

# **SOFTWARE REQUIREMENT SPECIFICATION**

## **FOR**

### **FOOD ORDERING SYSTEM**

**Under the Subject of**  
**Software Engineering**  
**(Semester – I)**

**MRS IRMI SAIYED**  
**(Faculty Guide)**

**MR KEVAL PATEL**  
**(FY5-Head of Department)**

**Academic Year**  
**2024 – 25**



**L. J. Institute of Engineering & Technology, Ahmedabad (LJU)**



**L. J. INSTITUTE OF ENGINEERING AND TECHNOLOGY**

**FY5-Department 2024 – 25**

## **CERTIFICATE**

Date:

This is to certify that the Software Engineering Work entitled “**FOOD ORDERING SYSTEM**”, carried out by the group of students mentioned below under my guidance is approved for the Degree of Bachelor of Engineering (Semester-I) of L. J. Institute of Engineering and Technology (LJU) during the academic year 2023-24.

1. Ammar Husain Gheewala (24002171710005)
2. Meet Jayantilal Kachhadiya (24002171910010)
3. Yug Vipulbhai Goyani ( 24002171810010)
4. Kevin Lalitkumar Vasani (24002171310177)
5. Krish Maganbhai Kachhadiya (24002171310055)

**Guide**

**MRS IRMI SAIYED**

FY5- Department, L.J.I.E.T.

## **Table of Contents**

<b>Sr. No.</b>	<b>Particulars</b>	<b>Page No.</b>
<b>1</b>	<b>INTRODUCTION</b>	<b>1</b>
1.1	Purpose	1
1.2	Scope	1
1.3	Glossary	1
1.4	Overview	2
<b>2</b>	<b>OVERALL DESCRIPTION</b>	<b>3</b>
2.1	Problem Statement	3
2.2	Existing System	3
2.3	Proposed System	3
2.4	Product Functions	3
2.5	User Characteristics	4
	2.5.1 User requirements	4
	2.5.2 User Education Level	5
	2.5.3 User's Technical Expertise	5
2.6	Constraints	5
2.7	Assumption and Dependencies	5
<b>3</b>	<b>REQUIREMENT SPECIFICATION</b>	<b>6</b>
3.1	Functional Requirements	6
	3.1.1 Performance requirements	6
	3.1.2 Design constraints	6
	3.1.3 Hardware Requirements	6

	<b>3.1.4 Software Requirements</b>	<b>7</b>
	<b>3.1.5 Other Requirement</b>	<b>7</b>
<b>3.2</b>	<b>Non-Functional Requirements</b>	<b>8</b>
	<b>3.2.1 Security</b>	<b>8</b>
	<b>3.2.2 Reliability</b>	<b>8</b>
	<b>3.2.3 Availability</b>	<b>8</b>
	<b>3.2.4 Maintainability</b>	<b>8</b>
	<b>3.2.5 Supportability</b>	<b>8</b>
<b>4</b>	<b>DIAGRAMS</b>	<b>9</b>
<b>4.1</b>	<b>Data Flow Diagram</b>	<b>9</b>
	<b>4.1.1 Level-0 DFD</b>	<b>9</b>
	<b>4.1.2 Level-1 DFD</b>	<b>10</b>
	<b>4.1.3 Level-2 DFD</b>	<b>11</b>
<b>4.2</b>	<b>Activity Diagram</b>	<b>12</b>
<b>4.3</b>	<b>User Case Diagram</b>	<b>13</b>
<b>4.4</b>	<b>Class Diagram</b>	<b>14</b>
<b>4.5</b>	<b>Sequence Diagram</b>	<b>15</b>
<b>5</b>	<b>CONCLUSION</b>	<b>16</b>
<b>6</b>	<b>SCOPE OF THE PROJECT</b>	<b>17</b>

## **1. Introduction**

### **1.1. Purpose:**

The primary purpose of the Food Ordering System is to provide a seamless and efficient way for users to browse, order, and manage food deliveries from multiple restaurants. The system enhances user experience by digitizing the entire process, from restaurant selection to final billing. It allows users to register, log in, and access menus from various restaurants, ensuring a smooth and convenient ordering process. By automating order tracking and bill calculations, the system streamlines operations for both customers and restaurant owners, making food ordering quick, reliable, and user-friendly.

### **1.2. Scope:**

The Food Ordering System will empower users to search for restaurants, browse menus, and place orders efficiently. The system will allow users to:

- Create and manage accounts.
- Browse food items categorized by restaurant and cuisine.
- Add selected items to the cart and process orders.
- Calculate the total bill, including applicable taxes and discounts.
- Participate in a quiz-based discount system to earn additional savings.
- Ratings.

### **1.3 Glossary:**

- User: A person who registers on the system to place orders.
- Restaurant Database: A collection of restaurant profiles and menus.
- Menu Browsing: Feature allowing users to view and select food items.
- Order Cart: A temporary storage of selected food items before finalizing the order.
- Checkout Process: The final step where users confirm and pay for orders.
- Quiz-based Discounts: A system where users answer questions to receive discounts.
- Billing System: Calculates total bill with applicable taxes and discounts.

