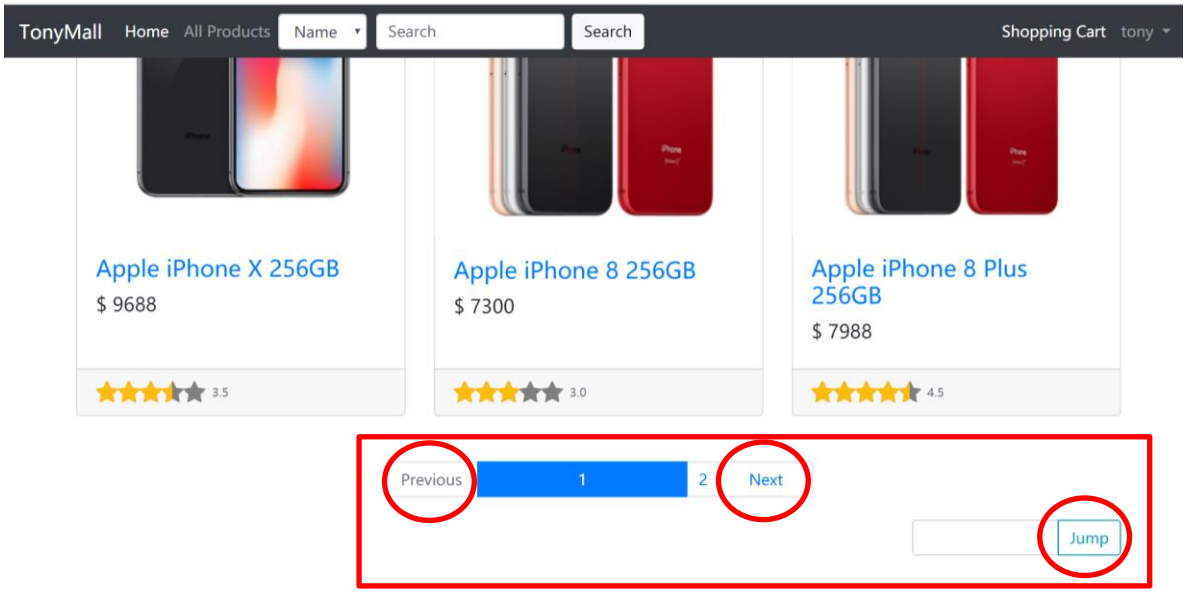
TonyMall has modern features of a shopping website. It not only provides great shopping experience to customers, but also supports practicable controls for vendor. The following sections discuss the outcome for customers and vendor.

5.1.1 Features for customers

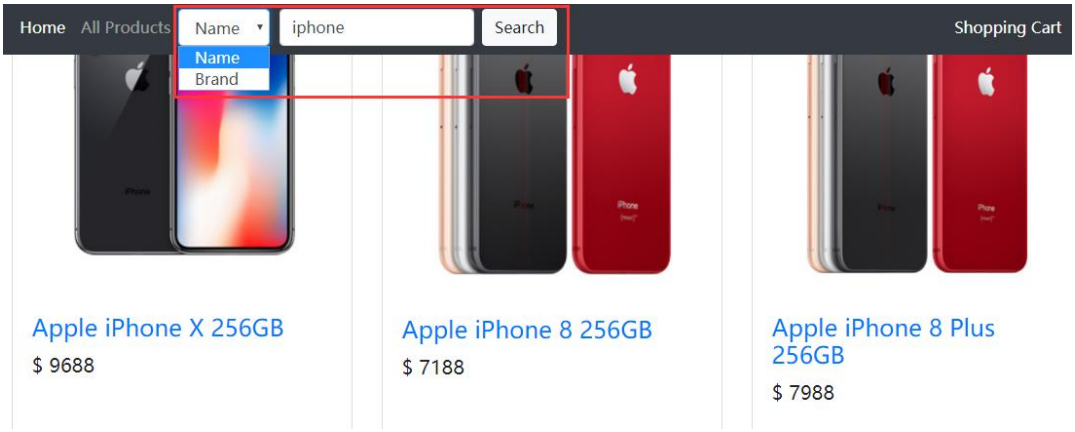
Various features are provided for customers in TonyMall.

- Product list

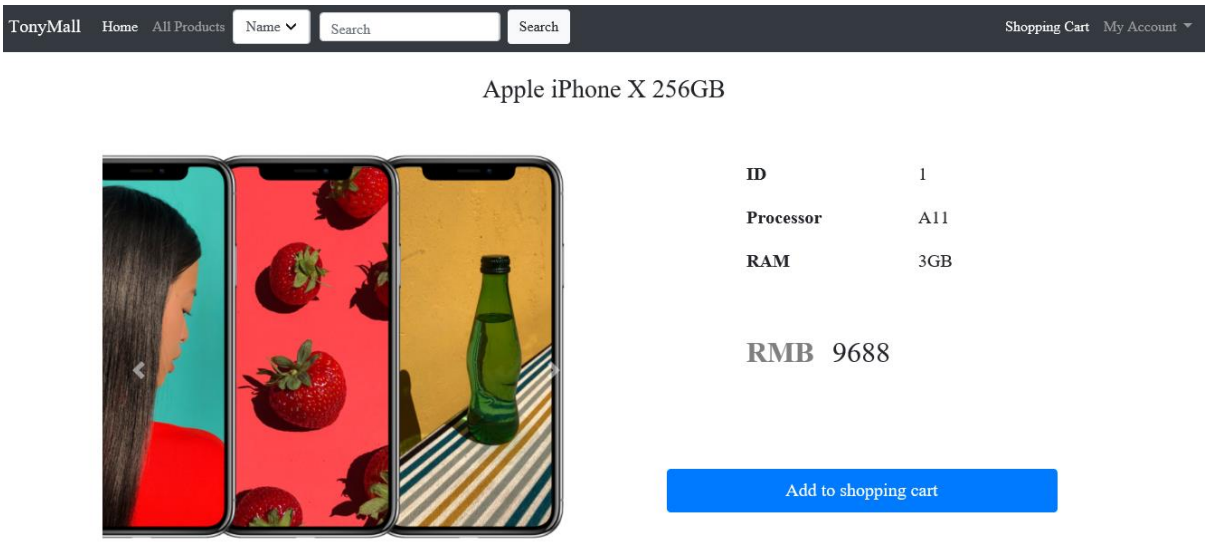
  A customer may click ‘All Products’ to browse products in the product list. The list shows basic information of products. The customer can navigate the product list by ‘page up’, ‘page down’ and jump to a specific page.



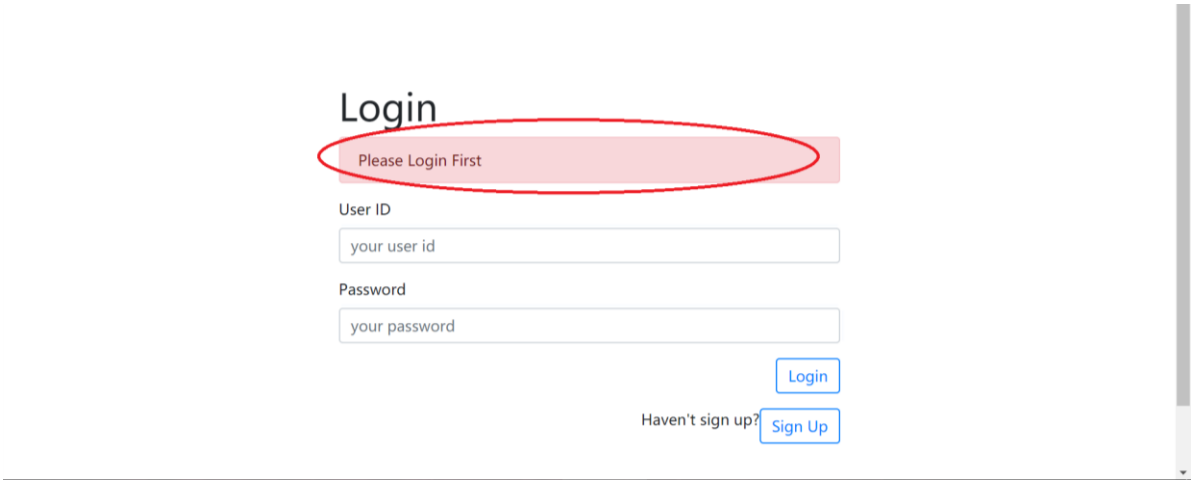
  The customer may filter the product list by keywords of product name or brand.



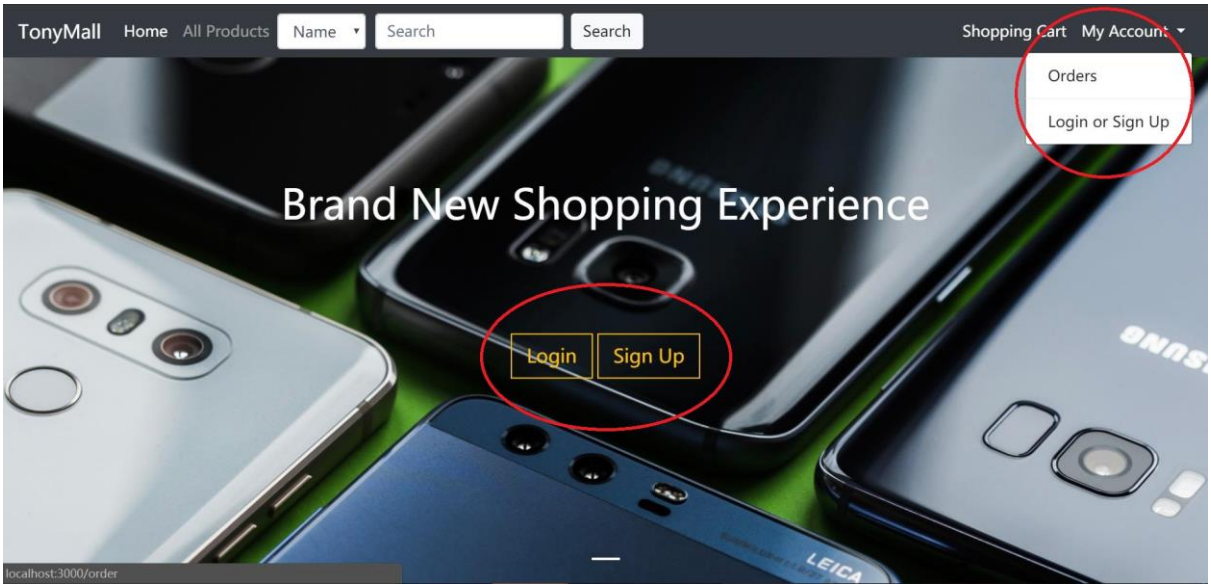
The customer can select a product to go to the product detail page which shows more information about that product.



