COMP3211 Software Engineering

Semester Project

**SOFTWARE REQUIREMENTS and DESIGN SPECIFICATION**

CHENG Hoi Yan 13069036D

XU Haiyan 13102963D

YIP Kai Yan 13068237D

**Part I: Software Requirements Specification**

**1.0 Background**

**1.1 Role of each Team Member**

Customer: CHENG Hoi Yan

She is the manager of a restaurant. She signed an contract with the project team, asking project team to write an app for ordering food in her restaurant..

Software Project Manager:XU Haiyan

She is responsible for managing the project, distribute works to team members and make sure the project finished on time.

Software Project Team Member:YIP Kai Yan

She is responsible for programming and interface design in the project team.

**2.0 Introduction**

**2.1 Goals and objectives**

We are going to develop an app operating on mobile phone. The app allows customers of Bamboo Coffee Shop to order food themselves. It also offer them with discount of taking dinner here. So that through the app, Bamboo Coffee Shop can not only attract more customers but also increase the efficiency and patronage during the peak hour . It can release workload and responsibilities of staff since the app can let customer order food themselves.

**1.2 Statement of scope**

Essential Requirement:

* Customer order food through the app.
* Customer check receipt/what they have ordered through the app.
* Staff know what to cook without talking to customer.
* Staff know how much customer should pay by scan bar code on the app.

Future Requirement

* Customer pay for the meal through the apps.

**1.3 Software context**

CHENG Hoi Yan, the manager of Bamboo Coffee Shop signed up with us to develop an app operated on mobile phone for her Coffee Shop. The average consumption of a customer at the Coffee Shop is $35 and the average meal time of a customer here is 20 minutes. Most of the customers take a fast meals here. There are always lots customers waiting for ordering at peak hours,such as in the morning, lunch time and in the evening, Many customers order only 1 item each time, and there are many requests for new order at the same time. Waiters are responsible for both ordering and sending plates, and we only have 5 full time staffs for each branches. Therefore, they are usually out of hand. One of the solution for this is to hire more staff, but it would be too expensive. Miss CHENG would like to have a mobile app for her coffee shop, so that customers can order food by themselves and then give them some discount. She hope that the discount can attract more customers to use the app to make the order.

**1.4 Major constraints**

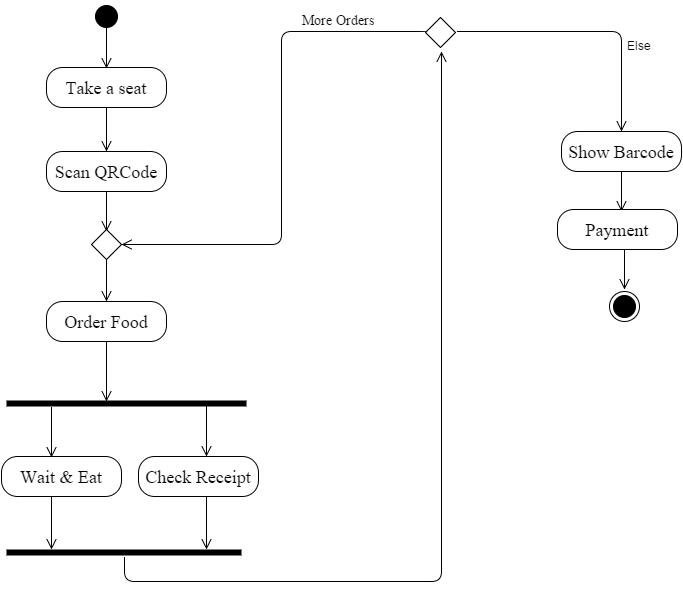
If we are going to develop the pay function of the app, we need to cooperate with banks. This also related to credit problems and additional development cost.

The app need to connect to the Internet. Bamboo Coffee Shop has already set Wifi within the shop. The Wifi ability is limited. The app should not require much data from the Internet or upload too much data to the Internet. For example, pictures should be saved in the app.

Some customers come to the Coffee shop and download the app. The size of the app small enough to be downloaded within 3 minutes. Since customer here usually takes 15 minutes to eat, they won’t spend too much time on downloading apps.

**2.0 Action Steps**

**2.1 Activity Diagram**



**2.2 Activity Description**

After customers take a seat in the Coffee Shop, they can make use of their mobile devices to enjoy the meals. They can scan the QR code to order the food on condition that they have to download the app first. The orders will be sent to the kitchen automatically through the system. Besides, customers can have more orders while they are waiting and eating. They can also check the receipts. If there is no ordering, they can settle the payment by showing the Bar code on the app to the staff. The payment is in form of cash exchanging instead of online payment.

