Software Requirement Specification

Group Number 6

Group Name:

Harmony

Group Members:

Syed Zakaria Mehmood (202180090138)

Shahriar Sarlak (202180090149)

Ali Nobakhtnamini (202180090107)

TABLE OF CONTENTS

1. Introduction

1.1 Purpose

- The purpose of this Software Requirements Specification (SRS) document is to provide a detailed description of the requirements for users to make the order and delivery of food more efficient.
- This application aims to enhance the dining experience for students by providing a convenient and efficient way to order and manage their food orders

1.2 Project Scope

- The purpose of the proposed model is that it will permit the users inside university make the orders and delivery of the food more convenient.
- The ordering system would provide the employees, students, staff and other targeted audience to order food efficiently inside the canteen hall.
- It will also allow them to order meals online from school's canteen to be delivered to a specified campus location.

2. Overall Description

2.1 Product Perspective

- This version is a newly invented software system which replaces the current manual processes for ordering food while eating in canteen or delivering food online from canteen application.

2.2 Special Terminologies & Notations

- SRS: Software Requirements Specification
- API: Application Programming Interface

2.3 Operating Environment

- Windows: Google chrome (all versions), Windows Internet explorer, Firefox, Edge

- Mobile Version: Mobile version would be compatible with android version 7.0 and above and iOS version 11 and above.

2.4 Targeted Audience

- The targeted audience to use the Software would be University students, staff, teachers, employees.

2.5 Actors and User Classes

- User: A student, a teacher, an employee or anyone who wants to eat food
- Canteen Staff
- Restaurant manager: The person responsible for organizing inventory products and upgrade menus on regular basis.
- Customer service advisor
- Administrator of the system: Staff that would have access to all restaurants and their menus.
- Meal deliverer: The person to deliver ordered/pre ordered food to specific destination inside campus

Use Cases

1. Login / Continue as guest:
   Users should be able to add their information such as phone number and login to the system or they can enter the system as a guest.

2. Order Meal:
   The menu of foods and restaurants will be shown, and users will be able to choose their desired meal by selecting each item.

3. Confirmation Page:
   After choosing the foods and items, there should be a confirmation page which will show the chosen items and prices and the other details of the order such as restaurant information. At the below of the page, there should be a confirm button.

4. Cancel Order:
   Both user and admin of system should have access to cancelation. There should be two situations for cancelation of order. The first one is canceling before confirming the order which can be done easily by user. The second one is cancelation after confirming the

order. At this point user needs to confirm the cancelation with the admin, if the food has not been prepared yet, it can be canceled.

5. Payment:
   Both user and admin of system should have access to payment page. In this page, the user will be asked to select a payment method to do the payment. The system should provide various methods such as Wechat/Alipay/Student card and etc. The system will securely process the payment and update the order status.

6. Online Support:
   The user should be able to access online support during the whole process of ordering. Meanwhile the admin and AI of the page are responsible to help and support the user's issues related to their orders. The system will provide a complete helpline or chat support for prompt resolution of customer concerns.

7. Receipt:
   At the end of the process, the receipt will be shown to the user which include the all information of the order, such as all ordered items and prices with the time and addresses. This receipt will be also sent to the restaurant which is responsible for preparing the foods.

8. Delivery:
   This use case is for delivery personnel who use the food ordering system to receive and fulfill orders and deliver it to the user's location. Meanwhile, the user and restaurant should have access to delivery person location and information.