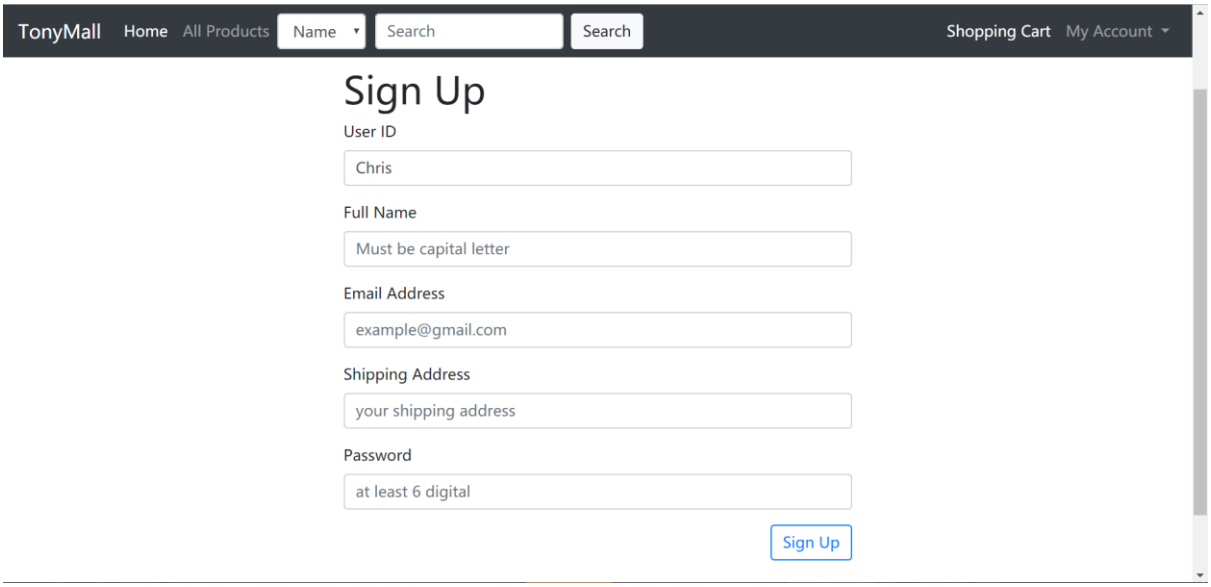
- Account management for customers.
  Customers can not add a product to shopping cart, check shopping cart and orders until he or she login to system. If a customer clicks the button “Add to shopping cart” without login, the system will automatically redirect the login page for customers, with a warning.



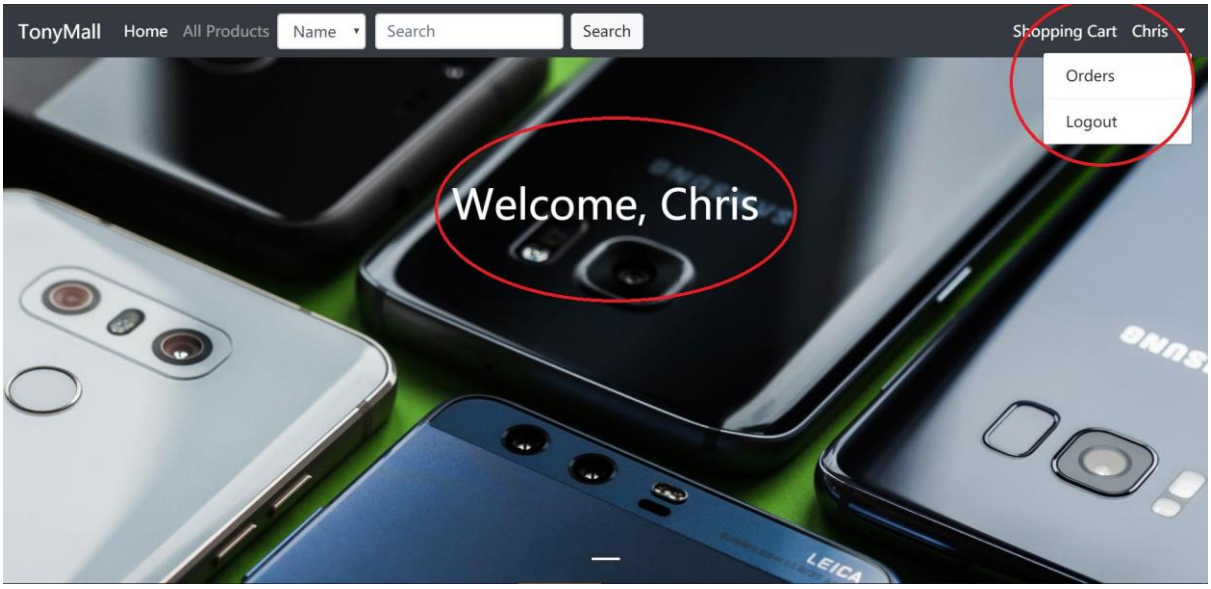
In the home page, there are “login” button and “sign up” button for customers, or by the menu button.



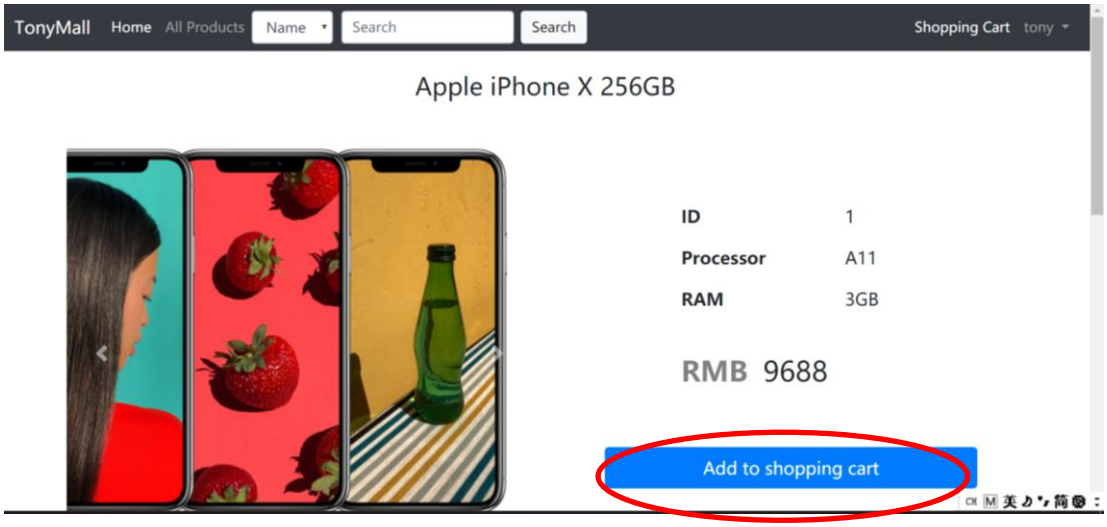
Besides, customers can sign up by filling the form. After signing up, the system will auto-login for the customer.



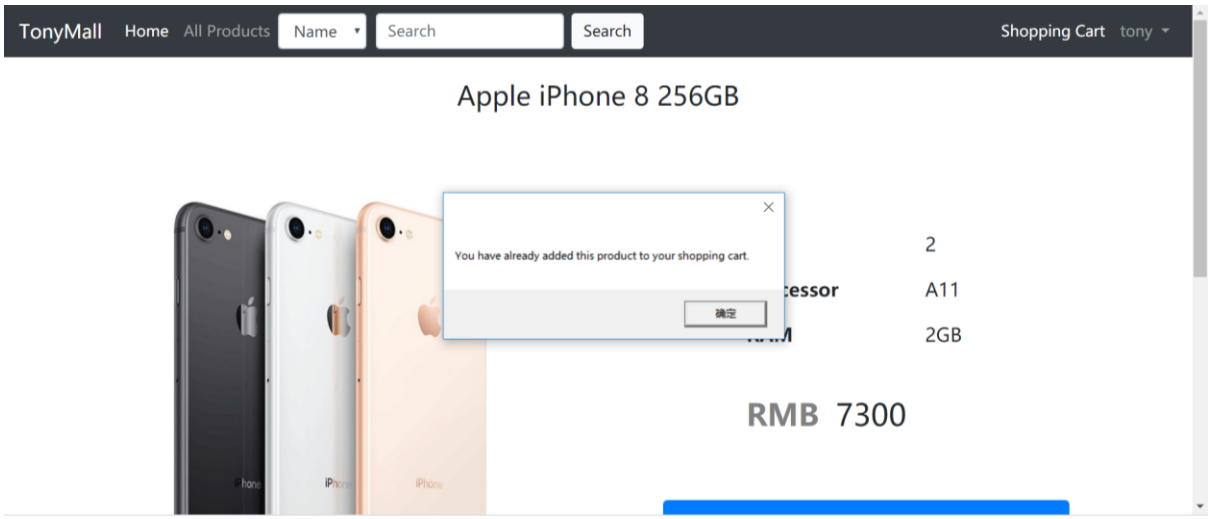
After login or sign up, the home page and the right-up menu button will show the username of customer. Also, the previous “login” button is changed to “logout”, and the logined customer can now manage orders.