## **1.4 Overview:**

The Food Ordering System is designed to provide an easy-to-use digital platform for ordering food from multiple restaurants. The system allows users to register, log in, and browse categorized menus. Users can place orders, track their food, and make payments while enjoying potential discounts through interactive quizzes. The platform also ensures a seamless experience by notifying users about order updates and estimated delivery times. With enhanced search and filtering options, users can quickly find their favorite meals. The system aims to improve convenience and efficiency in food ordering, making it more accessible and reliable for both customers and restaurant owners.

## **2 OVERALL DESCRIPTION**

### **2.1 Problem Statement**

Manual food ordering processes can be inefficient, leading to incorrect orders, long wait times, and customer dissatisfaction. Users may struggle to find their preferred meals due to poor categorization and search functionality. Lack of automation in billing and discount application results in errors and delays. A digital Food Ordering System will address these issues by centralizing restaurant listings, streamlining order placement, and automating billing, ensuring accuracy and efficiency in the ordering process.

### **2.2 Existing System**

Traditional food ordering methods involve phone calls or physical visits to restaurants, which can be time-consuming and prone to miscommunication. Customers have limited visibility into available dishes and pricing. There are no automated discount mechanisms or real-time tracking, which impacts convenience and satisfaction. Additionally, manual billing and order records increase the risk of errors and inefficiencies.

### **2.3 Proposed System**

The proposed Food Ordering System will automate and enhance the ordering experience. Key features include:

- User Account Management: Allows users to register, log in, and manage their profiles.
- Restaurant and Menu Management: Displays detailed menus for multiple restaurants.
- Order Processing: Enables users to select dishes, add to cart, and confirm orders.
- Billing Automation: Calculates total order cost, including taxes and discounts.
- Quiz-based Discounts: Users can answer questions to earn discounts

### **2.4 Product Functions**

- Restaurant Database Management: Maintain accurate restaurant and menu data.
- Order Management: Process orders and track transactions.
- Search Functionality: Allow users to search dishes by name, category, or restaurant.
- Automated Billing and Discounts: Apply discounts and calculate GST automatically.
- User Interface: Ensure a user-friendly interface for seamless navigation.

- Table Reservation System: Allow users to book tables in advance at restaurants.
- Customer Reviews & Ratings: Provide options for users to rate and review restaurants and dishes.
- Push Notifications & Alerts: Send real-time updates about order status, offers, and promotions.
- AI-Based Recommendations: Suggest dishes and restaurants based on user preferences and past orders.
- Multi-Language Support: Offer multiple language options for a better user experience.
- Delivery Partner Management: Assign and track delivery personnel for efficient order FulFillment.
- Loyalty & Reward Programs: Provide reward points or cashback for frequent users.
- Inventory Management for Restaurants: Help restaurants manage stock levels and ingredient availability.
- Analytics & Reports: Generate insights on sales, customer preferences, and restaurant performance.
- Customizable Menu Options: Allow restaurants to update menus, pricing, and availability in real time.
- Order Scheduling: Let users pre-order meals for a specific time.
- Complaint & Dispute Resolution: Provide a system for handling customer complaints and refunds.

## **2.5 User Characteristics**

### **2.5.1 User Requirements**

- Access to user accounts and restaurant menus.
- Ability to search for food items by category, restaurant, or dish name.
- Efficient order placement, tracking, and bill calculation.
- Administrative reporting for restaurant owners.

### **2.5.2 User Education Level**

- Users range from general customers to restaurant owners.
- Basic digital literacy required for smooth navigation.
- At least User of the system should be comfortable with English Language

### **2.5.3 User's Technical Expertise**

- Users may have varying levels of technical knowledge.
- System designed for ease of use, requiring minimal technical skills.

## **2.6 Constraints**

- Limited Scalability: System is optimized for small to medium-scale food businesses.
- Security Concerns: Potential vulnerabilities in user data and payment processing.
- Platform Dependency: May be restricted to certain operating systems.
- Internet Dependency: Requires stable internet connection for real-time functionality.
- User Training: New users may need guidance to navigate the platform efficiently.

## **2.7 Assumptions**

- Users have access to the internet.
- Restaurants will update their menus regularly.
- Customers will enter valid order details.
- Payments will be processed through third-party gateways.
- System will be used by a single user per session.

### 3 REQUIREMENT SPECIFICATION

#### 3.1 Functional Requirements

##### 3.1.1 Performance requirements

- **User Satisfaction:** The system is such that it stands up to the user expectations.
- **Response Time:** The response of all operations is good.
- **Error Handling:** Response to user errors and undesired situation has been taken care of to ensure that the system operates without halting.
- **Safety and Robustness:** The system is able to avoid or tackle disastrous action. In other words, it should be fool proof.
- **Portable:** The software should not be architecture specific. It should be easily transferable to other platforms if needed.
- **User Friendliness:** The system is easy to learn and understand. A native user can also use the system effectively, without any difficulties.

##### 3.1.2 Design constraints

There are a number of factors in the client's environment that may restrict the choices of a designer. Such factors include standards that must be followed, resource limits, operating environment, reliability and security requirements and policies that may have an impact on the design of the system.