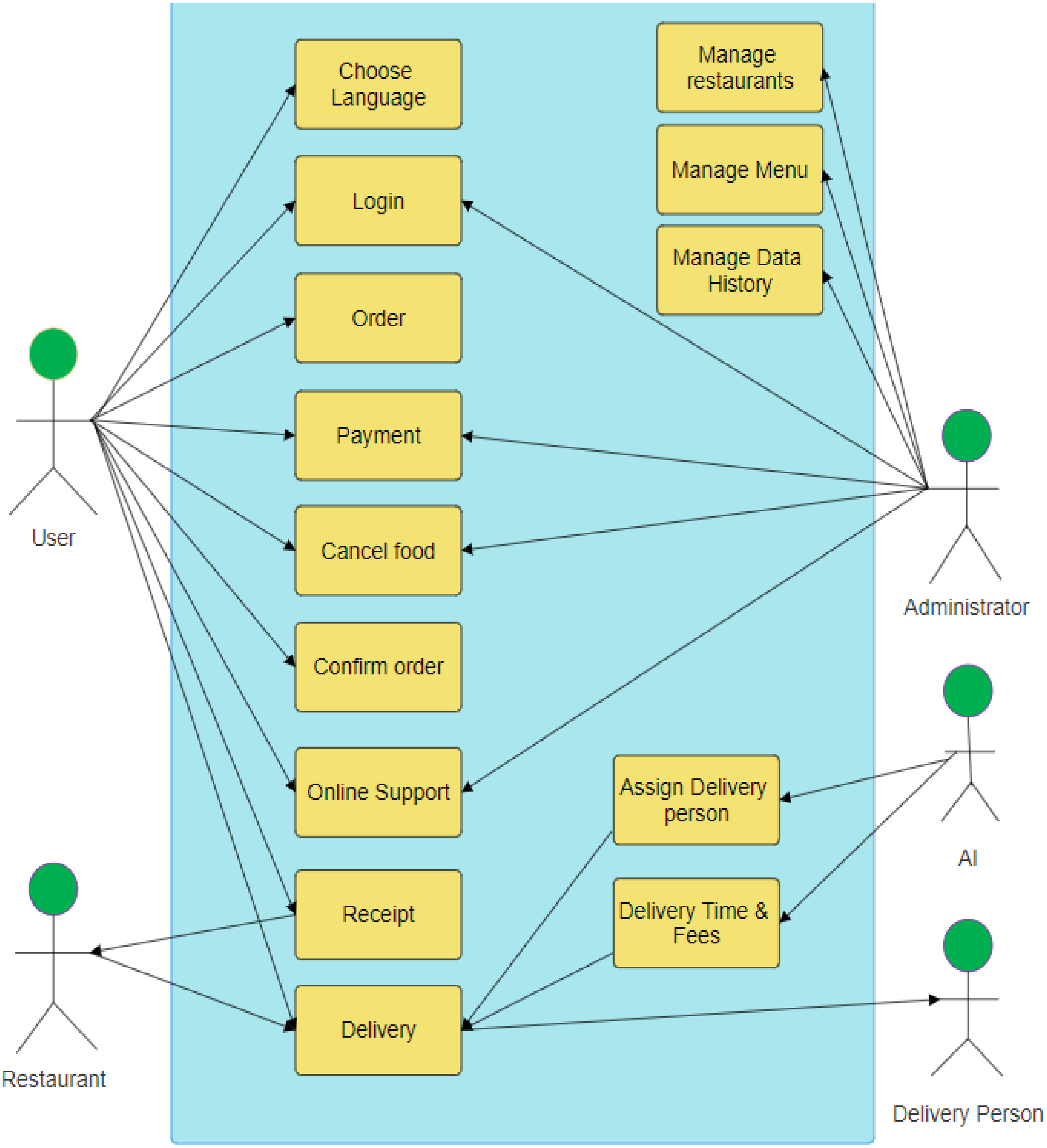
9. Special Order:
   Special order should be designed for the users who need to be guided for choosing the most appropriate food for themselves. It will be a page which will get some information by asking some questions from the user to provide and suggest the best options.

10. Manage Restaurants and Menu:
    This use case is for the administrator of the system to have access to all pages of restaurants and their menus, which includes tasks such as updating the menu, managing orders, tracking deliveries, adding or deleting items, etc.

11. Manage Data History:

This use case will be for administrator of the system to have access to all orders histories with all the information related to them.