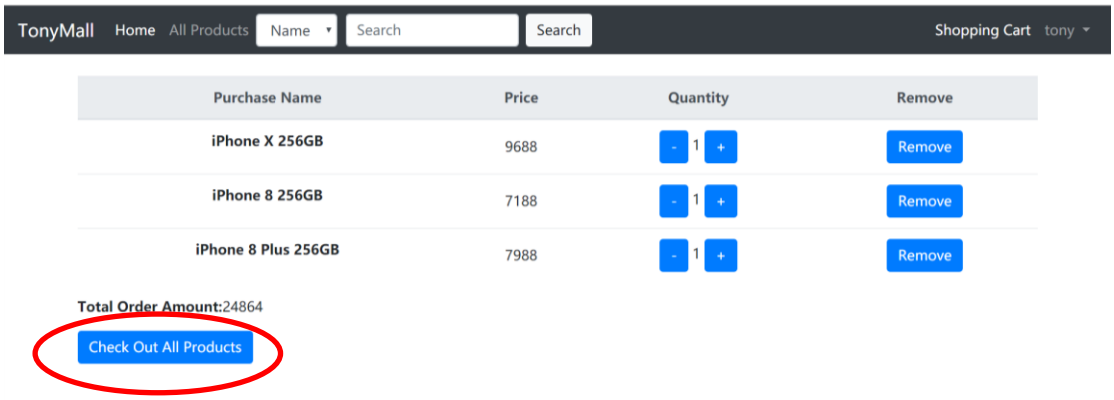
- Shopping cart for customers.
  After logging in to system, user can add a product to his/her shopping cart by clicking the button “Add to shopping cart”.



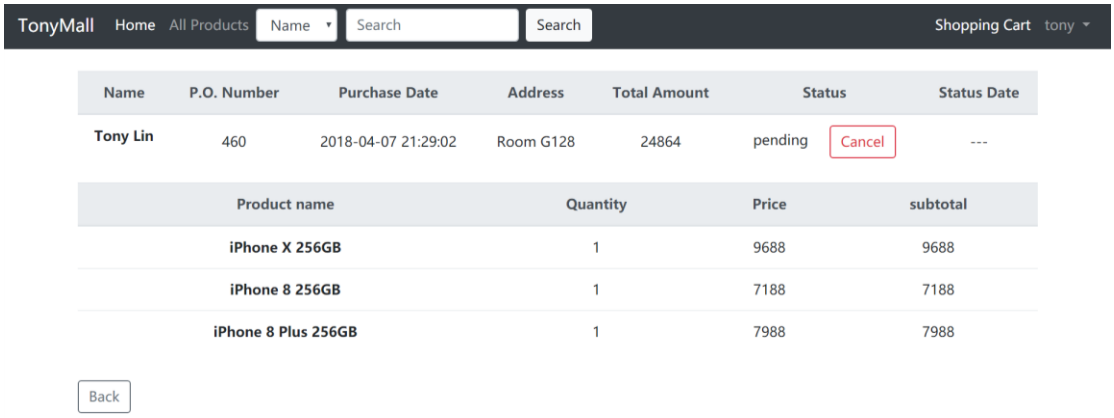
The customer will receive a warning message if he or she adds a duplicate product to the shopping cart.



The quantity to buy a product is assumed to be 1 and then customer can list the products in his/her shipping cart which shows the subtotal of each product and the total order amount. Customers can change the quantity or remove a product from the shopping cart. The customer can press the button “Check Out All Products” in the shopping cart page to check out all items in the shopping cart.

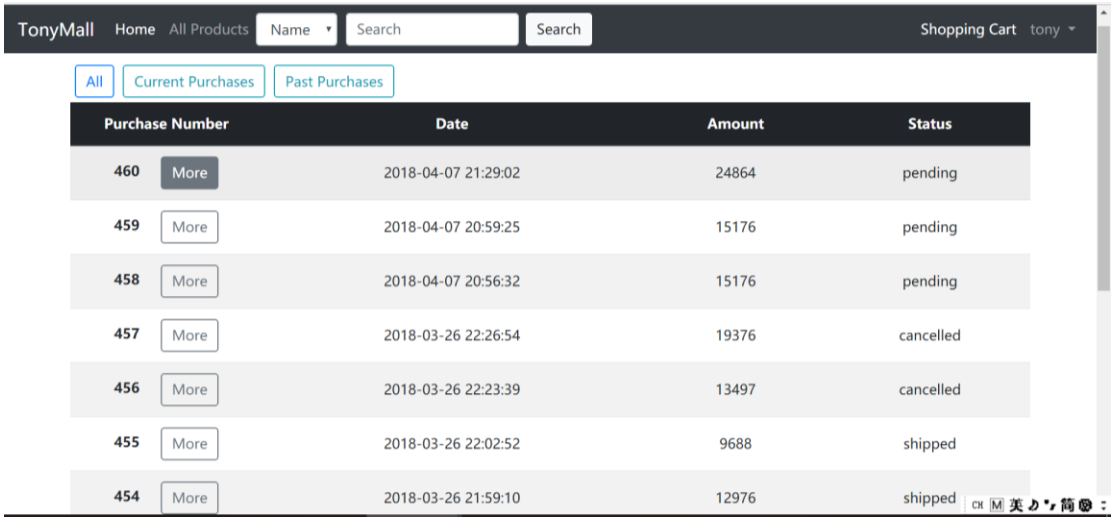


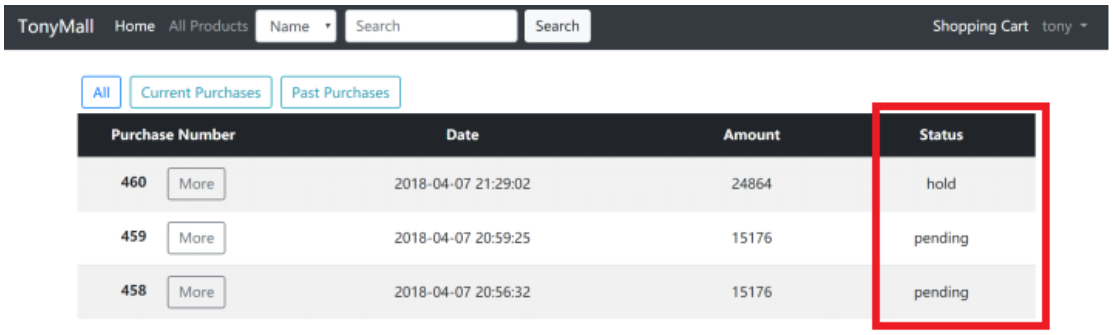
A purchase order will be created and the system shows the purchase order detail page of the newly created purchase order. The purchase order detail page shows the P.O. number, the purchase date, the customer name, the shipping address, the total order amount and the purchase order status.



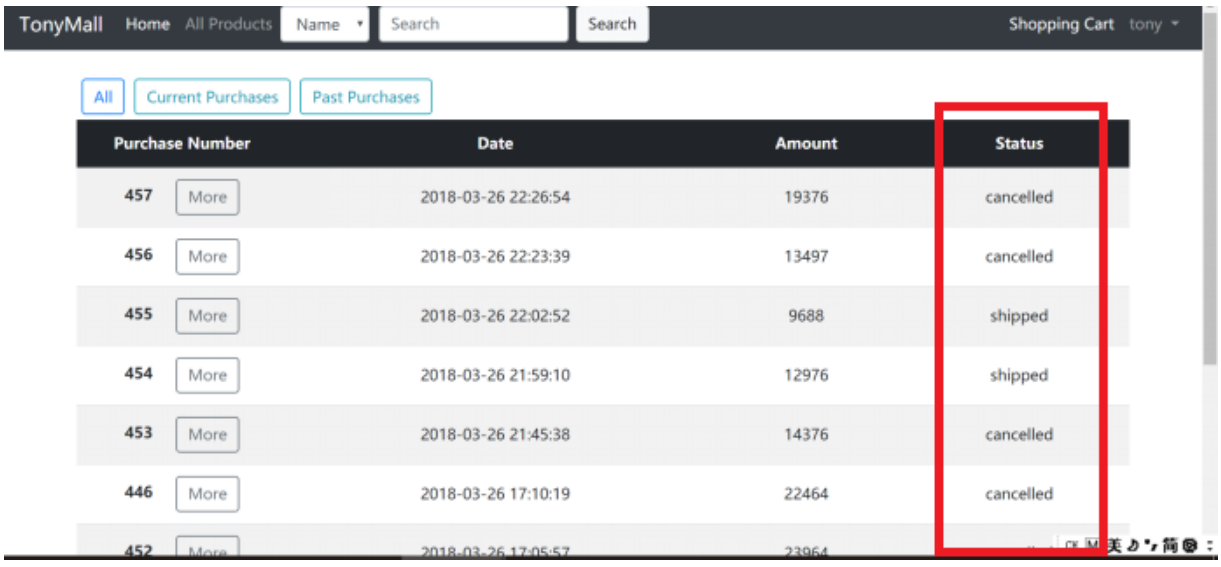
- Purchasing tracking for customers.

  After logging in to system, user can track their own purchase orders. The purchase tracking page lists the purchase orders that the customer has placed and the purchase orders are displaced in reverse chronological order of purchase date. The customer can filter the list of purchase orders in two ways: “current purchases” and “past purchases”.