**3.0 Usage scenario**

**3.1 User profiles**

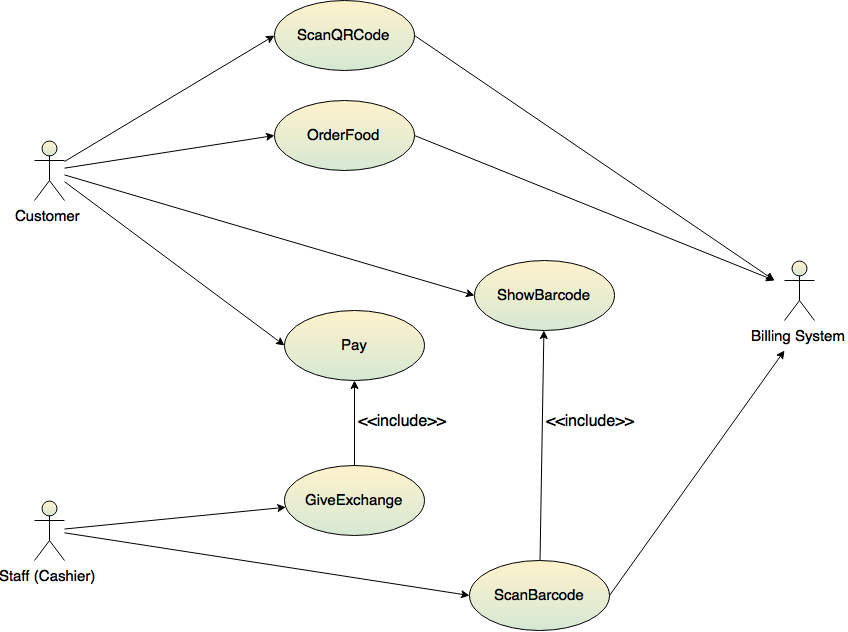
Customer of Bamboo Coffee Shop:

* They are normal citizens in the city or travellers. They just want to have a quick meal in the Coffee Shop. They also hope for some discount in the Coffee Shop. They don’t want to spend much time download the app. They don’t want to the app occupy too much space in their mobile as well. They have no interest to read help documentation for the app.

Staff of Bamboo Coffee Shop:

* They have no programming background but have basic computer manipulation. They will spend some time read help documentation for the app.

**3.2 Use-Case Diagram**



**3.3 Use-Case Descriptions**

**Use case includes:**

* Customer: Customer of Bamboo Coffee Shop.
* Staff: Staff of Bamboo Coffee Shop responsible for update data in the app.
* Billing System: Record what customers has ordered. Record whether customer has paid for the bill.

**Usage scenario (Fiona)**

Scenario: Customer makes an order

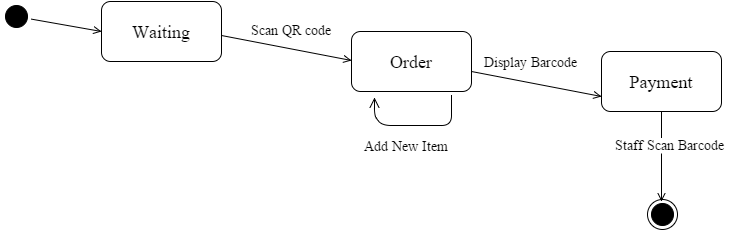
1. Customer downloads the mobile app for Bamboo Coffee Shop.
2. Customer scans QR Code on the table using the mobile app.
3. Billing System recognizes the table number.
4. Customer orders food using the mobile app.
5. Billing System receives and records the order.

Scenario: Customer settles the payment

1. Customer opens the mobile app.
2. Customer scans QR Code on the table using the mobile app.
3. Billing System recognizes the table number.
4. Customer clicks “Pay”.
5. System displays the receipt.
6. Customer checks the receipt, and Click “Pay”.
7. Customer shows the barcode on the mobile app to staff(cashier).
8. Cashier scans the barcode.
9. Billing System recognizes the table number and the order.
10. Customer pay the money.
11. Cashier gives exchange.

**4.0 Behavioural Model and Description**

**4.1State Diagram**



**4.2Description for software behaviour**

The application has 3 state: Waiting, Order and Payment.