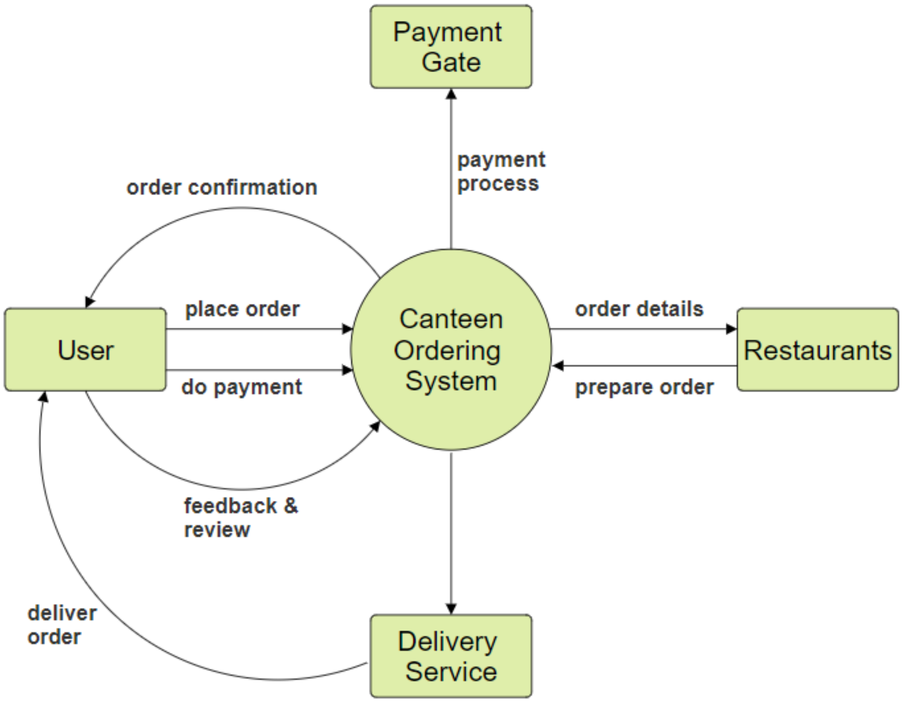
12. Assign Delivery Person, Fees and Time:
    This use case is for the AI of the system which will be able to choose the delivery person and calculate the fees and time of the orders to be shown to users.



Use Cases Diagram

2.6 Design and Implementation

- The system will use the standard Oracle Database that the company currently uses.
- All HTML code shall meet the HTML 5.0 standard.



Data Flow Diagram level 0 for COS

2.7 Assumption and Dependencies

- Canteen is open for breakfast, lunch, supper, dinner throughout the 7 days of the week with staff expected to be on duties.
- The operation of the system depends on the changes being made in inventory to update the availability of food.

3. System Features

3.1 User Registration and Login:

Description:

This feature allows users to create an account and log in to the food ordering system or they can easily enter as a guest.

Functional Requirements:

- Users should be able to register with their personal information, such as name, address, and contact details.
- Users should be able to log in using their registered credentials.
- The system should verify the user's identity and provide access to their account.
- The system should be bind to mobile number or Wechat of each user.

3.2 Language Selection:

Description:

This feature asks users to choose a language as they desire at the first page.

Functional Requirements:

- System should provide English and Chinese language as the main languages on the top.
- Users should be able to select their desired language by clicking on other languages button.
- After choosing the language, all the system interfaces should be turned to that language.

3.3 QR-codes:

Description:

QR codes should be available at different places around the canteen to provide a convenient way for users to use and also it will be used for payment methods also.

Functional requirement:

- User should be able to see the canteen menu through scanning QR codes.
- QR-codes are also generated for payment purposes.
- User should be able to enter the first page which will be asked to select a language.

3.4 Menu Display:

Description:

This feature displays all the food items and their details for users to check, browse and select, while providing the name of the foods and restaurants and the ratings and prices.

Functional Requirements:

- The system should categorize food items into different sections, such as restaurants, main foods, best prices foods, best suggestions, drinks, desserts and etc.
- The system should display images, ratings, descriptions, and prices of each food item.
- Users should be able to filter and search for specific food items or restaurant.
- There should be a button for special order on the main menu.
- Users should be able to see the top-rated restaurants and hot sales foods at the main page.
- After selecting each food, user should be able to see the name of the materials which will be used in the food to check.
- In each restaurant menu, there should be different sections for different types of foods and drinks.

3.5 Cart Management:

Description:

This feature allows users to be able to customize their cart for the purposes of adding and removing items or canceling the order. In this page, all the selected foods and items with their quantity and prices should be visible for user.

Functional Requirements:

- Users shall be able to add items to their cart and specify quantities.
- Users shall be able to add special instructions or dietary restrictions for each item.
- Users shall be able to view and modify the contents of their cart before confirming an order.
- Users shall be able to remove items or cancel the foods in their cart.

3.6 Order Placement:

Description:

This feature allows users to place their food orders and specify delivery or pickup options.

Functional Requirements:

- Users should be able to add selected food items to their cart.
- Users should be able to specify delivery address or pickup location.
- The system should calculate the total cost of the order including taxes and delivery charges.

3.7 Food Delivery:

Description:

This feature allows users within the area of university to enjoy the feature of food delivery.

Functional Requirements:

- Before the payment page, user will be asked to eat-in or want delivery.
- The delivery information will be provided for user to check the process of the food preparation.
- An estimated time should be provided for user on the delivery page.

- User should be able to see and check the location of food.
- A delivery duration should be conducted to be ready as soon as possible and send out.

