**Mooli Ordering System**

**Software Requirement Specification (SRS)**

**April 8th, 2020**

**Prepared by Group X**

**Computer Science and Technology Program**

**United International College**

1. Table of Contents

[1. Table of Contents 2](#_Toc41265931)

[2. Introduction 3](#_Toc41265932)

[2.1. Purpose 3](#_Toc41265933)

[2.2. Document Conventions 3](#_Toc41265934)

[3. System Features 4](#_Toc41265935)

[3.1. Staff login / logout 4](#_Toc41265936)

[3.2. Permission system 4](#_Toc41265937)

[3.3. On-going orders management 4](#_Toc41265938)

[3.4. History orders management 5](#_Toc41265939)

[3.5. Product management 5](#_Toc41265940)

[3.6. Category management 5](#_Toc41265941)

[3.7. Coupons management 6](#_Toc41265942)

[3.8. Redeem card management 6](#_Toc41265943)

[3.9. Staffs management 6](#_Toc41265944)

[3.10. Changing password (staff end) 6](#_Toc41265945)

[4. External Interface Requirements 8](#_Toc41265946)

[4.1. User Interfaces 8](#_Toc41265947)

[4.2. Hardware Interfaces 8](#_Toc41265948)

[4.3. Communications Interfaces 8](#_Toc41265949)

[5. Other Nonfunctional Requirements 9](#_Toc41265950)

[5.1. Performance Requirements 9](#_Toc41265951)

[5.2. Security Requirements 9](#_Toc41265952)

[5.3. Software Quality Attributes 9](#_Toc41265953)

1. Introduction

## Purpose

The software requirement specification (SRS) includes the functional requirements (in chapter 3), external interface requirements (chapter 4), and nonfunctional requirements of the system (chapter 5).

## Document Conventions

<Describe any standards or typographical conventions that were followed when writing this SRS, such as fonts or highlighting that have special significance. For example, state whether priorities for higher-level requirements are assumed to be inherited by detailed requirements, or whether every requirement statement is to have its priority. Features highlighted in yellow means the features are in high priority than others. The feature must be thoroughly tested before the software is put in production mode.

1. System Features

## Staff login / logout

3.1.1 Description

The staff should be able to log in and out of the staff's end.

3.1.2 Functional Requirements

REQ-1: Only the staff should be able to log in. Regular customers aren't allowed.

REQ-2: The staff must provide a username and a correct password in order to log in.

REQ-3: When login failed, a message must be shown on the page to indicate the error.

REQ-4: The staff can log out of the system after logging in.

## Permission system

3.1.1 Description

Features in the staff end are only accessible according to the staff's role.

3.1.2 Functional Requirements

REQ-1: A role named "Superuser" should have all access to all features.

REQ-2: The role "cook" only has access to the orders.

REQ-3: The role "product manager" only has access to "products" and "categories."

REQ-4: The role "financial manager" only has access to "coupons" and "redeem cards."

REQ-5: The role "shop manager" has access to all features except "staff."

REQ-6: If the staff tries to access a feature that he or she doesn't have permission to, an error message indicating that the staff has no permission to the feature should be given.

## On-going orders management

3.1.1 Description

Manage all on-going orders.

3.1.2 Functional Requirements

REQ-1: When a new order is placed, the order and the products chosen should show up automatically when the staff is on the page.

REQ-2: All new orders should be in status "pending."

REQ-3: The cook should be able to cancel the order.

REQ-4: The cook should be able to confirm the order (from pending to processing).

REQ-5: The cook should be able to update the status of the processing orders to "ready for pickup" when the order is ready.

REQ-6: After the customer has picked up the orders, the order should be removed from the screen.

REQ-7: The longest-waiting order should come first.

REQ-8: Cancelled orders should not be refunded automatically.

## History orders management

3.1.1 Description

Manage all on-going and finished orders.

3.1.2 Functional Requirements

REQ-1: The page should contain information including the id of the order, the name (in most cases, the customer won't provide a real name, so it is likely that only the username is shown on the screen) and username of the customer, the total, the actual paid, the time the order was created, and the status of the order.

REQ-2: The staff should be able to view the detailed information of a given order. The detailed information should contain all the given information in REQ-1, and the products the customer bought, the amount of each product, and the amount the order was discounted.

REQ-3: The staff should be able to refund a customer the entire order.

REQ-4: The refund can only take place for once for every order.

## Product management

3.1.1 Description

Manage products

3.1.2 Functional Requirements

REQ-1: Every product should have a name, a price, a priority, and one or more categories that the product belongs to.

REQ-2: The page should contain the product name, the price at first glance.

REQ-3: The staff should be able to visit products of a specific category or all the products.

REQ-4: Each product can belong to multiple categories.