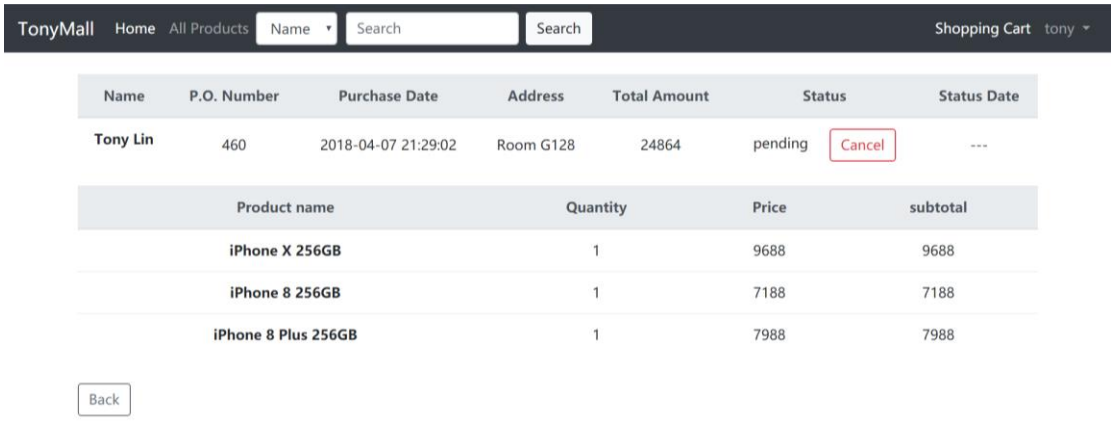


Result of “current purchase” filter

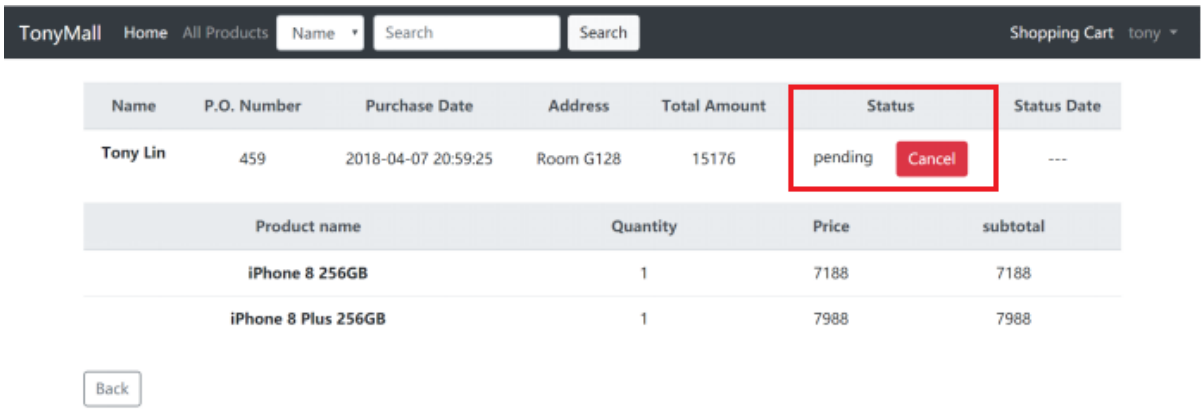


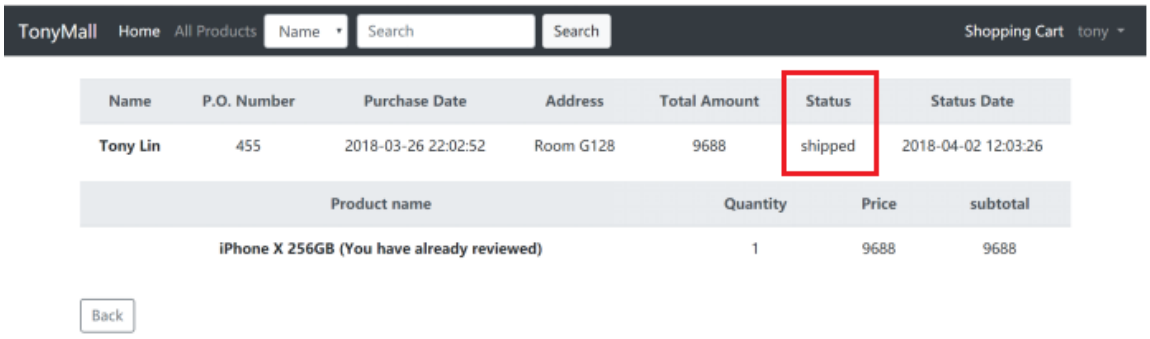
Result of “Past Product” filter

Customers are able to see the detail in purchase order detail page through clicking the “More” button.

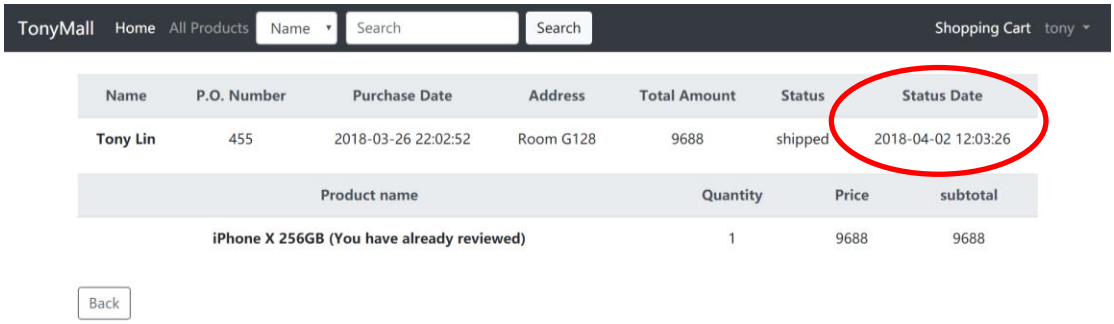


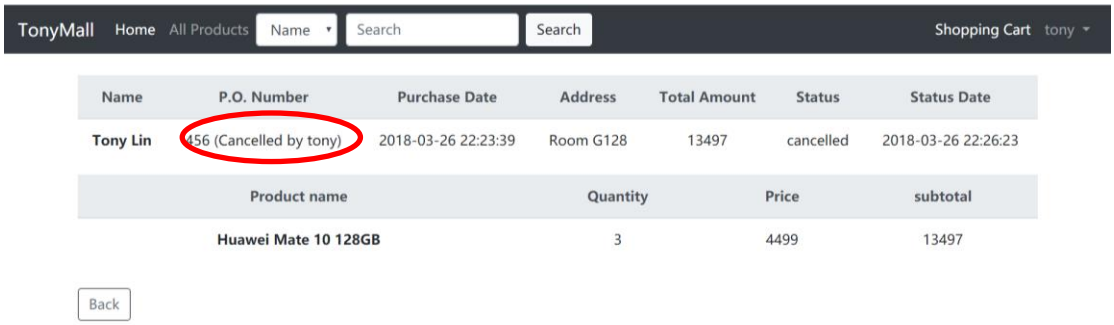
The customer can cancel the order by clicking the “cancel” button in the purchase order detail page before a purchase order is shipped. Besides, “cancel” button is not available for orders that have been shipped.





The purchase order detail page can show the shipment date of the shipped order and the order cancel data and who canceled the order of the canceled order. The page also shows each product’s information in the purchase order.



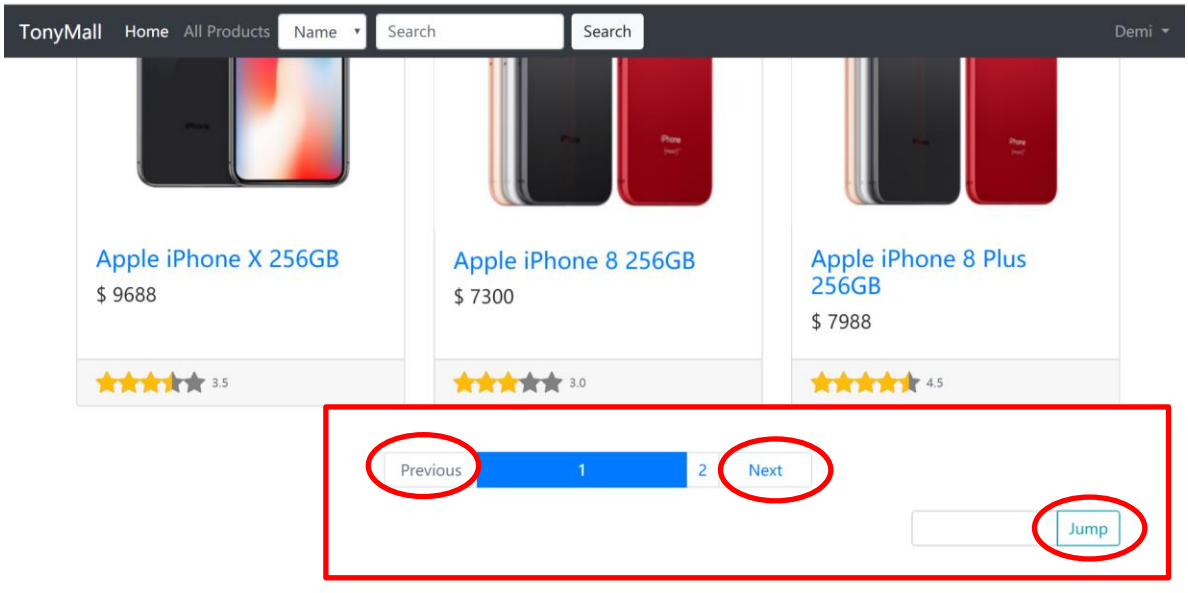


5.1.2 Features for vendors

Various features are provided for vendors in TonyMall.