3.8 Pre-order:

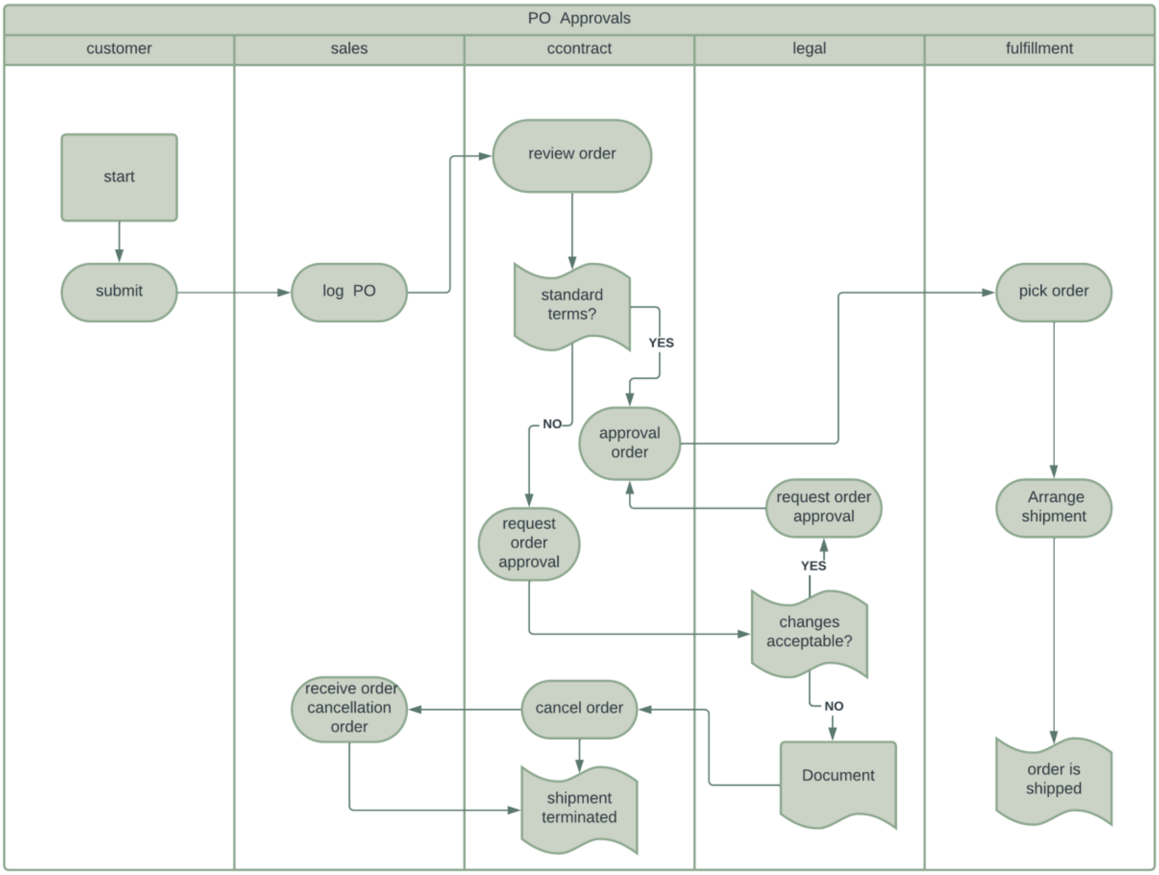
Description:

This feature allows user to place a Pre-order if they want to pick up food from canteen later by selecting pickup timings.

Functional Requirements:

- User should be able to Pre-order after logging into application.
- The Pre-order button should be provided at the first page of application, so it will be visible for all users.
- After selecting the Pre-order button, the user should follow the normal process of ordering foods.
- The only difference will be the Pre-order page which will contain different timetable to be selected by the user.
- System should provide different time options for pickup corresponding to the canteens availability during that moment.
- System should provide a digital receipt with full information of restaurant name, shop address, and order number to the user.
- Users can only Pre-order before the 10am for lunch and also before the 4pm for dinner
- After the deadline, timetable for the following meal should be closed.
- Pre-order process should follow an approval steps to be accepted in the system.

The PO approval takes the process of the below table



Swimlane Diagram for pre-order 1

3.9 Membership:

Description:

This feature provides user a lot of great offers and deals and user can buy discounted deals exclusively by purchasing membership for canteen.