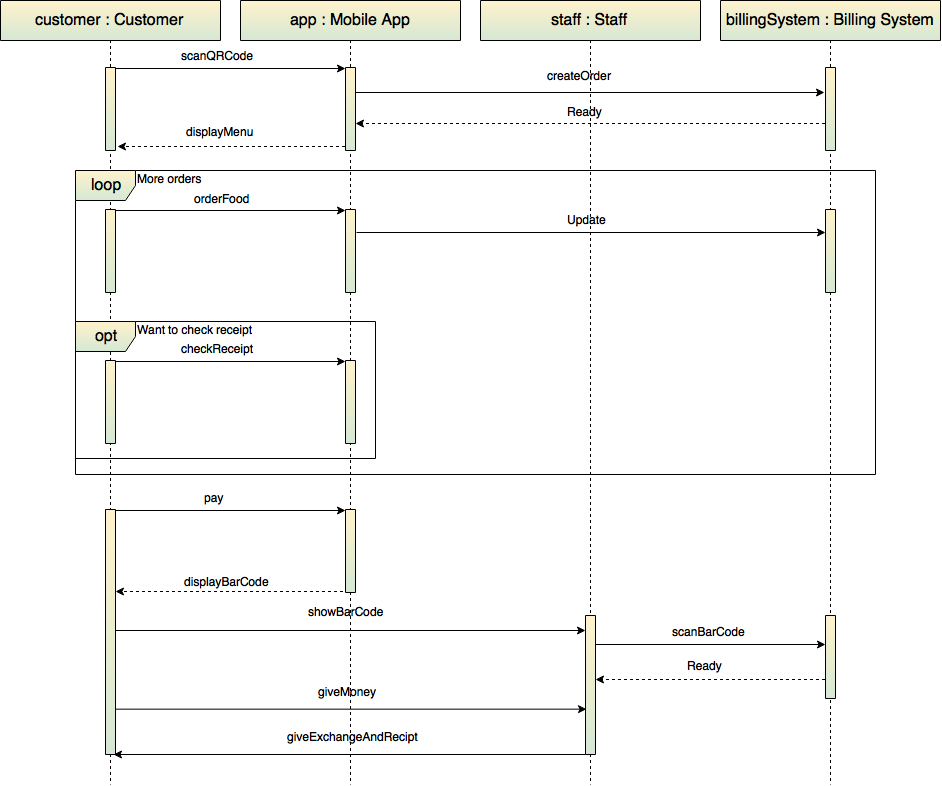
After user open the application, it comes into Waiting state. In Waiting state, application is waiting customer to take a seat and scan the QR code on the table. Once user scan QR code on the table, the application comes into the Order state.

In order state, user can use the application to order new food items, check what they have ordered and check the price of this meal. After finishing the meal, customer press Pay button and the application will display a barcode. Now the application come into the Payment state.

In Payment state, application is waiting for staff scanned the barcode. Customer will pay money for the meal and give the money to staff. After receiving money, staff scan the Barcode. Now the meal is get paid and the Payment state ends.

**5.0 Interaction between Objects**

**5.1 Sequence Diagram**



**5.2 Description of Sequence Diagram**

* Enter the Order Interface

First customer needs to enter the order interface. When user enter the restaurant, they will found 2 QR code on the table. One for download the app, the other for enter the order interface. Customer use the application to scan the QR code, the application will connected to bill system and set up a order record here. After successfully set up the record, it send back signals to the application. Application received the signal and turn to the order interface.

* Order items and eat

Now user comes to the main menu of the application, order Interface. In this interface, user can check what they have ordered, how much they should paid for the meal. They can also order food. Only when they confirmed what they want to order, application update their order to the database. Customer can order as many as times they want.

* Pay for the meal

After enjoying the meal, customer click Pay button. The application will show a bar code on the screen. Staff in this stage is involved. They receive money from customer, by cash or by octal card. They scan the barcode and signal will send to the bill system to end the order.

**6.0 Restrictions, Limitations, and Constraints**

1. Only one mobile device enables to make the order in each table
   1. Inconvenience
   2. Cannot share table
2. Cannot delete the ordered items
3. Wifi connection is necessary
4. Narrow target customer(people with mobile phone)
   1. The limitation is not severe. Most coffee shop customers have their own phone.
5. The size of the application should not over 10M so that user can download it quickly.
6. The response time of the application should not be bigger than 1s.