- **Standard Compliances:** This specifies the requirement for standards the system must follow. The standards may include the report format and accounting properties.
- **Hardware Limitations:** Hardware limitations can include the types of machine to be used, operating system available on the system, languages support and limits on primary and secondary storage.
- **Reliability and Fault Tolerance:** Fault tolerance requirement can be placing a constraint on how the system is to be designed. Recovery requirements are often an integral part here, detailing what the system should do if some failure occurs to ensure certain properties. Reliability requirements are very important for critical application.
- **Security:** Security requirements are particularly significant in defence system and database system. They place restrictions on the use of certain commands, control access to data, provide different kinds of access requirements for different people, require the use of passwords and cryptography techniques and maintain a log of activities in the system.

##### 3.1.3 Hardware Requirements

- **Processor:** Minimum 2 GHz dual-core processor for efficient performance.

- **Display:** Monitor with a resolution of 1024x768 or higher for clear visibility of the Command Prompt interface.
- **Input Devices:** Standard keyboard and mouse for navigating the command-line interface.
- **Backup Device:** External hard drive or network-attached storage (NAS) for regular data backup and recovery.

#### **3.1.4 Software Requirements**

- **Operating System:** Windows 10/11, Linux (Ubuntu)
- **Command Line Interface (CLI):** Default command prompt or terminal provided by the operating system.
- **Programming Language:** Java (23) with necessary libraries installed
- **Text Editor/IDE:** Code editor such as Visual Studio Code, IntelliJ.
- **Backup Software:** Manual backup tools or scripts for regular data backup and recovery processes.

#### **3.1.5 Other Requirement**

- **Maintenance and Updates:**
  1. Regular software updates to address bugs and security issues.
  2. Scheduled maintenance for system optimization.
- **Data Privacy:**

Compliance with data privacy regulations to protect user information.
- **User Authentication:**
  1. Secure login system for staff and patrons.
  2. Role-based access control to restrict permissions.
- **Customization:**

Flexibility to customize system features and settings according to library needs.
- **Security**
- **Portability**
- **Correctness**
- **Efficiency**
- **Flexibility**
- **Testability**
- **Reusability**

## **3.2 Non-Functional Requirements**

### **3.2.1 Security**

The system must automatically log out all customers after a period of inactivity. The system should not leave any cookies on the customer's computer containing the user's password. The system's back-end servers shall only be accessible to authenticated management.

### **3.2.2 Reliability**

The reliability of the overall project depends on the reliability of the separate components. The main pillar of reliability of the system is the backup of the database which is continuously maintained and updated to reflect the most recent changes. Also, the system will be functional under a container. Thus, the overall stability of the system depends on the stability of the container and its underlying OS.

### **3.2.3 Availability:**

The system should be available at all the times, meaning the user can access it using a web browser, only restricted by the down time of the server on which system runs. A customer friendly system which is in access of people around the worlds should work 24 hours. In case of a hardware failure or database corruption, a replacement page will be shown. Also, in case of hardware failure or database corruption backups of the database should be retrieved from the server and saved by the Organizer. Then the service will be restarted. It means 24x7 availability.

### **3.2.4 Maintainability:**

In case of a failure, a re-installation of the system will be done. Also, the software design is being done with modularity in mind so that maintainability can be done efficiently.

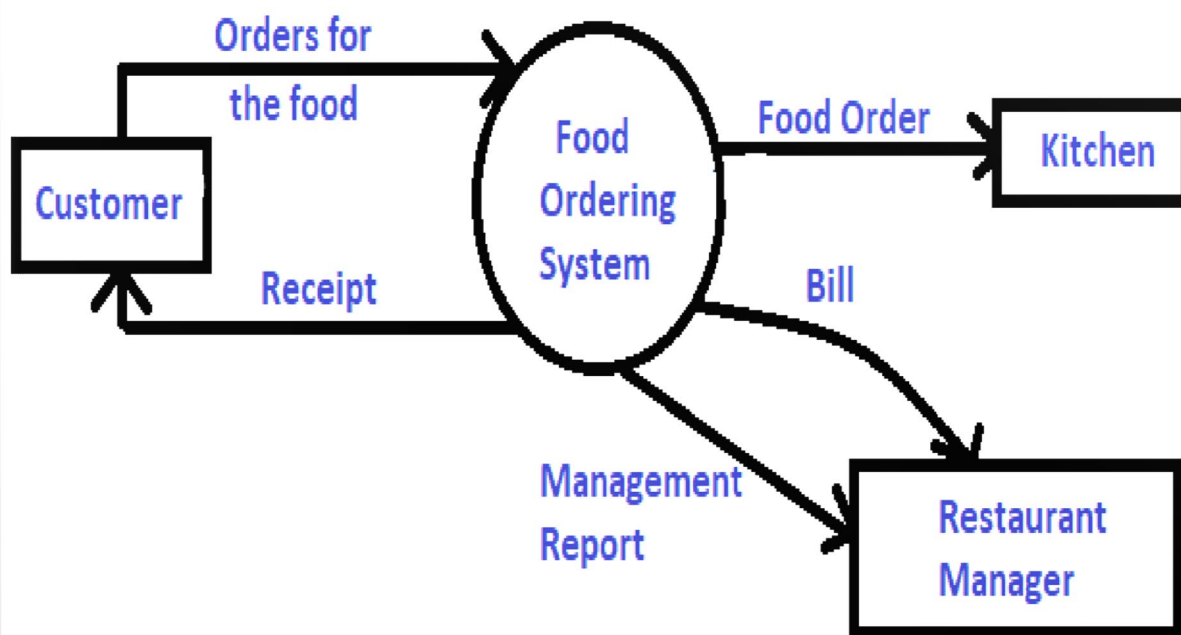
### **3.2.5 Supportability:**

The code and supporting modules of the system will be well documented and easy to understand. Online user documentation and Help system requirements will be provided.

