# System Design

In this chapter, we will describe the data modeling and dynamic modeling for our mobile online shopping application. In the data modeling section, we will use Entity Relationship Diagram to describe the structure of our database and our design decisions. In the dynamic modeling section, we will use state diagrams, activity diagrams, and sequence diagrams to illustrate the interaction and dynamic aspects of our system.

## Data Modelling

In this section, we will use the Entity Relationship Diagram to describe the main entities of our system and the relationships between them.

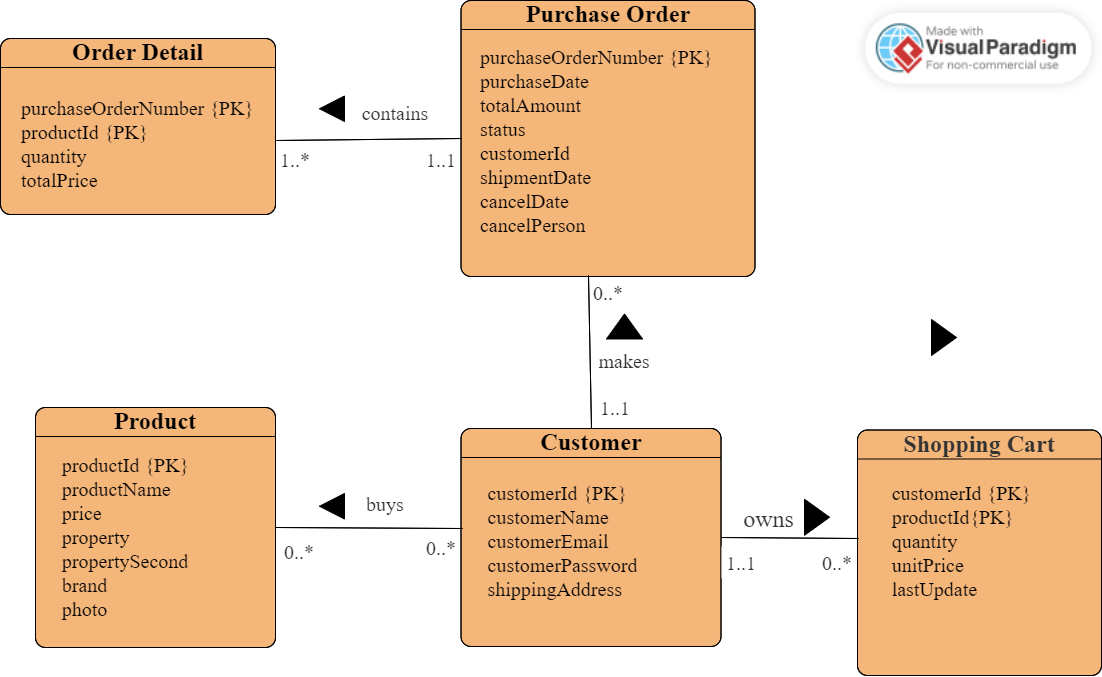


Figure 3-1: ER Diagram

Figure 3-1 illustrates our ER diagram for Niubility. The main entities are:

* Customer: This entity represents the user who has registered in Niubility (Note that we assume there is only one vendor in this application). It stores information about customers’ unique ID number, names, email addresses, password, and shipping addresses.
* Product: This entity stores information about each product that is available for sale on Niubility, such as its distinctive ID number, name, price, two properties, brand, and photo.
* Shopping Cart: This entity stores all the products the customer chooses which also includes its quantity, unit price, and date when it is added.
* Purchase Order: This entity stores information about all orders the customer makes, which includes its unique number, purchase date, total amount, order status, and customer Id. If the products of this order are shipped, it will record the shipment date. Or the order is canceled, it will record cancel person and cancel date.
* Order Detail: This entity stores the detailed information of each order, such as the specific product(productId), the number of the specific product, and the total price of this product.
* Preference: This entity stores a list of products which are stamped as “like”, including the unique customer Id number, product Id number, and the identifier whether the customer like this product.