REQ-5: Staff can create a product in the panel. The information the authorized staff should provide is product name, description, the price, the priority the product should display, and one or more categories that it belongs to.

REQ-6: Staff should be able to update the information described in REQ-5.

REQ-7: Staff can upload picture and thumbnail of the product.

REQ-8: Staff can delete products.

REQ-9: Each product should have a unique name.

## Category management

3.1.1 Description

Manage categories

3.1.2 Functional Requirements

REQ-1: Every category should have a name and a priority.

REQ-2: Staff should be able to edit the name and the priority of a category.

REQ-3: Staff should be able to delete categories.

REQ-4: Each category must have a unique name.

## Coupons management

3.1.1 Description

Manage coupons, which are often used in promotions.

3.1.2 Functional Requirements

REQ-1: The page should contain all the coupons' information.

REQ-2: A coupon contains a coupon code, which user can enter to be eligible for the sale, a value the coupon worth, the threshold (minimum value the user needs to purchase before he or she can be eligible for the coupon code), the time the coupon code activates, and the time the coupon code expires.

REQ-3: Staffs should be able to edit the coupon codes and its related information.

REQ-4: Staffs can delete the coupon codes.

REQ-5: Each coupon code should be unique.

## Redeem card management

3.1.1 Description

Manage redeem cards (gift cards).

3.1.2 Functional Requirements

REQ-1: The page should show all redeem cards available in the system.

REQ-2: Every redeem card has a value and the redeem code.

REQ-3: Staffs can generate redeem codes in batch.

REQ-4: Staffs can delete the redeem cards.

REQ-5: Redeem card codes should be randomly generated to ensure safety.

## Staffs management

3.1.1 Description

Manage staffs

3.1.2 Functional Requirements

REQ-1: The page should show all staff.

REQ-2: Authorized staff can add a staff to the system.

REQ-3: To add staff, the administrator must provide the username, email, password, and the role of the staff. Optional information includes first name, last name, gender, and phone number.

REQ-4: The administrator should be able to update the role, first name, last name, gender, and phone number of staff.

REQ-5: The administrator should be able to remove staff from the staff list. But the removed staff can still log in as a normal customer in the customer end.

## Changing password (staff end)

3.1.1 Description

Change the password of the logged-in staff in the staff end.

3.1.2 Functional Requirements

REQ-1: The staff must provide the old password, the new password twice in order to be eligible for the update.

REQ-2: Error message should be given when the update failed.

## Product display (customer)

3.1.1 Description

Customers can view the product on the home page of the customer end. The customer can click on a specific item to see the detailed information of a product.

3.1.2 Functional Requirements

REQ-1: In the home screen, the user should be able to see the products according to its category and listed in descending order according to the products' priority.

REQ-2: The user can click on a specific product to view the detailed information of the product.

## Authentication (customer)

3.1.1 Description

The customer should be able to register an account, login to the account, and logout of the account.

3.1.2 Functional Requirements

REQ-1: The user should be able to log in to an existing account. The user should provide a username and the account's password to log in.

REQ-2: Customers can sign up for a new account by providing his or her preferred username, email address, and password.

REQ-3: After logging in, the user should be able to logout of the system.

REQ-4: Personalized operation, including accessing the "cart" and "me" page, requires the customer to be logged in.

## Cart (customer)

3.6.1 Description

The user can choose the product he/she would like to purchase and add it to his/her cart.

3.6.2 Functional Requirements

The user can go to the cart from the navigation bar. To access this page, the user must log in first. If the user is not logged in, opening the cart will result in the user being redirect to the user to the login page.

REQ-1: The user must be logged in before accessing the cart.

REQ-2: The user must be logged in before he or she can add items to the cart.

REQ-3: The user can adjust the number of items in the cart.

REQ-4: A grand total should be updated instantly after the user has decided to adjust the amount.

REQ-5: The user should be able to delete the items in his or her cart.

## Placing an order

3.6.1 Description

The user should be able to place an order on all the cart items the customers have in the customer's cart.

3.6.2 Functional Requirements

REQ-1: The user must be logged in before placing an order.

REQ-2: The user should be asked whether he or she has a coupon code, in which the customer could receive a discount from the coupon code.

REQ-3: The customer should be able to skip the process if the customer does not have a coupon code.

REQ-4: If the coupon code is valid, the user should be able to know that the bill has been given a discount before paying.

REQ-5: If the user doesn't have enough balance in his or her balance, the payment should be rejected.

## Updating personal information

3.6.1 Description

A customer should be able to update his or her personal information, including the name (first name and last name), gender, and phone number, and etc.

3.6.2 Functional Requirements

REQ-1: The user should be able to able to update his or her basic personal

Information including first name, last name, gender, and phone number.

REQ-2: The customer should be able to update his or her avatar.

REQ-3: The customer should be able to change his password.

REQ-4: Changing the password requires the customer to enter the old password and the new passwords twice.