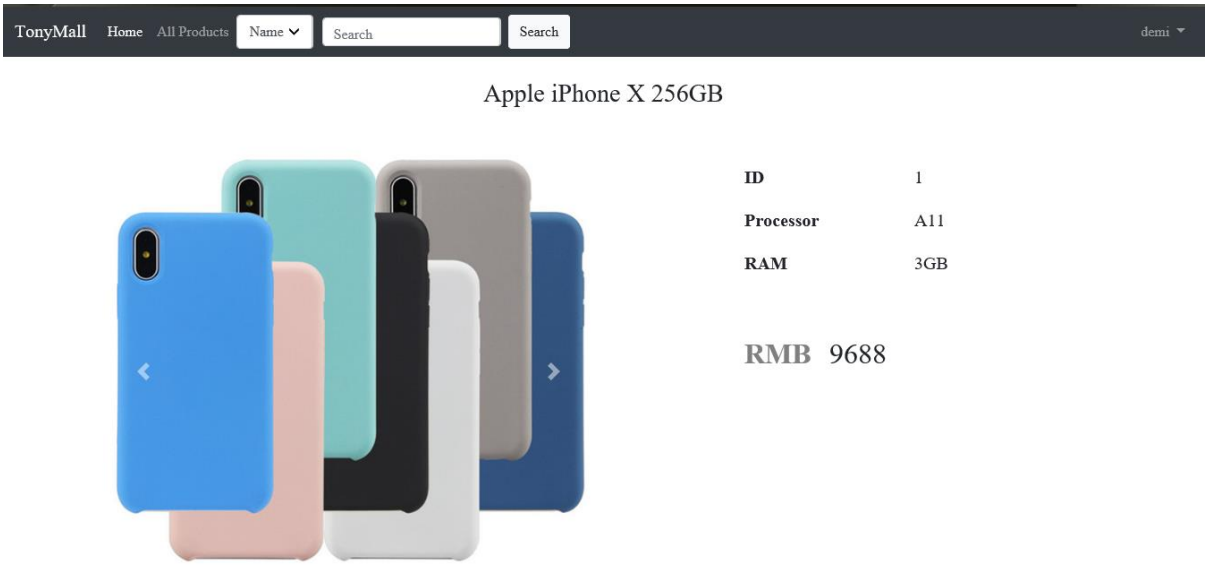
- Product list

  A vendor may click ‘All Products’ to browse products in the product list. The list shows basic information of products. The vendor can navigate the product list by ‘page up’, ‘page down’ and jump to a specific page, similar to the customer’s interface.



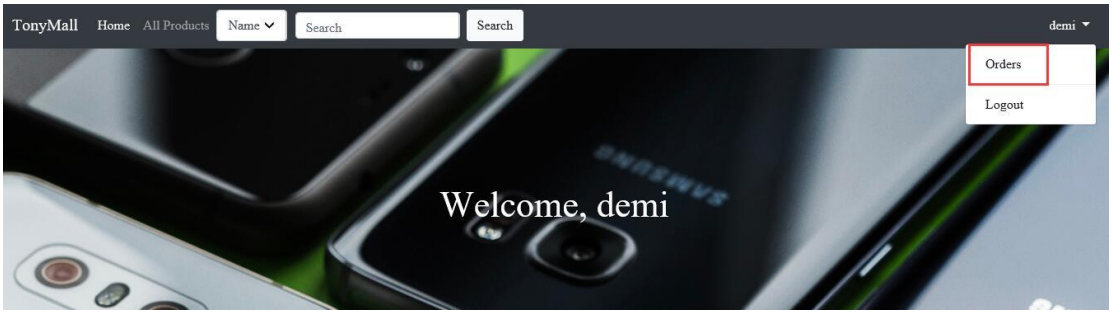
In addition, the vendor may filter the product list by the keywords of product name, brand or ID.

The vendor can select a product to go to the product detail page which shows more information about that product. Shopping cart button is not available for the vendor.

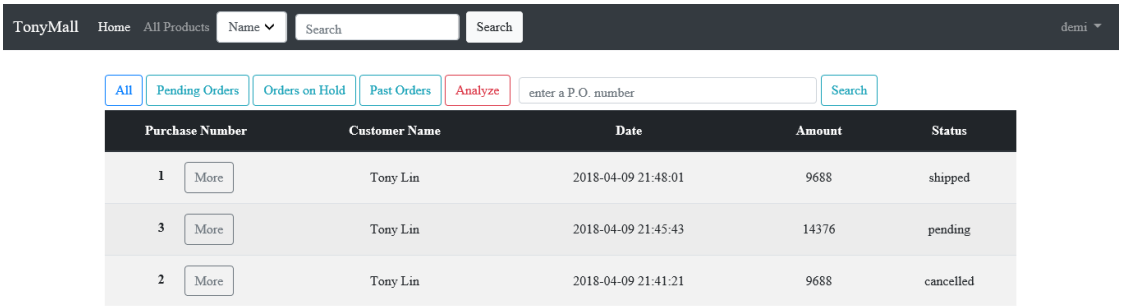


- Purchase order processing

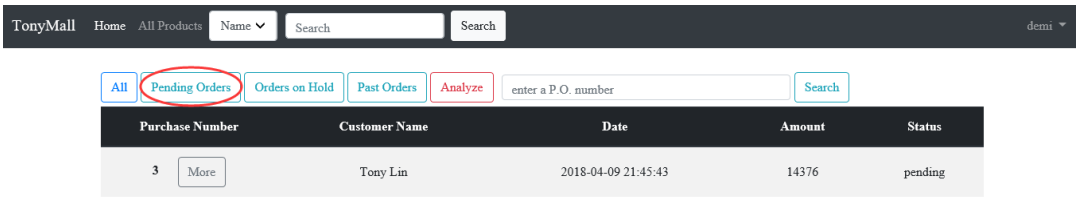
  The vendor can click the ‘order’ button to open a purchase order processing page.

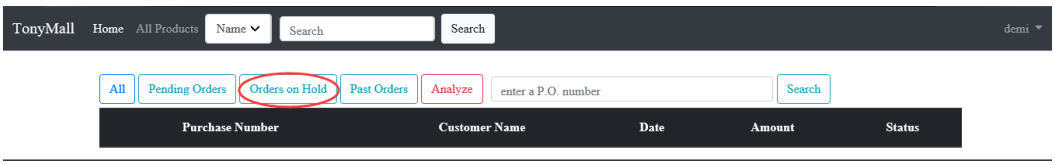


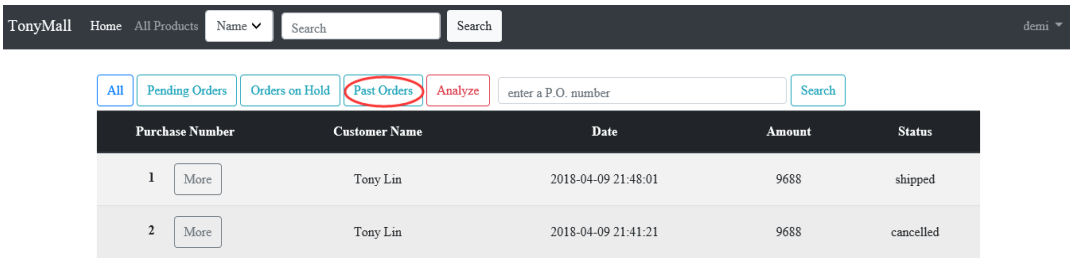
The vendor is able to view and process all the orders which customers have placed.



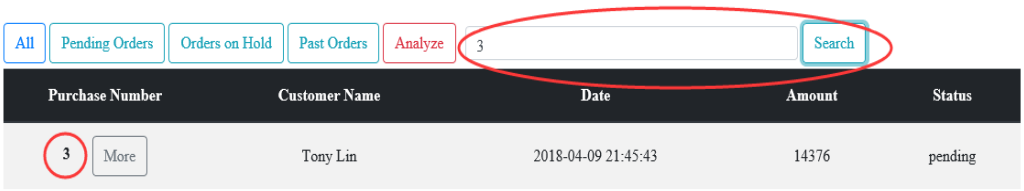
The vendor can filter the purchase order list by ‘pending orders’, ‘orders on hold’ and ‘past orders’.







The vendor can enter a P.O. number to view and process a specific purchase order.



The vendor can hold a purchase order whose status is ‘pending’.

Name	P.O. Number	Purchase Date	Address	Total Amount	Status	Status Date
Tony Lin	3	2018-04-09 21:45:43	Room G128	14376	pending Ship Hold Cancel	---

Product name	Quantity	Price	subtotal
iPhone 8 256GB	2	7188	14376

All Pending Orders Orders on Hold Past Orders Analyze 3 Search

Purchase Number	Customer Name	Date	Amount	Status
3 More	Tony Lin	2018-04-09 22:40:29	14376	hold

The vendor can ship a purchase order whose status is ‘pending’, or unhold and ship a purchase order whose status is ‘hold’.

Name	P.O. Number	Purchase Date	Address	Total Amount	Status	Status Date
Tony Lin	3	2018-04-09 22:40:29	Room G128	14376	hold Ship Cancel	2018-04-09 22:40:29

Product name	Quantity	Price	subtotal
iPhone 8 256GB	2	7188	14376

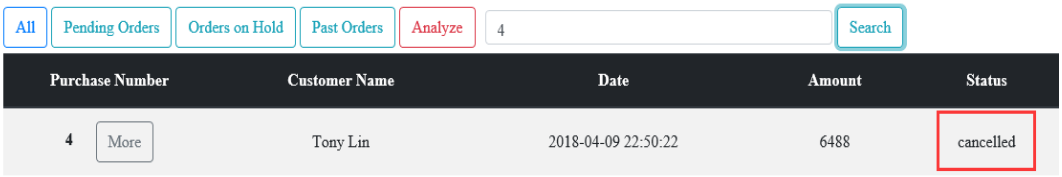
All Pending Orders Orders on Hold Past Orders Analyze 3 Search

Purchase Number	Customer Name	Date	Amount	Status
3 More	Tony Lin	2018-04-09 22:46:21	14376	shipped

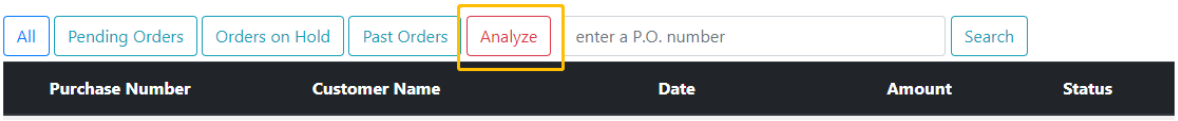
The vendor can cancel a purchase order whose status is ‘pending’ or ‘hold’.