Functional Requirements:

- Download canteen application
- The user should log-in to the system for accessing discounted deals.

- The system should provide user with different types of membership packages to purchase.
- The system should provide the user with discounted deals.

3.10 Phone Navigation Access:

Description:

User should allow access to phone navigation while using Food Delivery and Pre-Order methods.

Functional Requirements:

- The system should access the user’s phone navigation.
- Check whether user’s current location is within range (Inside University).
- When not in range user should be unable to proceed payment.
- User should be able to navigate the location by their phones to check the location of the food.

3.11 Payment Processing:

Description:

This feature should facilitate secure payment processing for the placed orders.

Functional Requirements:

- Users should be able to choose from various payment methods, such as Alipay, Wechat, and also student card.
- The system should securely process the payment and provide a confirmation of the transaction.
- System should provide a QR code for the payment for the ordering machines inside the canteen.

3.12 Order Tracking:

Description:

This feature allows users to track the status of their placed orders.

Functional Requirements:

- Users should receive real-time updates on the status of their order, such as order confirmation, preparation, and delivery.
- Users should be able to view the estimated delivery time and track the delivery person's location if applicable.
- User should be able to contact the restaurant and the support.

3.13 Customer Feedback and Reviews:

Description:

This feature allows users to provide feedback and reviews for the food and services received.

Functional Requirements:

- Users should be able to rate and write reviews for the food items and overall experience.
- The system should display the average ratings and reviews for each food item and restaurant.
- After the food is delivered, the user will be asked to rate the food and delivery services.
- The user should be able to see the reviews of other users.

3.14 Order History & Favorites

Description:

This feature allows the user to place order based on the previous order history and marked favorite meals.

Functional Requirements:

- The system should store a record of all past orders for each user.
- Users can view their order history and details such as date, time, items ordered, and total amount.
- They can also mark certain orders as "favorites" for easy reordering.

3.15 Admin Dashboard:

Description:

This feature provides a dashboard for restaurant owners or administrators to manage orders and menu items.

Functional Requirements:

- Admins should be able to view and manage incoming orders, update order status, and communicate with customers if needed.
- Admins should be able to add, edit, or remove menu items and update their details.
- Admins should be able access to data history of orders and all the information regarding to the users and restaurants.

3.16 Loyalty Program

Description:

This feature rewards customers for their loyalty by offering discounts, free items, or other incentives. It encourages customers to keep ordering from the same restaurant.

Functional Requirements:

- The system should have a loyalty program in place with different tiers or levels.
- Customers can earn points for each order they place.
- They can redeem their points for discounts or free items.
- The system should keep track of each customer's points and update them accordingly.

3.17 Customer Support

Description:

This feature provides customer support to users in case of any queries or issues. It includes a chatbot, FAQs, and a contact form for users to reach out to the support team.

Functional Requirements:

- The system should have a chatbot that can answer common queries of users.
- It should also have a list of frequently asked questions with answers.
- Users can fill out a contact form to submit their queries or issues.
- The support team should have access to the contact form submissions and respond to them promptly.
- Users should be able to access Supports in all the steps of ordering.

3.18 Dietary Restrictions and Allergen Information

Description:

This feature provides information about the ingredients and allergens present in each menu item. It helps customers with dietary restrictions or allergies to make informed choices.

Functional Requirements:

- The system should display the list of ingredients for each menu item.
- It should also highlight any common allergens present in the dish.
- Users should be able to filter menu items based on their dietary restrictions or allergies.

3.19 Advanced Search and Filter Options

Description:

This feature allows users to search for specific menu items or filter them based on their preferences. It helps them find their desired dish or restaurant quickly and easily.

Functional Requirements:

- The system should have a search bar where users can enter keywords or dish names.
- Users can also filter menu items based on categories such as cuisine type, dietary restrictions, price range, etc.
- The system should display the results based on the user's search or filters.

3.20 Table Reservation

Description:

This feature allows users to reserve a table at the restaurant for dine-in. It helps to avoid long waiting times and ensures a smooth dining experience.

Functional Requirements:

- The system should display the availability of tables at the restaurant.
- Users can select the date, time, and number of people for their reservation.
- The system should send a confirmation email or notification to the user and the restaurant.

3.21 Special Order

Description:

This feature will provide best suggestions for the user, based on their taste in the foods, by asking some questions.