## 4 DIAGRAMS

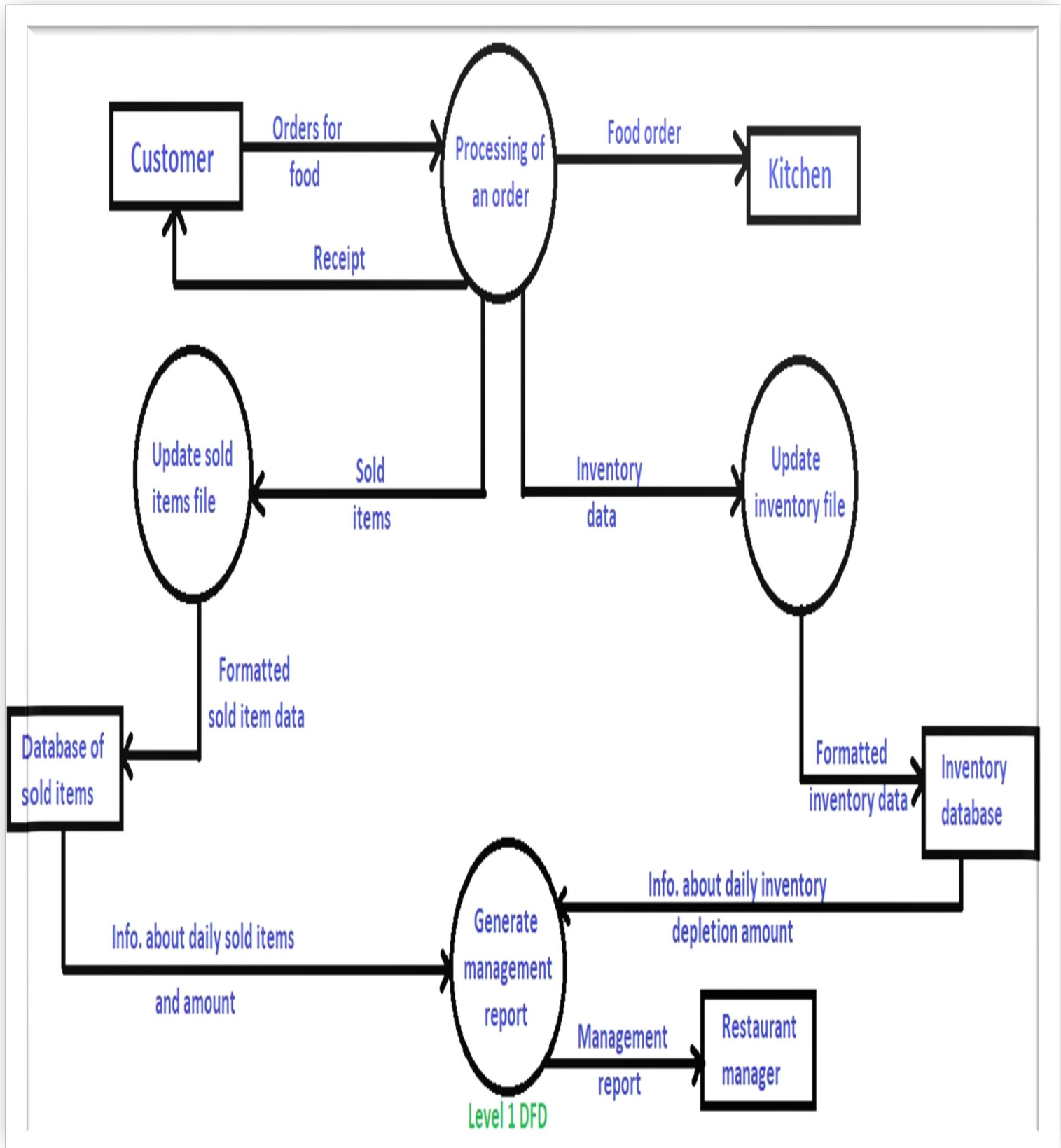
### 4.1 Data Flow Diagram

#### 4.1.1 