Name	P.O. Number	Purchase Date	Address	Total Amount	Status	Status Date
Tony Lin	4	2018-04-09 22:28:17	Room G128	6488	pending Ship Hold Cancel	---

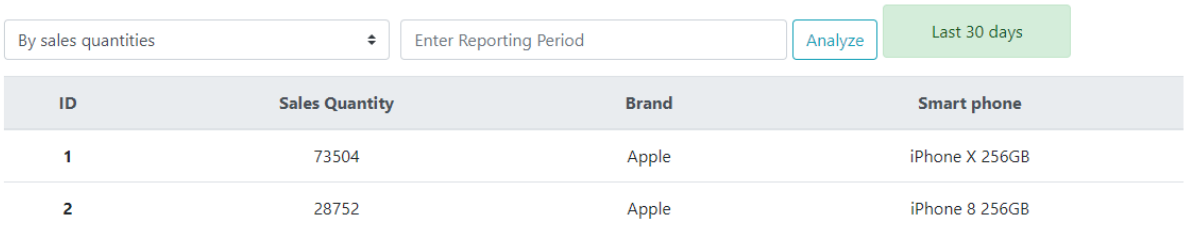
Product name	Quantity	Price	subtotal
Samsung Note 8 64GB	1	6488	6488



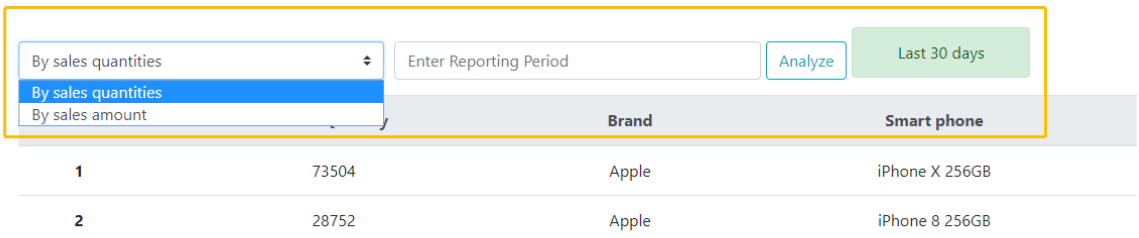
The vendor can also analyze the best selling products by clicking ‘Analyze’ button.



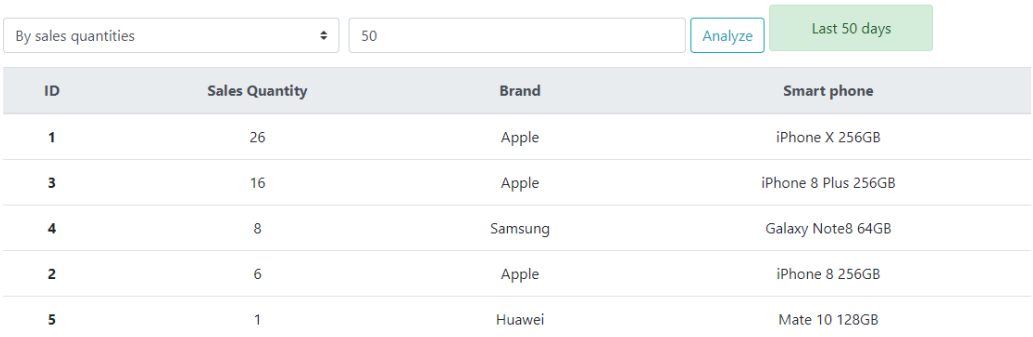
The default period is 30 days.



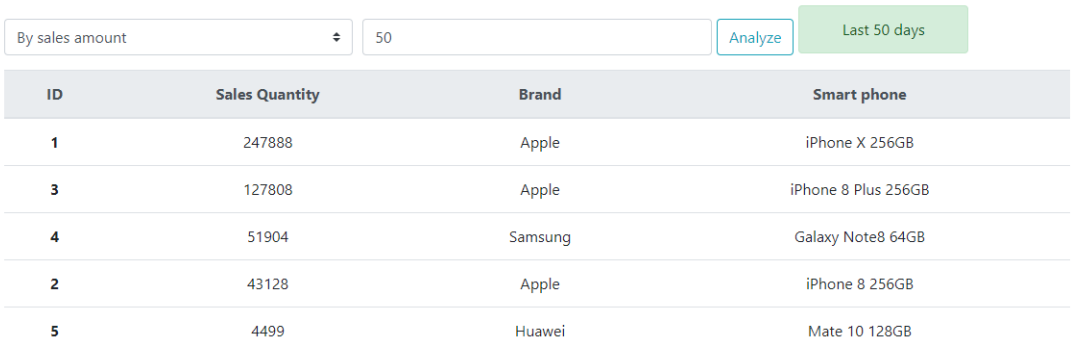
Furthermore, vendor can customize the reporting period and search by sales quantities or by sales amount. The green box will show the period.



For example, this is the result of search the best selling products by sales quantities last 50 days.



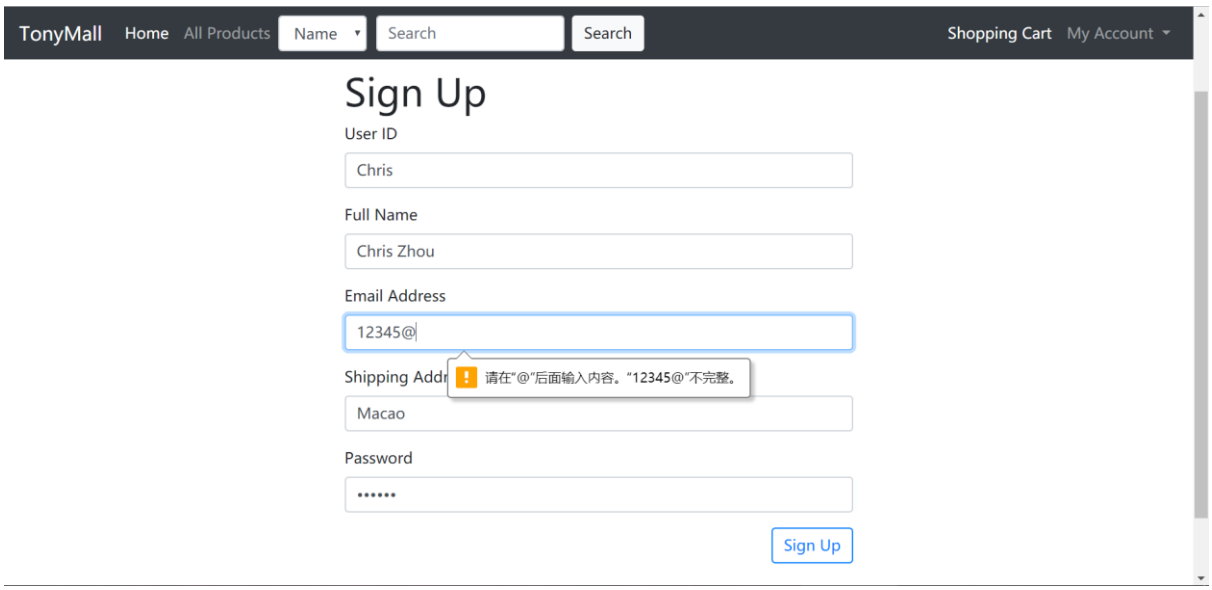
This is the result of search the best selling products by sales amounts last 50 days.



ID	Sales Quantity	Brand	Smart phone
1	247888	Apple	iPhone X 256GB
3	127808	Apple	iPhone 8 Plus 256GB
4	51904	Samsung	Galaxy Note8 64GB
2	43128	Apple	iPhone 8 256GB
5	4499	Huawei	Mate 10 128GB

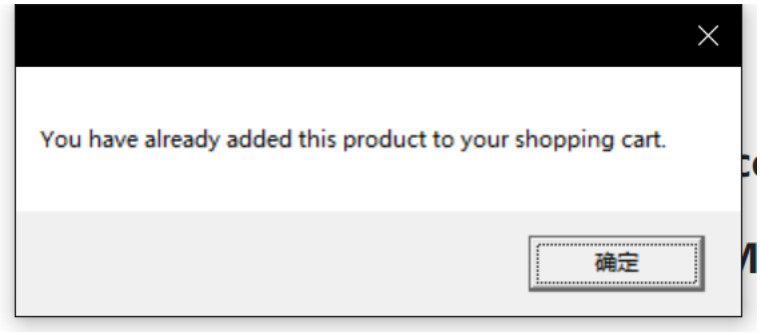
5.2 Testing and System Evaluations

We implement testing in many ways. For instance, in register page, we use regular expression to make sure that user type a correct and valid input. The first letter in "username" must be uppercase, the email address must contain "@". The password must be at least 6 digital. In testing, we type "12345" into email address blank, the system will remind that "@" must be included. Then we type "12345@", the system says that there must be some character after "@". Next, we type "12345@g.com", finally the system can accept that.