The relationships between these entities are:

* A customer can buy zero or many products. A product can be bought by zero or many customers.
* A customer can own zero or many shopping carts(Note: the entity Shopping Cart records one product and one customer at one time. You could read the figure below to get better understanding). A shopping cart can be owned by one and only one customer.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| customerId | productId | quantity | unitPrice | lastUpdate |
| 1 | 6 | 2 | 5000 | 2023-3-26 15:00:06 |

Fig. : one shopping cart in the database

* A customer can make zero or more purchase orders. A purchase order can be made by one and only one customer.
* A purchase order can contain one or many order details. But an order detail can be contained by one and only one purchase order.
* A customer can have one or more preferences. But one preference could only belong to one customer.

The logical data model is listed below, giving further details on the foreign key allocation.

Customer(customerId, customerName, customerEmail, customerPassword, shippingAddress)

Primary Key: customerId

Product(productId, productName, price, property, propertySecond, brand, photo)

Primary Key: productId

Shopping Cart ( customerId, productId, quantity, unitPrice, lastUpdate)

Primary Key: customerId, productId

Foreign Key: customerId references Customer(customerId)

productId refereces Product(productId)

Purchase Order (purchaseOrderNumber, purchaseDate, totalAmount, status, customerId, shipmentDate, cancelDate, cancelPerson)

Primary Key: purchaseOrderNumber

Foreign Key: customerId references Customer(customerId)

Order Detail (purchaseOrderNumber, productId, quantity, totalPrice)

Primary Key: purchaseOrderNumber, productId

Foreign Key:

purchaseOrderNumber references Purchase Order(purchaseOrderNumber)

productId refereces Product(productId)

Preference (customerId, productId, like)

Primary Key: customerId, productId

Foreign Key: customerId references Customer(customerId)

productId refereces Product(productId)

Figure ? shows the data dictionary of all the entities and the related attributes of Niubility.

## 3.2 Dynamic Modelling

The dynamic modeling section of an online shopping mall project report should describe the interaction and dynamics of our system. In this section, we will use the state diagram, activity diagram, and sequence diagram to model the dynamic behavior of our mobile online shopping application.

### **State Diagrams**

The state diagram is to describe the behavior of a single object in response to a series of events in a system. Figure 3-2-1-1 shows the order state diagram and the order state changes with the interaction between the customer and vendor. This state diagram shows the different states that an order can be in: “pending”, “hold”, “shipped”, and “cancelled”. First, the state of the order is “pending” after the customer checks out. After that, the vendor can change the state to “hold” or “shipped” according to the stock. For out-of-stock, if there is stock later, the vendor can change the state from “hold” to “shipped”. In addition, both customer and vendor can only change the state to “cancelled” if the order is in the “pending” or “hold” state.