Functional Requirements:

- This feature should be visible to the users at the first page of the menu.
- By clicking on this button, user should be able to enter to a new page called special orders.
- In the special orders page, user will be asked some questions about the food they want to have, such as which flavor do they prefer, what type of food do they consider (burgers, noodles, rice, sea food, and etc)
- After answering all questions, the system will provide a filter on the menus and restaurants to provide the best choices for the user.
- All the process after the special order page should be the same as before.

4. Non-Functional Requirements/Quality Attributes

4.1 Performance

- The response time of the system to user actions (e.g., menu browsing, order placement) should be within 2 seconds under normal load conditions.

- The system should handle a minimum of 500 simultaneous transactions during peak hours without degrading performance.

4.2 Scalability

- Design the system to accommodate a 20% growth in users and menu items over the next 3 years without a noticeable drop in performance.

4.3 Reliability

- The system should have a mean time between failures (MTBF) of at least 2000 hours, ensuring a reliable and consistent user experience.

- Scheduled maintenance should occur during non-peak hours, and users must be notified at least 48 hours in advance.

4.4 Availability

- The system should have an uptime of 99.9%, allowing for maintenance windows of no more than 1% of total operating hours annually.

- A redundant server infrastructure should be in place to minimize downtime in case of hardware failures.

4.5 Security

- User authentication should use industry-standard encryption algorithms, and passwords must be stored securely using salted hashes.

- Payment transactions must comply with Payment Industry Data Security Standard Requirement.

4.6 User Authentication

- Multi-factor authentication (MFA) should be implemented to enhance the security of user accounts.

4.7 Data backup and Recovery

- Daily automated backups of the entire system database should be performed, and backup integrity should be regularly verified.

- A disaster recovery plan should be in place, including procedures for restoring the system to its normal operation in the event of data loss.

4.8 Usability

- The user interface should adhere to accessibility standards (e.g., WCAG) to ensure usability for individuals with disabilities.

- Conduct user testing to ensure that 90% of users can place an order without assistance after minimal training.

4.9 Audit Trails

- Keep detailed logs of user actions, system events and errors for atleast 90 days.
- Ensure that audit logs are secure and cannot be accessed or altered by unauthorized users.

4.10 Compliance

- Conduct regular audits to make sure the system complies with data protection laws, university policies, and industry standards.

4.11 Integration

- The system should seamlessly integrate with the university's student database for accurate user authentication and billing information.

4.12 Customer Support

- The customer support team should respond to support tickets within 24 hours and resolve critical issues within 48 hours.

4.13 Documentation

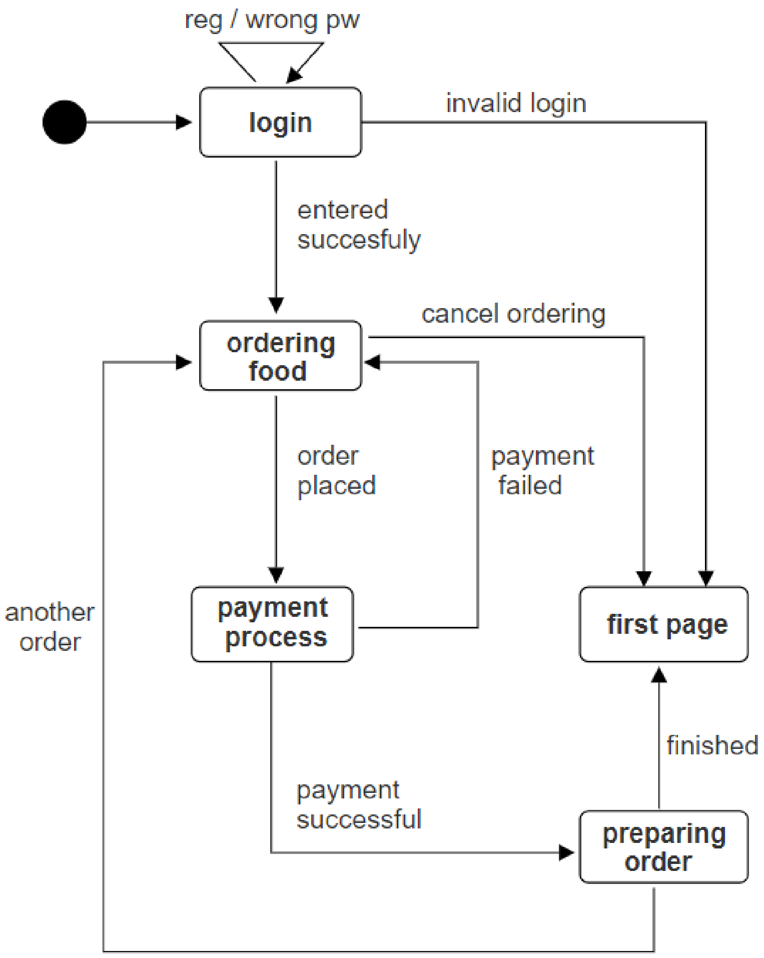
- Provide comprehensive documentation including user manuals, system architecture, and troubleshooting guides for administrators, end-users, and support staff.