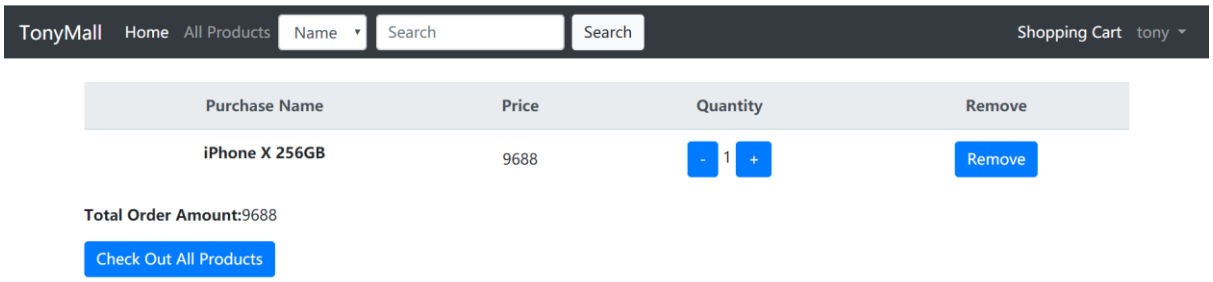


We found that a user may add the same product to the shopping cart, which will result in duplicate product and data inconsistency. Thus, we implement a checking method to check whether there is the same item already exists inside the shopping cart. For example, we add

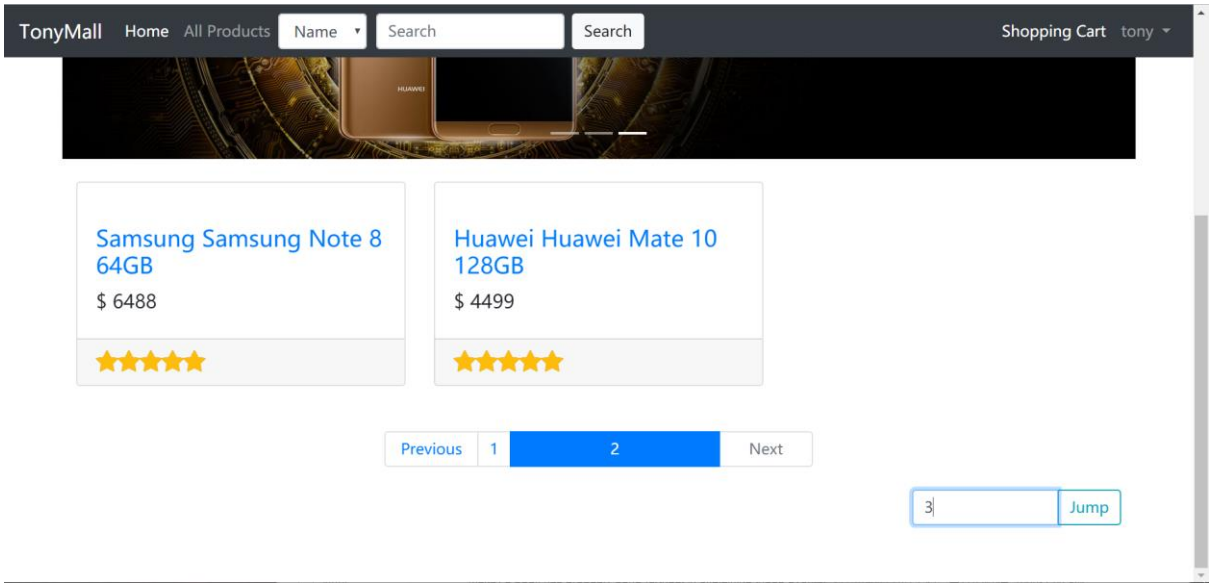
iPhone X into the shopping cart first, it's successful. But when we try to add again, an alert window appears, which telling the user that the product is already inside.



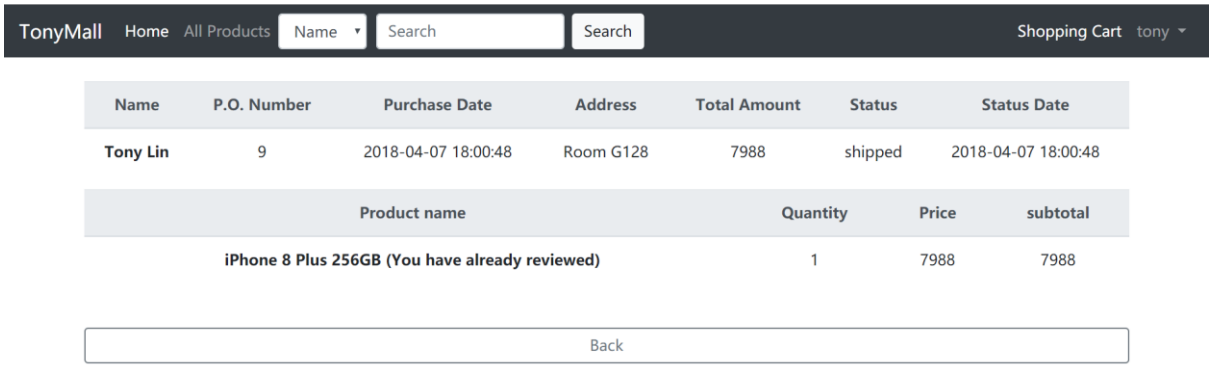
In shopping cart, a product quantity cannot be equal to 0, otherwise it occurs to be removed from the shopping cart rather than being "0" and stay there. We add an iPhone X into shopping cart, and change the quantity of iPhone X. The quantity can be increased infinitely and successfully. Then we decrease the quantity. When the quantity is equal to "1", the quantity won't be decreased to "0", it's still "1". If a user wants to remove the product in shopping cart, he only has to click the "Remove" button.



In product page, a user may want to go to another page, so he types a number in the blank and the system will jump to that page. But sometimes a user can type a huge number that dose not exist. There is only 2 pages in the product page, and when I type 2 in the blank, it will jump to the second page. If I type 3, the system will jump to the furthest page (i.e. page 2) rather than page 3, which can prevent user from going to a blank page.



In rating system, a user should review only once, and he can't change or delete what he just wrote down after he submitted the review. In TonyMall, we made a review for iPhone X after we received it. And when we go back to the purchase order of that iPhone X, we find that it said "You have already reviewed", and there is no "review" button any more.



6 Conclusion and Further Work

We have achieved lots of functions in this online shopping mall project. The first main part is for customers. First of all, we have done the product list part for customer view. We display our products in pages. Besides, we allow customers to search our products by name or brand. We also provide corresponding detail page for each product. Customers are able to see a product's detail page by clicking the product in the product page.

Secondly, we complete user registration, login and logout parts. Through our e-commerce website, users are able to register an account and then they can just login easily to see their shopping cart and purchasing order at any time.

Thirdly, we finish the shopping cart part. Customers can add or remove products in the shopping cart easily. They can also change the number of products in the shopping cart conveniently.

Fourthly, we have done the purchase tracking part. Customers can see their purchase order status and choose to cancel the purchase order that is in pending status. The last important point is that we have designed some great features for customers. We provide a platform for customers to review and give star rating for the products that they have purchased and the feedback is transparent to both vendor and all customers.

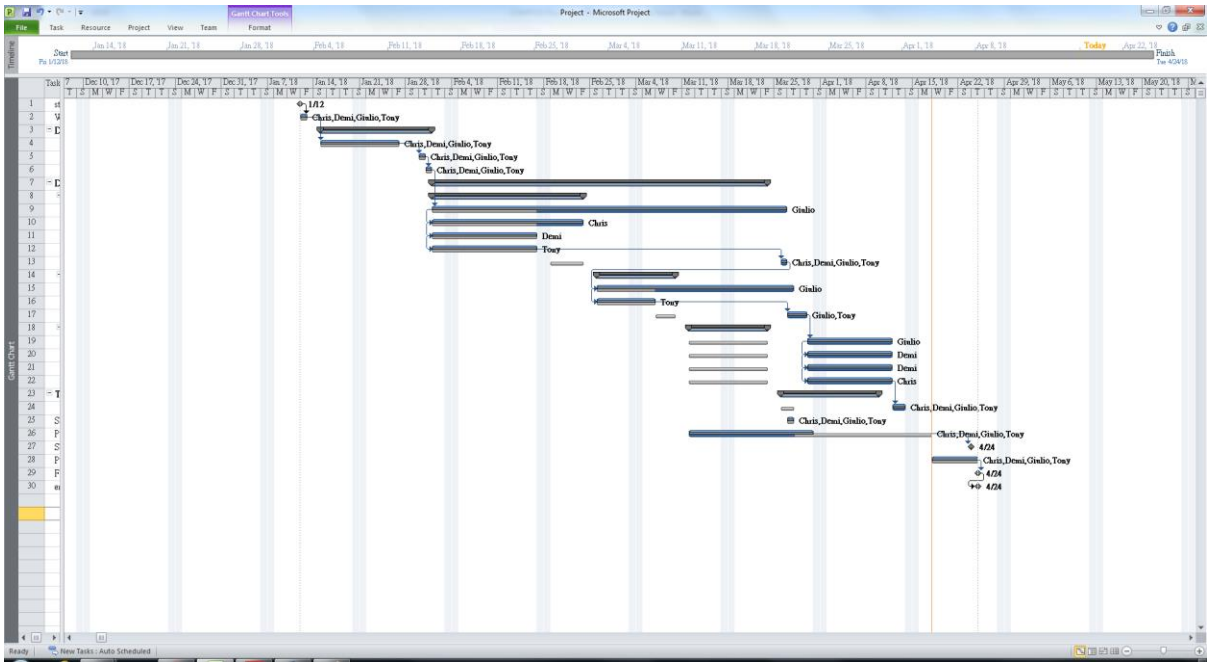
The second main part is vendor part. There is only one vendor in our e-commerce platform. Firstly, we display products for the vendor and "shopping cart" will not be available in the vendor view. Besides, vendor can easily find products through a product ID or name. Secondly, vendor can see all purchase orders and has the right to manage the purchase order status.

Actually, there are still some shortcomings for our e-commerce website. The most obvious disadvantage is for the vendor, he or she is unable to add new products or edit the information of products. In the future, we plan to provide the vendor a platform to add new products or delete existing products. Another problem is that customers are unable to sort products by price through our e-commerce platform. We design to add a sorting by price button in the future. The last weakness for our website is that customers will not receive any notification for the change of purchase order status. We plan to add a notification function for informing customers the change of status of their purchase orders in the future.