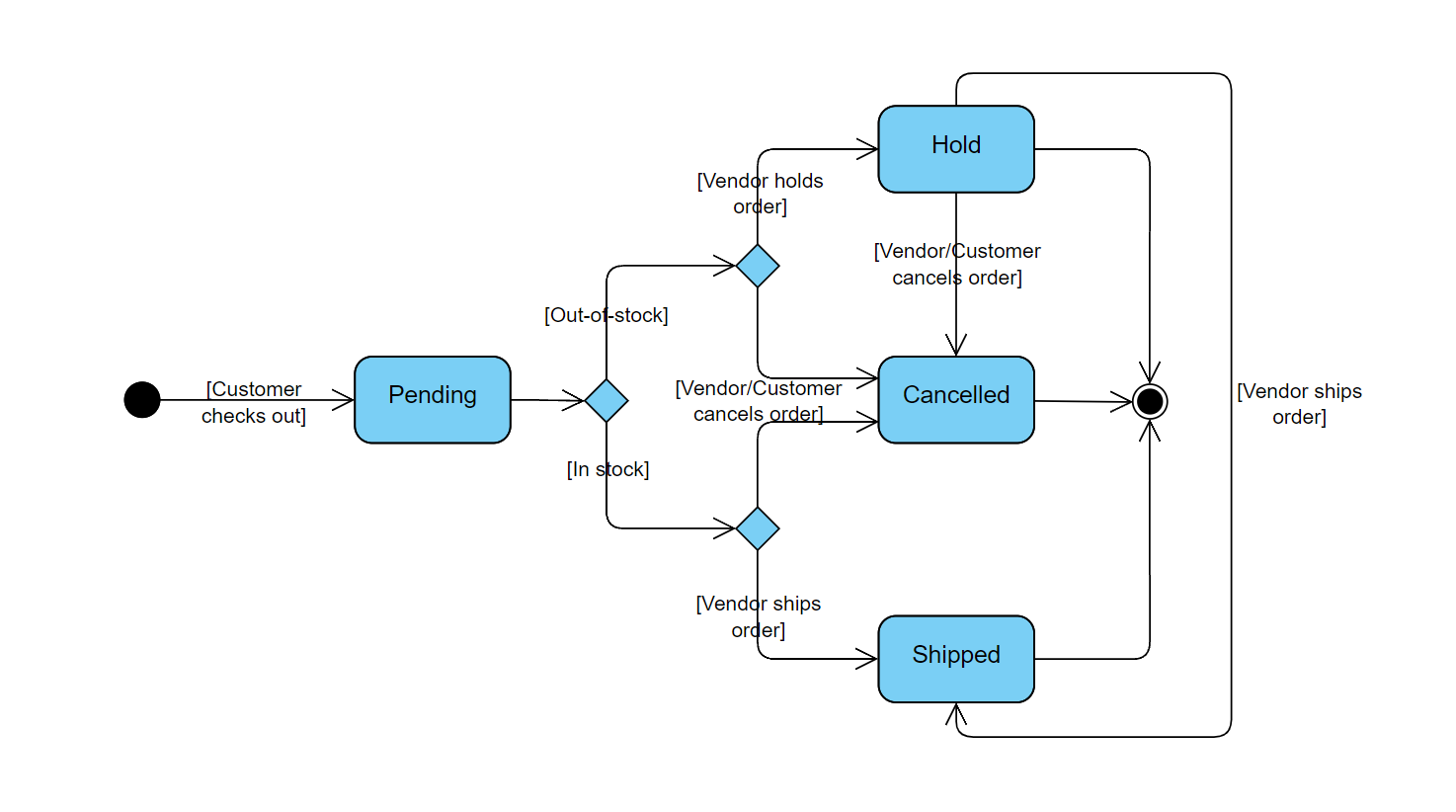


Figure 3-2-1-1: Order State Diagram

### **Activity Diagrams**

The activity diagram is to show the flow from one action to another in a system or process. In this section, we will use three activity diagrams to model the behavior of customer who is not logged in, customer logged in, and vendor.

Figure 3-2-2-1 shows the activity diagram for customers who are not logged in or the new user. In this diagram, customers who are not logged in (including the new users) can only operate on a small number of pages, such as the home page, search page, and product detail page. First, customers can go to the home page to browse the product list and they can filter the products by brands. Moreover, if they want to find the products that they are interested in, they can search the products by keywords on the search page, and they can also filter the products by brands on the search page.