Level 0 DFD:

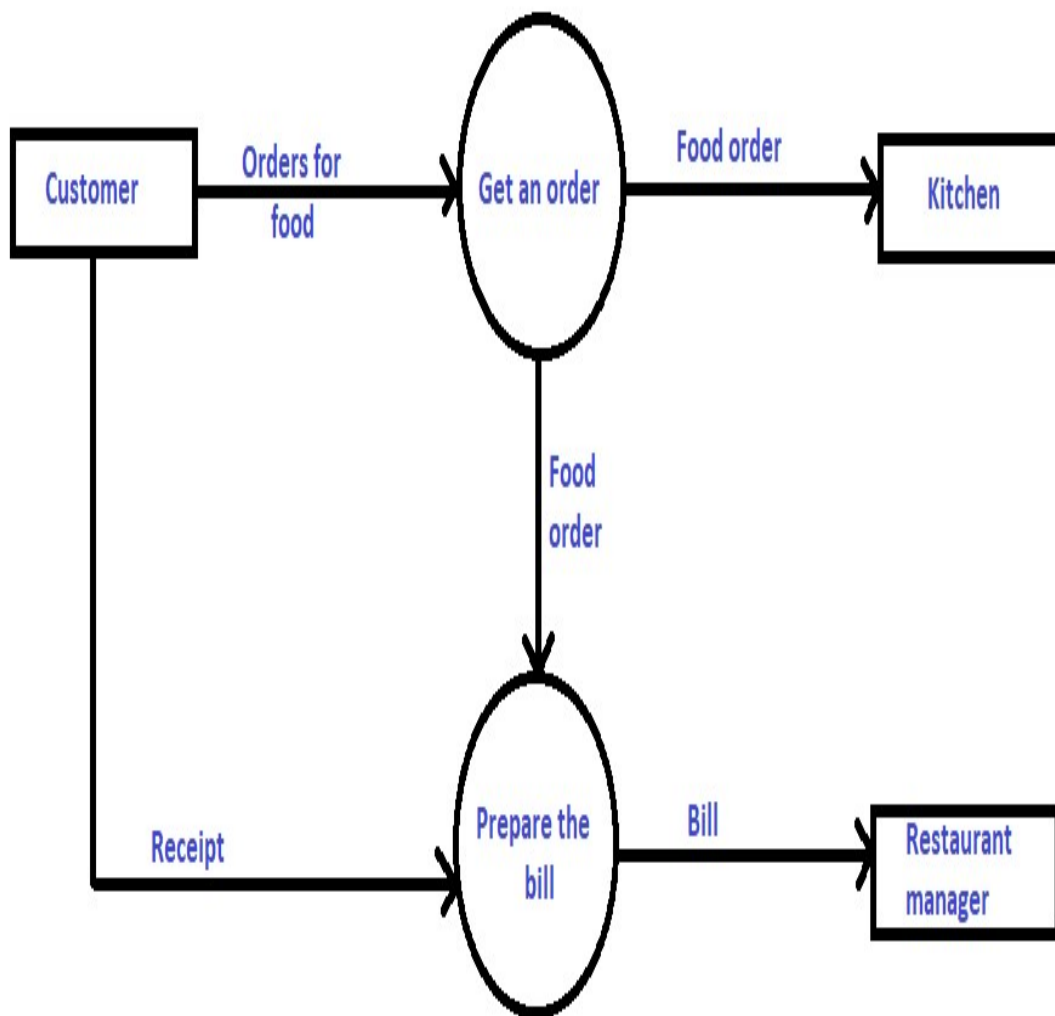


Level 0 DFD

#### 4.1.2 Level 1 DFD:

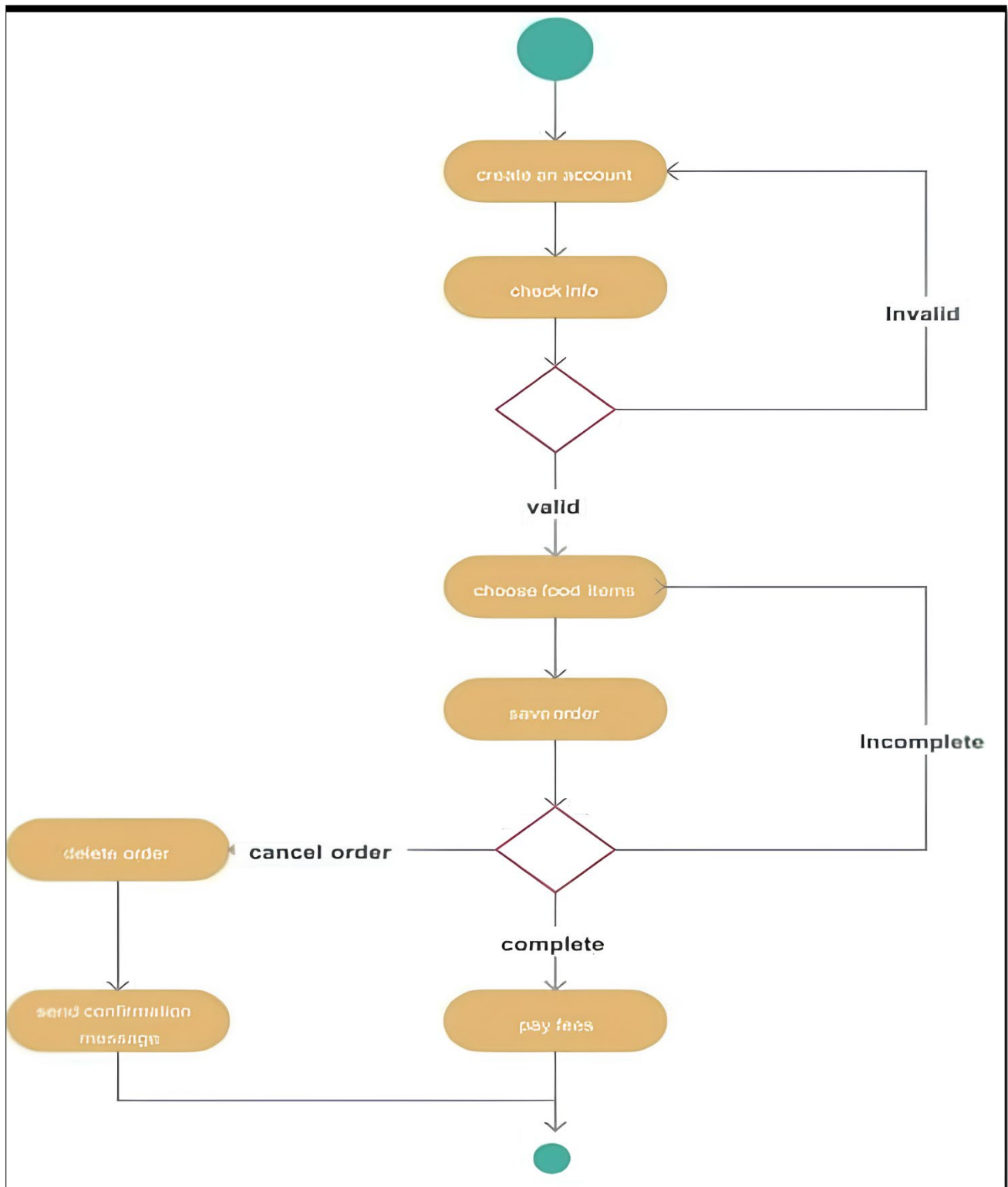


#### 4.1.3 Level 2 DFD:

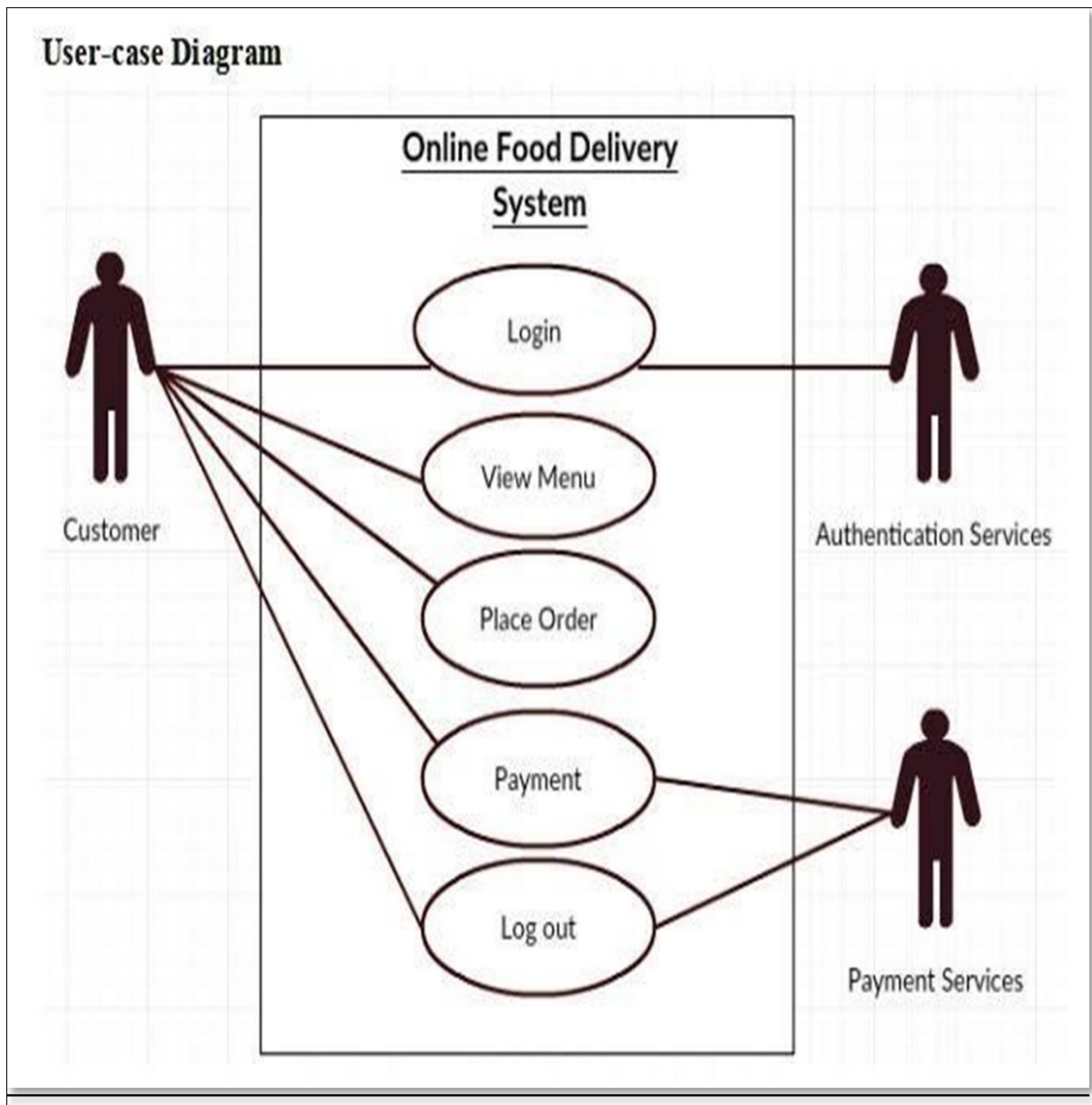