**7.0 Validation Criteria**

**7.1 Classes of tests**

1. Same item order for 20 orders
2. Make order request in an order for 200 times.
3. Same items through different devices, order for 100 times:
   1. through customer’s mobile device
   2. through staff’s mobile device.
4. Order over 20 items
5. Order no items
6. Order items over $1000
7. Order items less than $10
   1. Include 0$, customer didn’t order anything and leave

**7.2 Expected software response**

1. offer 5% off discount with right total price
2. offer no discount on customer’s mobile
3. accurate calculation
4. same items order results in same price

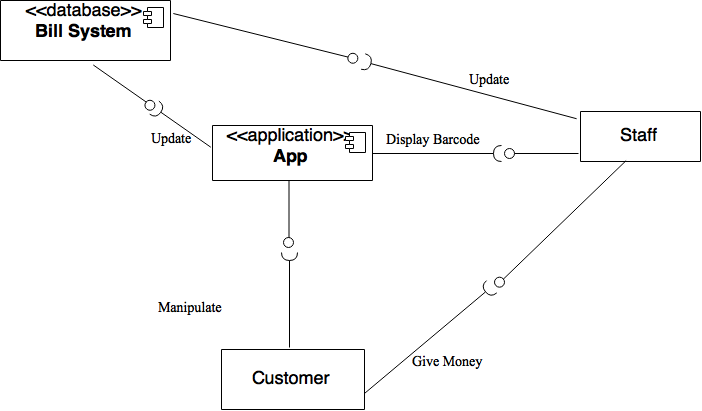
**7.3 Performance bounds**

1. After pressing "Confirm button"→ within 3 seconds to complete the operation
2. After pressing "Pay" button→ within 3 seconds to complete the operation
3. Check receipt→ 0.5 second to complete the operation
4. Success rate>99.9%

**Part II: Software Design Specification**

**1.0 Architectural and component-level design**

**1.1 System Architecture diagram**



**1.2 Description for Component n**

Customer:

* Customer of Bamboo Coffee Shop.
* They use mobile phone to manipulation App by tapping.
* They show bar code in the App and give money to the staff.

Staff:

* Staff of Bamboo Coffee Shop.
* They receive money from customer.
* They scan the barcode customer showed to them.
* After they scan the barcode, information update to Bill System

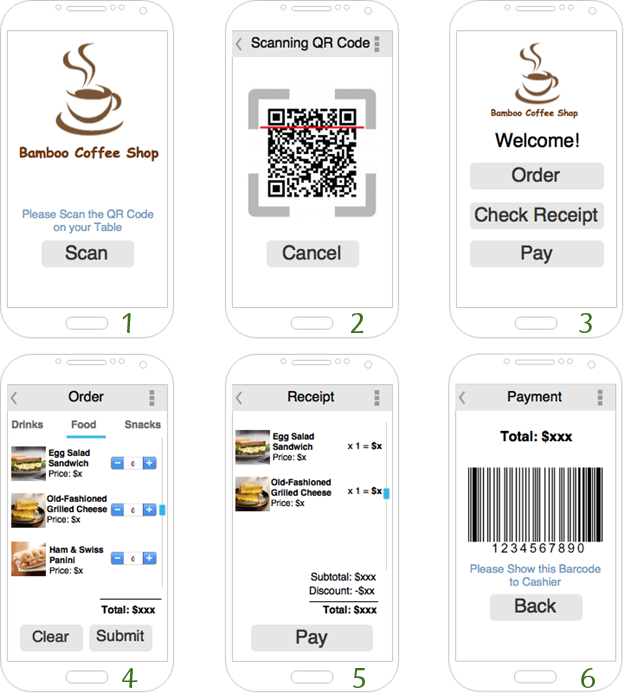
App:

* The application developed in this project.
* It receive button clicking from customers.
* It displays bar code to Staff.
* It updates the bill system when necessary.

Bill System:

* The database used for record order information.
* It receives update instruction from the app.
* It receives update instruction from the staff.

**2.0 User interface design 2.1 Screen images**



**2.2 Objects and actions**

**press which button will go which interface, what happened(Fiona)**

Interface 1: This is the first page that customer that see after they download and open the app. User tap the “Scan” button to scan the QR Code on their table in order to make an order or other request, and it will proceed to Interface 2.

Interface 2: This is the page for scanning QR Code, and the app would like to access user’s camera on their smart phone. The system can then recognize their table number, and it will proceed to Interface 3.

Interface 3: This is the page user can see after they have scanned the QR Code on their table. User can tap “Order” to make an order, or tap “Check receipt” to view what they have ordered and the total, or tap “Pay” to settle payment after they have finished their meal. It will proceed to Interface 4 and 5 correspondingly.

Interface 4: This is the page showing the menu that customer can order. The menu is categorized into “Drinks”, “Food”, and “Snacks”. User can view the name and price of each item in this page. They can click “+” to add item to their order, and click “-” to reduce the quantity. The “Clear” button is used to clear all inputs by the user. The total for this order is displayed at the bottom of the page. User can then tap “Submit” to send out this order.