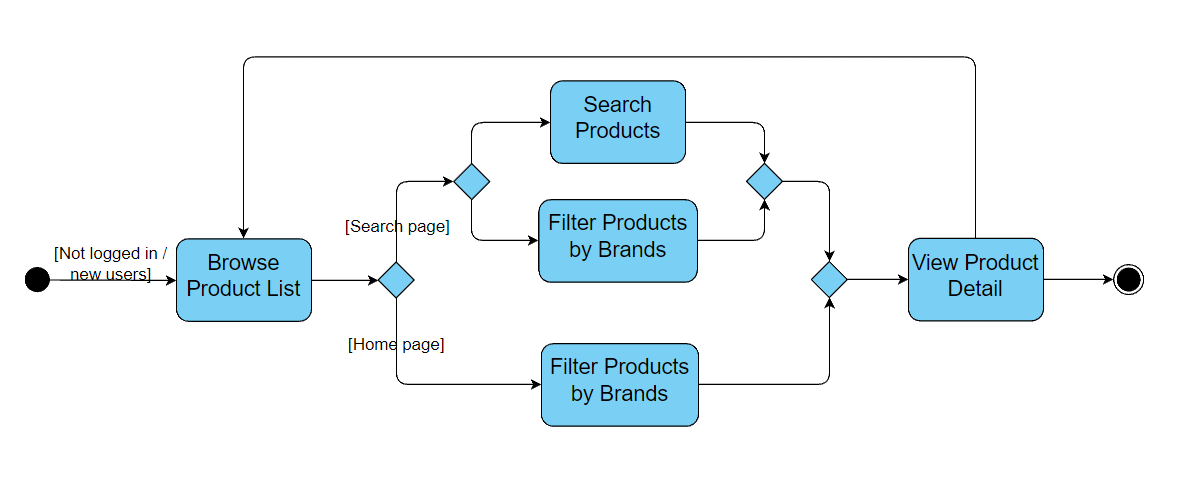


Figure 3-2-2-1: Customer Not Logged In or New Customer

Figure 3-2-2-2 illustrates the customer activity diagram that describes the activities of the customers who can log in. First, customers can log in to the system by entering their login credentials. For the new users, they can create their accounts on the register page, and after registration, they will log in automatically. Next, they can browse the products and add the products they want to buy to their shopping cart. In the shopping cart, customers can choose to change the quantity of the items they added or remove the items from their cart. Then, customers can check out and go to the confirm page. In addition, the shopping cart will become empty when they check out. On the confirm page, customers are allowed to change their addresses. After confirming the order, the system will show the order detail page of the newly created order. Additionally, customers can change their usernames and passwords on the account management page. At last, after the customers log in, they can directly navigate to the home page, shopping cart page, order list page, and account page according to their needs.