Appendix A. Project Management

Gantt Chart

	Task Name	Work	Duration	Start	Finish	Actual Start	Actual Finish
1	start	0 hrs	0 days	Fri 1/12/18	Fri 1/12/18	NA	NA
2	Work Assignments and Plan	32 hrs	1 day	Fri 1/12/18	Fri 1/12/18	Fri 1/12/18	Fri 1/12/18
3	**Database**	**384 hrs**	**13 days**	**Mon 1/15/18**	**Wed 1/31/18**	**Mon 1/15/18**	**Wed 1/31/18**
4	Database Design	320 hrs	10 days	Mon 1/15/18	Fri 1/26/18	Mon 1/15/18	Fri 1/26/18
5	ER Diagram	32 hrs	1 day	Tue 1/30/18	Tue 1/30/18	Tue 1/30/18	Tue 1/30/18
6	Data Entry (MySQL)	32 hrs	1 day	Wed 1/31/18	Wed 1/31/18	Wed 1/31/18	Wed 1/31/18
7	**Development**	**1,232 hrs**	**37 days**	**Thu 2/1/18**	**Fri 3/23/18**	**Thu 2/1/18**	**Wed 4/11/18**
8	**Customers**	**664 hrs**	**17 days**	**Thu 2/1/18**	**Fri 2/23/18**	**Thu 2/1/18**	**Mon 3/26/18**
9	Develop Product List (A)	304 hrs	38 days	Thu 2/1/18	Mon 3/26/18	Thu 2/1/18	Mon 3/26/18
10	Develop Account Management (B)	136 hrs	17 days	Thu 2/1/18	Fri 2/23/18	Thu 2/1/18	Fri 2/23/18
11	Develop Shopping Cart (C)	96 hrs	12 days	Thu 2/1/18	Fri 2/16/18	Thu 2/1/18	Fri 2/16/18
12	Develop Pruchase Tracking (D)	96 hrs	12 days	Thu 2/1/18	Fri 2/16/18	Thu 2/1/18	Fri 2/16/18
13	Test and Integration	32 hrs	1 day	Mon 3/26/18	Mon 3/26/18	Mon 3/26/18	Mon 3/26/18
14	**Vendor**	**280 hrs**	**10 days**	**Mon 2/26/18**	**Fri 3/9/18**	**Mon 2/26/18**	**Thu 3/29/18**
15	Develop Product Catalog Maintenance (E)	176 hrs	22 days	Mon 2/26/18	Tue 3/27/18	Mon 2/26/18	Tue 3/27/18
16	Develop Purchase Order Processing (F)	56 hrs	7 days	Mon 2/26/18	Tue 3/6/18	Mon 2/26/18	Tue 3/6/18
17	Test and Integration	48 hrs	3 days	Tue 3/27/18	Thu 3/29/18	Tue 3/27/18	Thu 3/29/18
18	**Advanced Features**	**288 hrs**	**10 days**	**Mon 3/12/18**	**Fri 3/23/18**	**Fri 3/30/18**	**Wed 4/11/18**
19	Z1 - Analyze	72 hrs	9 days	Fri 3/30/18	Wed 4/11/18	Fri 3/30/18	Wed 4/11/18
20	Z3 - Rating	72 hrs	9 days	Fri 3/30/18	Wed 4/11/18	Fri 3/30/18	Wed 4/11/18
21	Z4 - Reviews	72 hrs	9 days	Fri 3/30/18	Wed 4/11/18	Fri 3/30/18	Wed 4/11/18
22	Z5 - Price Change of Products	72 hrs	9 days	Fri 3/30/18	Wed 4/11/18	Fri 3/30/18	Wed 4/11/18
23	**Testing**	**64 hrs**	**11 days**	**Mon 3/26/18**	**Mon 4/9/18**	**Thu 4/12/18**	**Fri 4/13/18**
24	Test and Integration	64 hrs	2 days	Thu 4/12/18	Fri 4/13/18	Thu 4/12/18	Fri 4/13/18
25	Submit Peer Assessment Form	32 hrs	1 day	Tue 3/27/18	Tue 3/27/18	Tue 3/27/18	Tue 3/27/18
26	Project Report	480 hrs	15 days	Mon 3/12/18	Tue 4/17/18	Mon 3/12/18	Tue 4/17/18
27	Submit Project Report	0 hrs	0 days	Tue 4/24/18	Tue 4/24/18	Tue 4/24/18	Tue 4/24/18
28	Prepare Presentation	160 hrs	5 days	Wed 4/18/18	Tue 4/24/18	Wed 4/18/18	Tue 4/24/18
29	Final Presentation & Demonstration	0 hrs	0 days	Tue 4/24/18	Tue 4/24/18	Tue 4/24/18	Tue 4/24/18
30	end	0 hrs	0 days	Tue 4/24/18	Tue 4/24/18	Tue 4/24/18	Tue 4/24/18



Appendix B. Peer Assessment Form

澳門理工學院 Instituto Politécnico de Macau Macao Polytechnic Institute	BSc. in Computing 2017/18 COMP321 Information System Implementation Peer Assessment Form		
Project number	1		
Team members		Student ID	Student name
	1.	*Demi*	P1507862
	2.	*Tony*	P1507921
	3.	*Chris*	P1507881
	4.	*Giulio*	P1507805
	5.		

Contribution **(**Each row must total to 100%)

	Demi	*Tony*	*Chris*	*Giulio*	
1. Project leadership	50%	30%	20%	0%	%
2. Data modeling	25%	25%	25%	25%	%
3. User interface design	10%	70%	10%	10%	%
4. Program development	30%	30%	30%	10%	%
5. Solving technical problems	25%	40%	25%	10%	%
6. Testing and sample data	25%	25%	25%	25%	%
7. Report writing	25%	25%	25%	25%	%
8. Preparing / giving presentation	25%	25%	25%	25%	%

By default, the eight items above have the same weight when calculating the overall contribution percentage. You are welcome to suggest different weight if you consider some aspects should carry more weight.

Appendix C. Project Specification

- (A1) A customer may browse products in a list of products. The list shows basic information of products, including product name, brand, price and a thumbnail image. Each product belongs to one of the pre-defined brands. (You can also use category instead of brand).
- (A2) 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 properly after applying a filter or sorting as listed below.
- (A3) The customer can filter the product list by brand. They can also list products of all brands.
- (A4) The customer may filter the product list by searching keywords in the product name. This function work correctly with the brand filter.
- (A6) 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, which includes the product name, brand, price and a thumbnail image. In addition, the product detail page also shows detail 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 number of pages.
- (A7) The product detail page supports display of one or more detailed photographs of the selected product.
- (B1) A customer may register a new account. They have to provide full name, email address, password and shipping address. After registration, the user is logged in automatically.
- (B2) A customer may log in and log out, and the interface shows the name of the current user. 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.
- (B4) 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.
- (C1) The customer adds a product to his/her shopping cart by clicking a button in the product detail page. The quantity to buy is assumed to be 1. The items in

shopping cart are persisted across user sessions. Next time the customer logs in, they can still see the items in the shopping cart.

- (C2) The customer can list the products in his/her shopping cart in a shopping cart page. In 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) The customer can press a button in 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) 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) The customer can remove an item from the shopping cart.

  (C6) If the customer adds a duplicate product to the shopping cart, the application will give a warning message and does not change the content of the shopping cart.
- (D1) 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 purchase date. When the customer clicks an entry in the list, they can see the detail in a purchase order detail page.
- (D2) 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 status ‘shipped’ and ‘cancelled’.
- (D3) 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 cancelled, the page shows the order cancel date and who (customer or vendor) cancelled the order. The page also includes, for each product in the purchase order, the product name, the quantity, the unit price and the subtotal.

- (D4) Before a purchase order is shipped, the customer can cancel the order. This can be done by clicking a button in the purchase order detail page. This action will change the status of the purchase order to ‘cancelled’. Note that this action is only available for purchase orders in the status ‘pending’ or ‘hold’.
- (E1) The vendor may browse the product catalog in an interface similar to product list for customers. *(Refer to requirements A1, A2, A3 and A6)*. The vendor is not a customer, and no shopping cart or ‘add to cart’ button should be shown.
- (E2) The vendor can find products by searching keywords in product names. They can also find a specific product by entering a unique product ID.
- (F1) 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) The vendor can filter the purchase order list in three ways. They can show only the ‘pending orders’ (with status ‘pending’). They can show only the ‘orders on hold’ (with status ‘hold’). Finally, the vendor can select to show ‘past orders’ (with status ‘shipped’ or ‘cancelled’).
- (F3) The purchase order processing page shows similar information as the purchase order detail page *(refer to requirement D3)*. In addition, 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) The vendor can enter a P.O. number to view and process a specific purchase order.
- (F5) In 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) In 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) In 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’.

- (Z1) The vendor needs to analyze the sales of the products and find out the best selling products. The report measures sales by both sales quantities (number of items sold) and sales amount (the dollar amount received in sales). The default reporting period is the last 30 days, but the vendor may also customize the reporting period.
- (Z3) A customer can express their satisfaction with a product with 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. Decide whether a customer can rate one product more than once, and whether they can change the ratings afterwards. The product detail page shows customers’ average rating as a decimal number (e.g. 3.5 stars). Pay special attention when number of ratings is smaller (e.g. less than 2). Consider how to use the average ratings in product list for customers and the vendor.
- (Z4) In addition to star ratings, customers also want to write short reviews for products in the shopping mall. Design a feature to allow a customer to write a short review for a product. Consider how to show these reviews to other customers and the vendor.
- (Z5) Design how to implement price change of products. This is useful, e.g., for promotional price reduction or regular price adjustment. Price change should not affect the price in existing purchase order and other historical records.