Level 2 DFD

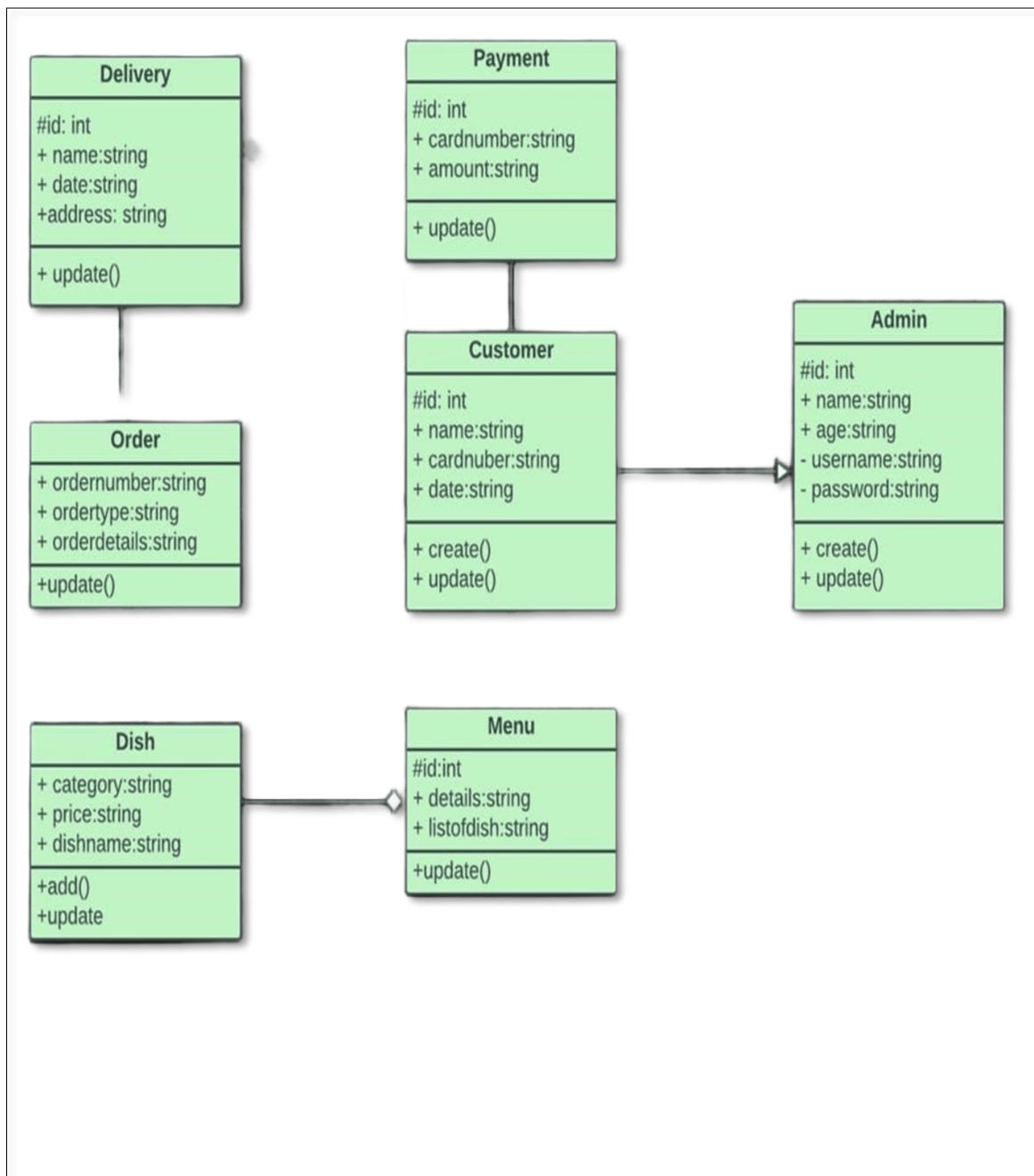
## 4.2 Activity Diagram:



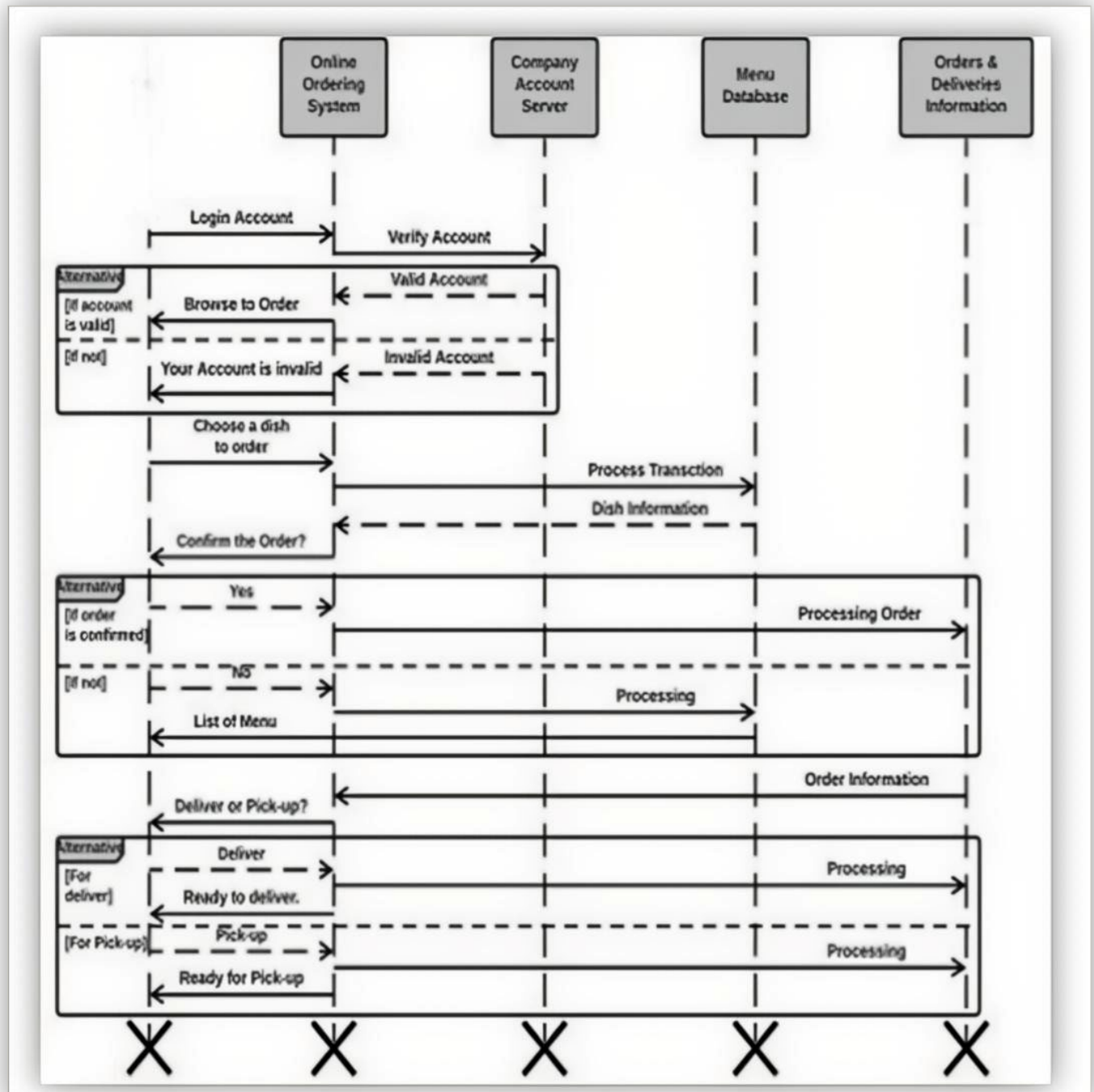
### 4.3 User-Case Diagram:



#### 4.4 Class Diagram:



#### 4.5 Sequence Diagram:



## 5 CONCLUSION

We can hereby conclude that:

- The system effectively automates the entire food ordering process, eliminating the need for manual order management and paperwork.
- The cost and benefit analysis shows that this system significantly reduces operational costs for restaurants while enhancing efficiency and customer satisfaction.
- The system is secure and scalable, capable of handling multiple orders simultaneously without performance issues.
- The design of the system ensures user-friendliness and efficiency, making it easy to use for customers, restaurant staff, and delivery personnel.

## 6 SCOPE OF THE PROJECT

The scope of this project is to replace the traditional manual food ordering process with an advanced computerized system. This system is designed to be user-friendly and does not require extensive training to operate. It eliminates paper-based order management, reduces human errors, and automates calculations, including bill generation. The system is capable of storing thousands of records efficiently and ensures smooth operations for both customers and restaurant staff.

Some key advantages of this system are:

- **Automatic Order Number Generation:** The system generates unique order numbers automatically, reducing the chances of duplication or errors.
- **Error Detection and Alerts:** If any invalid entry is made, the system immediately notifies the user with an error message to prevent mistakes.
- **Real-time Order Tracking:** Customers can track their orders in real-time, and restaurants can update order statuses seamlessly.
- **Automated Billing:** The system calculates the total bill, including taxes and discounts, and generates an invoice instantly.
- **Inventory Management:** Whenever an item is ordered, the stock is automatically updated, helping restaurants manage their inventory efficiently.
- **Seamless Customer Management:** The system keeps track of customer details, order history, and preferences, making it easier to serve repeat customers.

This food ordering system improves efficiency, reduces manual workload, and enhances the overall customer experience.