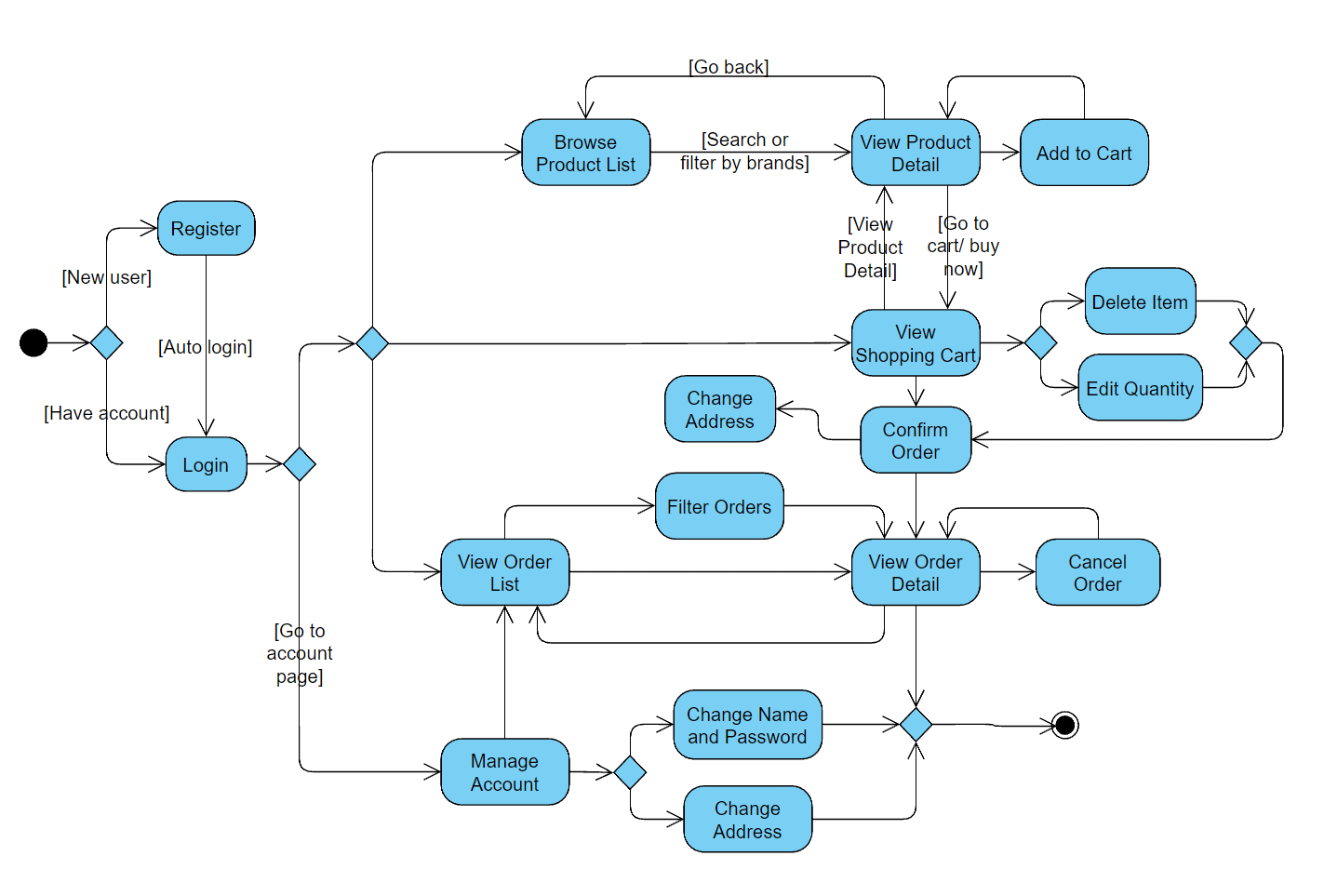


Figure 3-2-2-2: Customer Activity Diagram

Figure 3-2-2-3 exhibits the vendor activity diagram that shows the behavior of the vendor in the system. In this diagram, the vendor can enter detailed information about the new products and add them to the catalog. When browsing the product list, the vendor can search and filter the products in the same way as customers. In addition, the vendor can also find a specific product by entering the product ID. Moreover, the vendor can filter the orders by the order states or search for the specific order by entering the order ID. On the order detail page, the vendor can change the order states. The vendor can ship the orders if their states are “pending” or “hold” and hold the orders if the orders are in the “pending” state. The vendor can also cancel the order if necessary.

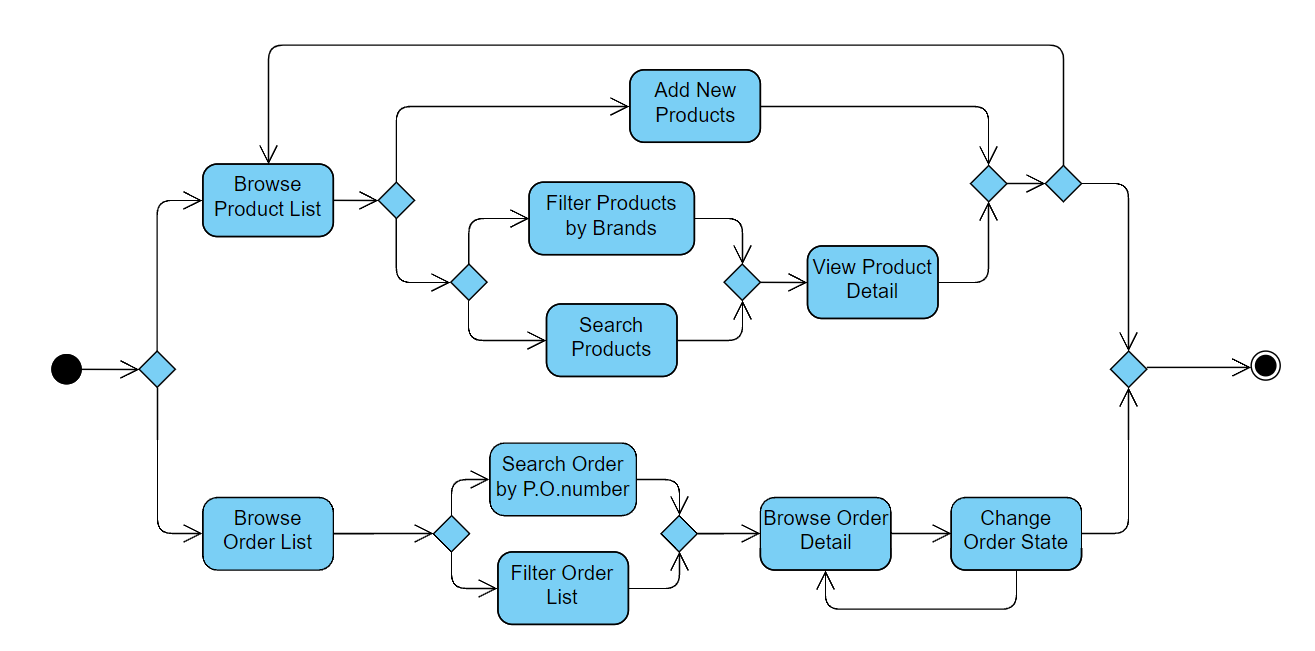


Figure 3-2-2-3: Vendor Activity Diagram

### **Sequence Diagram**

A sequence diagram consists of a group of objects that are represented by lifelines, and the messages that they exchange overtime during the interaction. In this section, we will use two sequence diagrams to illustrate the interaction between the customer, mobile online shopping app, and vendor.