## Redeem from a gift card

3.6.1 Description

The customer should be able to redeem from gift cards.

3.6.2 Functional Requirements

REQ-1: The user must be logged in before he or she can redeem it from a gift card.

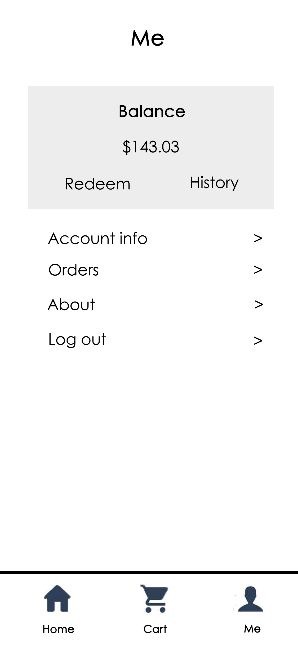
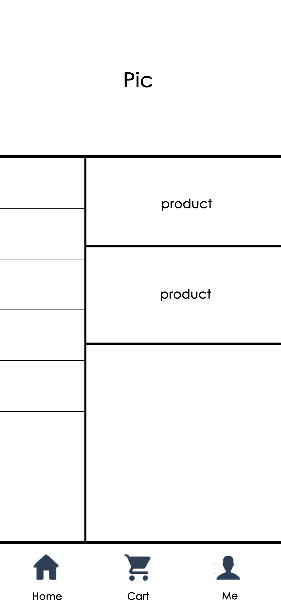
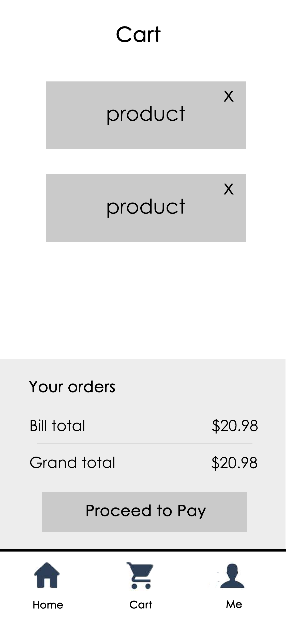
REQ-2: The user must provide a correct redeem card (gift card) code that the shop

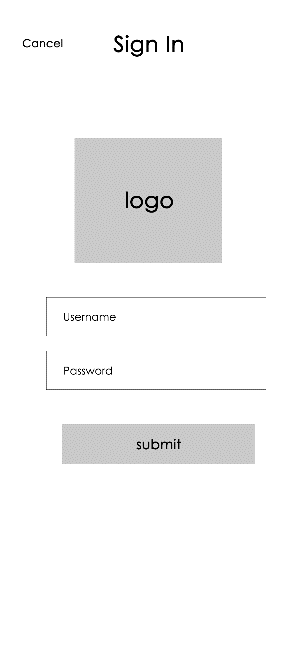
provides in order to redeem.

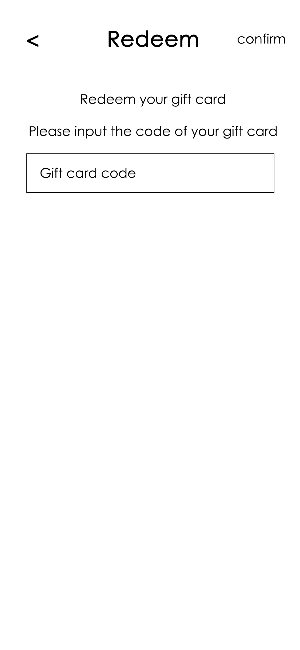
1. External Interface Requirements

## User Interfaces

The design of the customer end should follow the general UI pattern of modern smartphone. Most operations are done using a single click. There should not be the need to dive into a foldable menu. Here are some of the concepts for the design.