Interface 5: This is the receipt page user can see after they tap “Check receipt” or “Pay” in Interface 3. Customer can view what they have ordered, total before discount, discount, and the total in this page. User can tap “Pay” to settle the payment, and it will proceed to Interface 6.

Interface 6: This is the payment page that customer need to show it to cashier when settling the payment.

**2.3 Interface design rules**

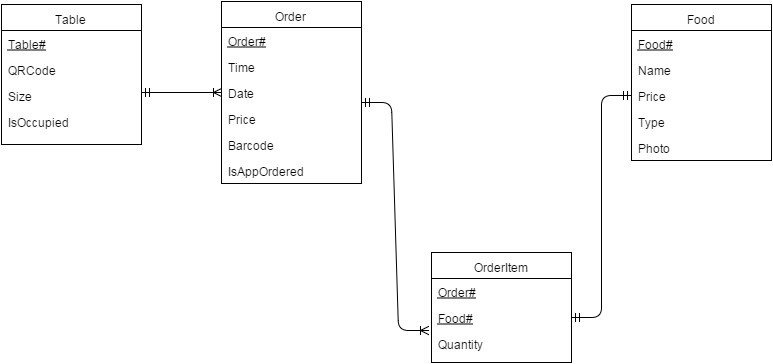
The interface should be simple and clear. It should be consistent to the style of Bamboo Coffee Restaurant.

**2.4 UIDS description**

* Android Studio is used to set the Interface.
* Microsoft Visio is used design the Interface.
* Photoshop is used to create the Image inside the Interface.

**3.0 Data Model and Description**

**3.1 Data objects (Entity-relationship diagram)**



There are four main entities in the diagram: Table, Order, Food and OrderItem.

Attributes of the Entity:

1. Table

* Table No. (primary key)
* the size of the table
* QR Code on each table to let customer to scan and start ordering through the app
* show that whether the table is occupied.

1. Order

* Order No.
* Time and date of each order
* Price of each receipt
* Bar code that let staff to scan to settle payment
* IsAppOrdered shows whether the order is made via app or traditional order

1. Food

* Food No.(primary key)
* Name
* Price
* Type
* Photo

1. OrderItem

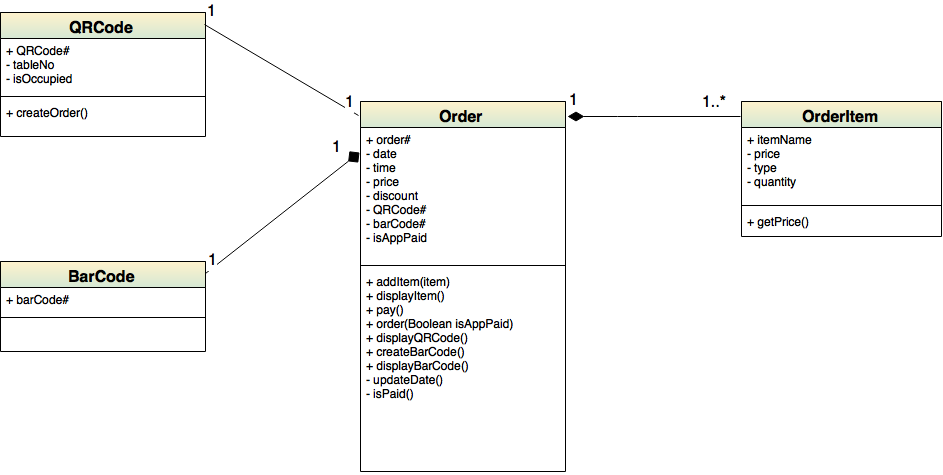
* Order No. (primary key)
* Food No.
* Quantity

**3.2 Relationships**

A table can create an order each time, which mean there is only one receipt in each table. In the whole operating day, several orders can be created by a table. We will link to Order to Food through the OrderItem to record the quantity.

**4.0 Functional Model and Description**

**4.1 Class diagrams**



**4.2 Software Interface Description**

If the order is initialized by the app, a QR Code will be created and displayed to start a new order. The app allows users to addItem and displayItem which can be connected to OrderItem, in which the price and other items information can be got.

During the payment, Bar Code will be created and displayed by the application. After the bill is paid, the information will be updated to the database immediately.

**5.0 Conclusion**

In conclusion, the app not only enables to increase the efficiency of staff but also brings out a small discount to customers to enjoy the meals. By alleviating the pressure of the staff, better services can be provided to the customers. Also, it can solve the overcrowded and inefficient problems to boost the patronage, especially the peak hour. It is hoping that the app can enhance our shop’s competitiveness among the competitors and build a good reputation.