Figure 3-2-3-1 shows the interaction between the customer and the app from the login to the checkout process. In the beginning, the existing users are required to enter their login credentials, and the new users are required to enter their registration information. After verification, the system will show the account page to users. Then, users can request to filter and search the products and the system will show the result to users. After the user clicks, the system will return the product detail page and the user can choose to add to their cart. Next, users can request the cart page to change the items’ quantity or remove the item from their cart. And then they can click to check out. After checking out, the system will show the confirm page to users and users need to confirm the order. Finally, the system will show the order detail page to users.

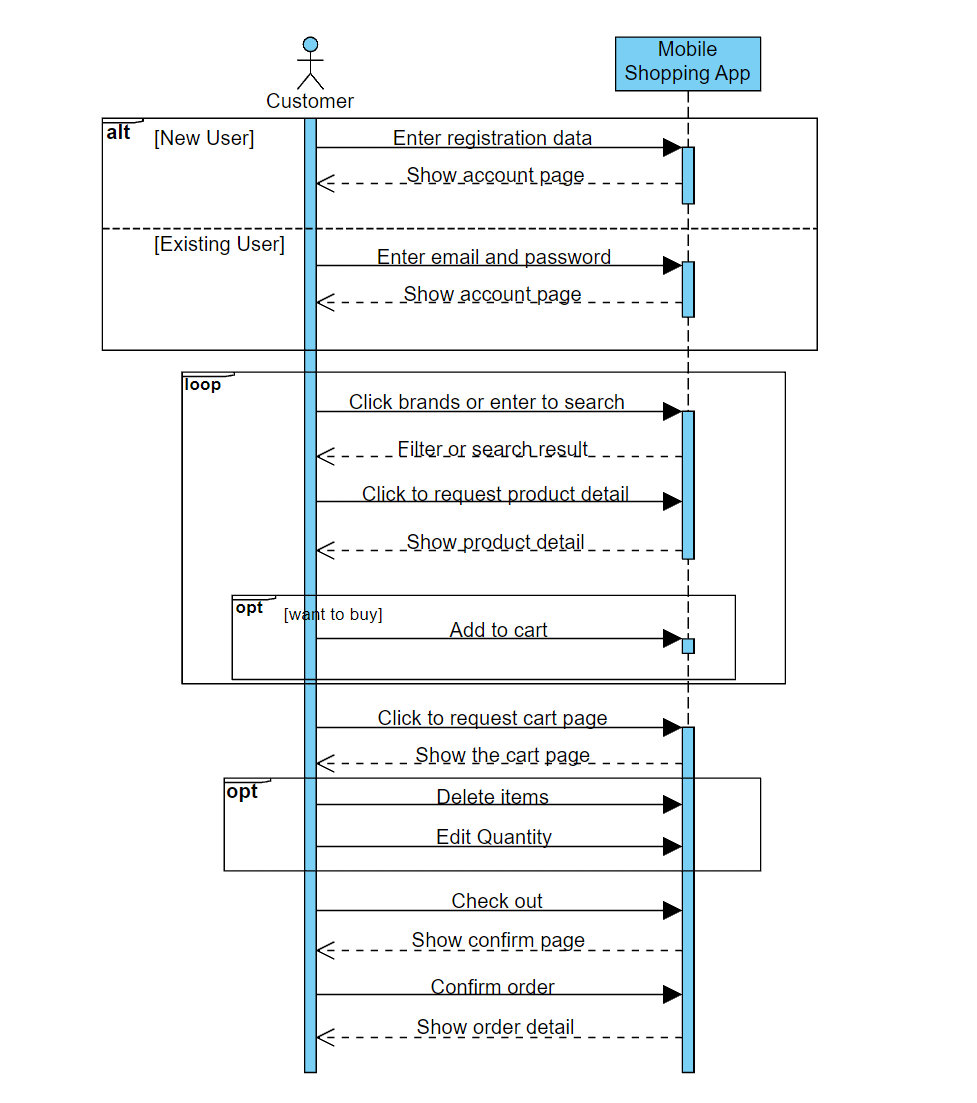


Figure 3-2-3-1: Customer Shopping Sequence Diagram