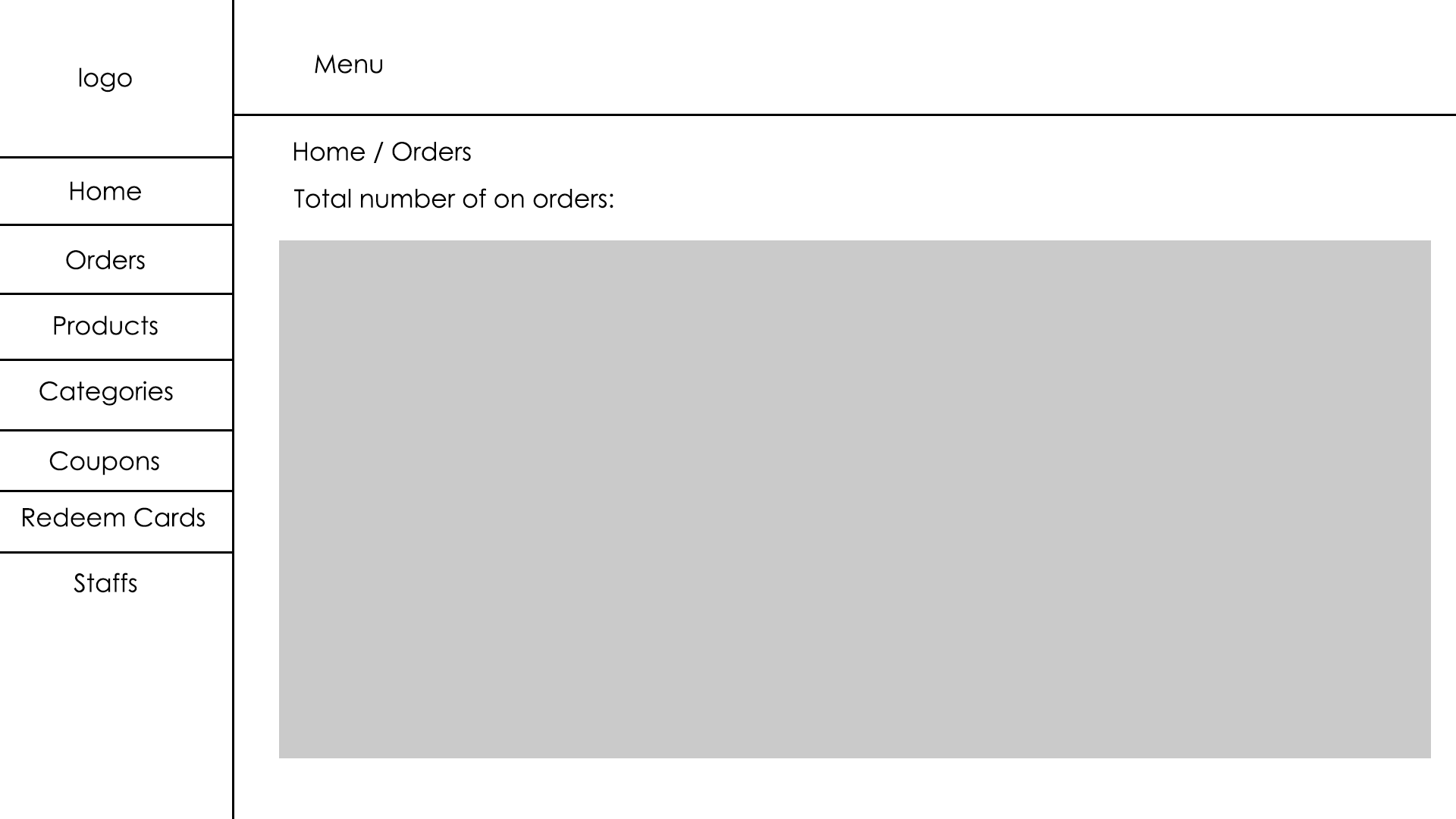






Since the customer end are designed to be running on a smartphone, error messages will pop directly using the system dialogue system.

The staff end of the software is designed to be running tablets, laptops, and desktop PCs. The design of the staff end will follow the general guideline of most management software. Messages will be shown in a message box within the webpage (also known as a modal). The following are two examples.





## Hardware Interfaces

The customers can access the software via a modern smartphone. The compatibility of the UI will be checked on apple devices starting from iPhone 6 to iPhone X. Since apple's iPhone series represents the overall trend of screen sizes of modern smartphones. The test should be meaningful to 95% of smartphones currently running.

The staffs can access the software via tablets (like iPads), laptops (like a MacBook), and desktop PCs (like an iMac).

## Communications Interfaces

The customers and staff can access the web via HTTP. However, due to the lack of encryption and poor security of HTTP, the application can be configured to transfer network traffic via HTTPS when TLS is appropriately configured on the server.

1. Other Nonfunctional Requirements

## Performance Requirements

The system takes less than 1 second to insert a record to any table and less than 2 seconds to delete or search a record (from the software to database).

## Security Requirements

The system should allow the user to configure TLS certificates on their own to secure all network traffic between the server and the users.

The system should avoid malicious attacks from hackers in the name of logged-in users.

## Software Quality Attributes

**Availability**: the system should be able to encapsulate all errors and notify the user when appropriate. In the case of a system crash, the system should be able to restart automatically and log the errors.

**Interoperability**: the system should be able to work when there are concurrent incoming connections and no inconsistency should be created.

**Reliability**: the system should guarantee that either all transactions complete or all failed, never leaving transactions in the middle of nowhere.

**Usability**: the system should be easy for the average users to use it without additional training.

Ethos: the system should be ethical. For example, all users' passwords should be encrypted so that no one, including the database administrator, should be able to see the passwords shown in plain text.