5. Software Interface Requirements

5.1 External Interface Requirements

5.1.1 User Interfaces

- The user interface should be intuitive, visually appealing, and responsive on both Android and IOS devices.
- It should have adherence to modern design principles and provide a seamless user experience.
- Basic and easy to use interface as a normal food delivery application.
- Create, modify, view canteen menus.
- Availability of return option on each interface.



State Transition Diagram for COS

5.1.2 Hardware Interfaces

- The system machine will utilize the hardware components of smartphones, such as payments and the camera for scanning QR codes for viewing canteen’s menu and payment processing.

5.1.3 Software Interfaces

- The system will interact with the canteen’s backend systems through APIs to retrieve menu information, process orders, and update order statuses.

5.1.4 Communication Interfaces

- The system will require an internet connection to communicate with the canteen's servers and process payments securely.

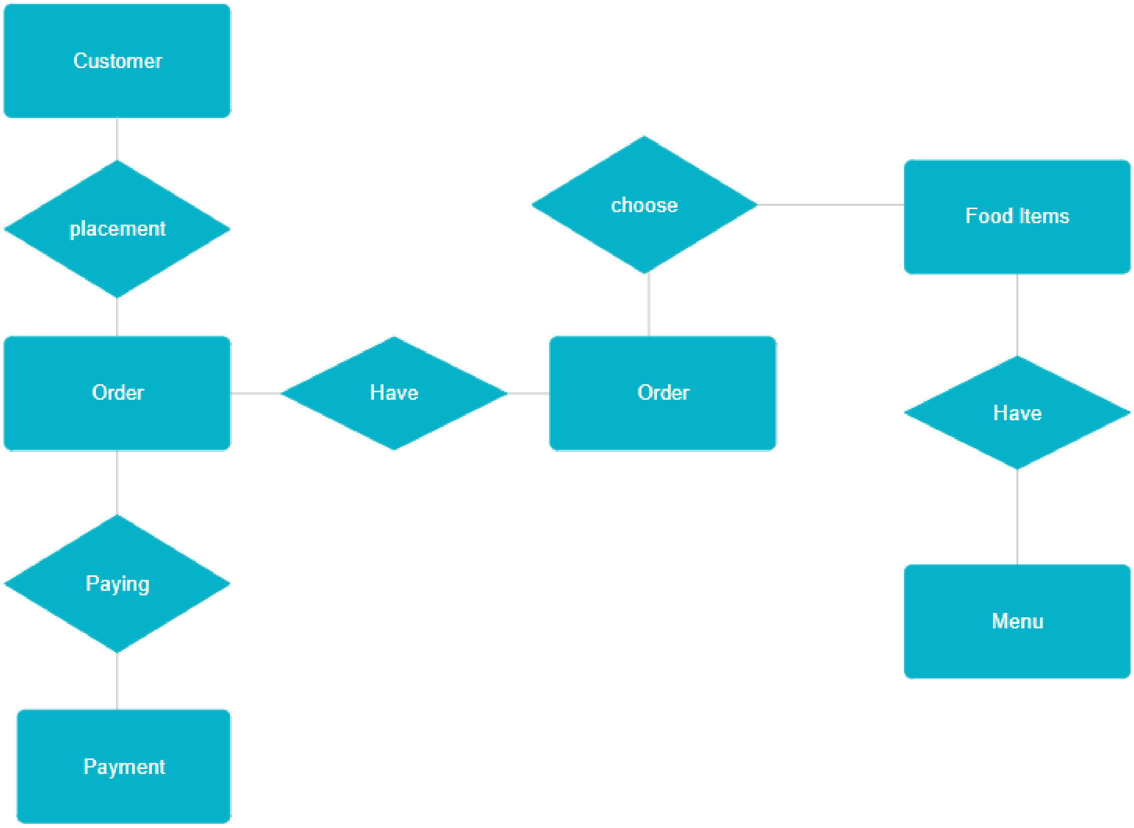
5.2 Internal Interface Requirements

5.2.1 Software Interface Requirements

- The software will transmit the selected food items to corresponding restaurant serving that food.
- The software shall conduct a poll to determine whether a requested food item is available.
- A non-available food item would be highlighted in advanced.

6. Data requirements

6.1 Logical Data Model



Logical Data Model

6.2 Data Dictionary

DATA ELEMENT	DESCRIPTION	DATA TYPE	LENGTH	VALUE
Order ID	A unique identifier for each food order placed by the customer.	Alphanumerical	8	Alphabets and numbers
Customer Name	The name of the customer who is placing the food order.	Alphabetic		
Customer Address	The address where the food order should be delivered.	Alphanumeric		Address be in Chinese
Customer Ph no.	The contact number of the customer for order confirmation and delivery updates.	Integers:	13	
Order Date	The day at which the food order was placed.	date, YYYY/MM/DD	10	
Order Time	The time at which the food order was placed.	time, HH:MM	5	
Order Items	The list of food items and quantities	alphabetic		

	ordered by the customer.			
Total Cost	The total cost of the food order, including taxes and delivery charges.	numeric	yyyy.mm	
Payment Method	The method used by the customer to pay for the food order (e.g., cash, credit card, online payment).	payment amount + payment method + transaction number	16	Alipay, Wechat, Student Card Pay later(via registering ph no./ wechat-id , only incase of dine in)
Order Status	The current status of the food order .	Alphabetic e.g confirmed, preparing, out for delivery, delivered).		incomplete, accepted, prepared, pending delivery, delivered, canceled
Delivery Driver	The name of the delivery driver assigned to deliver the food order.	Alphabetic		
Delivery Time	The estimated time for the delivery of the food order to the customer's address.	time, HH:MM	5	
Special Instructions	Any specific instructions provided by the customer for the preparation	Alphabetic		

	or delivery of the food order.			
Restaurant Name	The name of the restaurant from which the food order is being placed.	Alphabetic		
Restaurant Address	The address of the restaurant from which the food order is being placed.	Alphanumeric		Alphabets and numbers i.e (Name-shop number- floor no.)

6.3 Data Integrity, Retention, and Disposal

6.3.1 Data Integrity:

Data integrity ensures that the information stored is accurate, consistent, and reliable. This can be achieved by implementing measures such as validation checks, encryption of sensitive data, and access controls to prevent unauthorized changes to the data.

- **Data Accuracy:** The food ordering system must ensure that all customer information, order details, payment information, and delivery status are accurate and up-to-date. This includes real-time updates on order status and accurate customer details.
- **Data Consistency:** The system have to maintain consistency in the data with different modules such as order processing, payment processing, and delivery tracking. This ensures that all data related to an order is synchronized and consistent.
- **Data Security:** The system must ensure that all customer information, payment details, and order data are secure and protected from unauthorized access or misuse. This includes encryption of sensitive data and secure storage practices.
- **Data Validation:** The system must validate all input data such as customer details, order information, and payment details to ensure that they are correct and complete. This helps in preventing errors and inconsistencies in the data.

- **Data Accessibility:** The system must provide authorized users with access to the necessary data and ensure that it is available when needed. This includes role-based access control to restrict access to sensitive data.

6.3.1 Retention:

The retention of data in a food ordering system should be in compliance with applicable data protection laws and regulations. Customer personal information, such as name, address, and phone number, should be retained for as long as necessary to fulfill the purpose for which it was collected, such as processing and delivering food orders. However, it is important to regularly review and securely dispose of any data that is no longer needed to minimize the risk of unauthorized access or misuse.

- The food ordering system must retain all relevant data such as order history, customer feedback, and delivery records for future reference and analysis. This includes storing data for a specified period as per legal and business requirements.
- The system must maintain an audit trail of all data changes and user actions to track any unauthorized or suspicious activities. This helps in maintaining transparency and accountability in the system.

6.3.2 Disposal:

In disposing of data, it is important to do so securely to prevent unauthorized access or misuse of the information. This can involve permanently deleting electronic records using secure deletion methods and securely destroying physical records. It is also important to ensure that any backups or archived data are also securely disposed of when they are no longer needed.

- 3.1 The system must have a defined data disposal policy to securely delete or destroy data that is no longer required. This includes proper data sanitization methods to prevent any unauthorized access to discarded data.
- 3.2 The data disposal process must comply with regulatory requirements and best practices for data privacy and security. This ensures that sensitive information is not exposed after it is no longer needed.