



Bangladesh Open University  
School of Science & Technology  
Bachelor of Science in Computer Science and Engineering

**Notebook for Lab Exam**

Course Title: System Analysis & Design Lab

Course Code: CSE22P5

***Submitted by***

Safkat Jamil (Imon)

Student Id No.: 20-0-52-801-061

Session: 2020-2021

2<sup>nd</sup> Year 2<sup>nd</sup> Semester 2022

Term: 222

Dhaka Regional Center

***Submitted to***

Samrat Kumar Dey

Lecturer

School of Science and Technology

Bangladesh Open University

***Date of Submission***

13 December 2024

## Contents

### Lab Report

Report-1: Create a flowchart of general problem-solving approach.....	3
Report-2: Create a use case diagram of hospital management system .....	5
Report-3: Create a class diagram of an online processing system .....	8
Report-4: Create a sequence diagram of Dhaka Metrorail and the experiences of commuters .....	11

### Lab Problem

Problem-1: Sequence Diagram of a Phone Call.....	15
Problem-2: Use Case Diagram of ATM System .....	17
Problem-3: Class Diagram of a company management system .....	19
Problem-4: Level 1 Data Flow Diagram of an Airline Reservation System .....	22
Problem-5: Use Case Diagram of a Hotel Management System .....	24

# Lab Report

**Lab Report No.:** 01

**Lab Report Name:** Create a flowchart of general problem-solving approach.

**Date of Submission:** 8 March 2024

**Instrument:**

- Computer
- EdrawMax Software, etc.

**Procedure:**

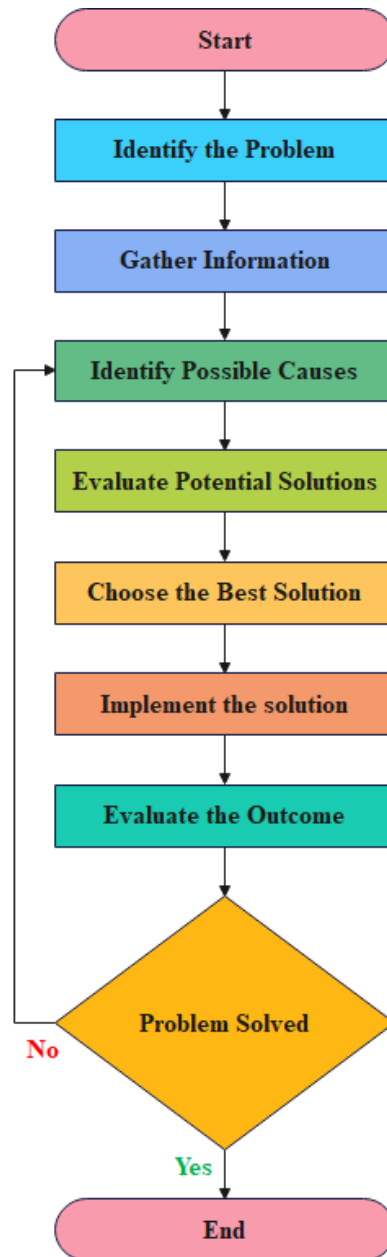
- At first, we have to know about general problem-solving.
- After knowing about general problem-solving we have to power on the computer.
- When the computer is ready to use, we have to open EdrawMax software.
- After opening EdrawMax software, we have to create the flowchart of general problem-solving approach.
- Finally, after drawing the flowchart, we have to ensure the flowchart is correct.

**General problem-solving steps:**

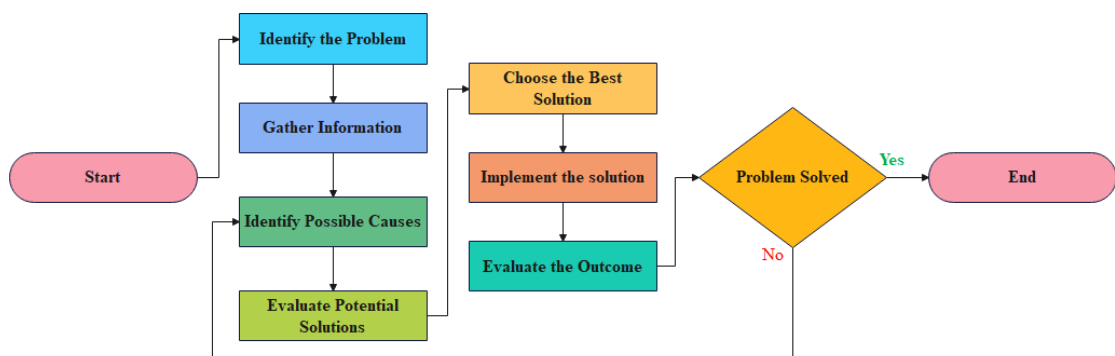
1. **Identify the problem:** Clearly state the issue you're trying to solve. Be specific and avoid ambiguity.
2. **Gather information:** Collect relevant data about the problem. This might involve asking questions, researching, or observing the situation.
3. **Identify possible causes:** Brainstorm potential reasons behind the problem. Consider different perspectives and factors that could be contributing.
4. **Evaluate potential solutions:** Analyze each possible solution based on its feasibility, effectiveness, and potential consequences. Consider constraints and resources available.
5. **Choose the best solution:** Select the solution that seems most likely to address the problem effectively and efficiently.
6. **Implement the solution:** Put your chosen solution into action.
7. **Evaluate the outcome:** Assess whether the solution successfully resolved the problem.
  - 7.1. Yes: The problem is solved.
  - 7.2. No: The problem persists or new issues arise. Re-evaluate the situation and iterate through steps 3-7, potentially trying a different solution or gathering more information.

## Flowchart:

1.



2.



**Lab Report No.:** 02

**Lab Report Name:** Create a use case diagram of hospital management system.

**Date of Submission:** 15 March 2024

**Instrument:**

- Computer
- EdrawMax Software, etc.

**Procedure:**

- At first, we have to learn about hospital management system.
- After learning about hospital management system, we have to power on the computer.
- When the computer is ready to use, we have to open EdrawMax software.
- After opening EdrawMax software, we have to create the use case diagram of a hospital management system.
- Finally, after drawing the use case diagram, we have to ensure the use case diagram is correct.

**Hospital Management System:**

1. Actors

- Patient: An actor related to hospital management system.
- Doctor: An actor related to hospital management system.
- Nurse: An actor related to hospital management system.
- Receptionist: An actor related to hospital management system.
- Employee: An actor related to hospital management system.
- Technologist: An actor related to hospital management system.
- Technician: An actor related to hospital management system.

2. Doctor Appointment Info: This likely refers to updating the system on appointments.

3. Check Up: This could be the initial consultation with a doctor.

4. Tests Appointment: This could be for lab tests, X-rays, etc.

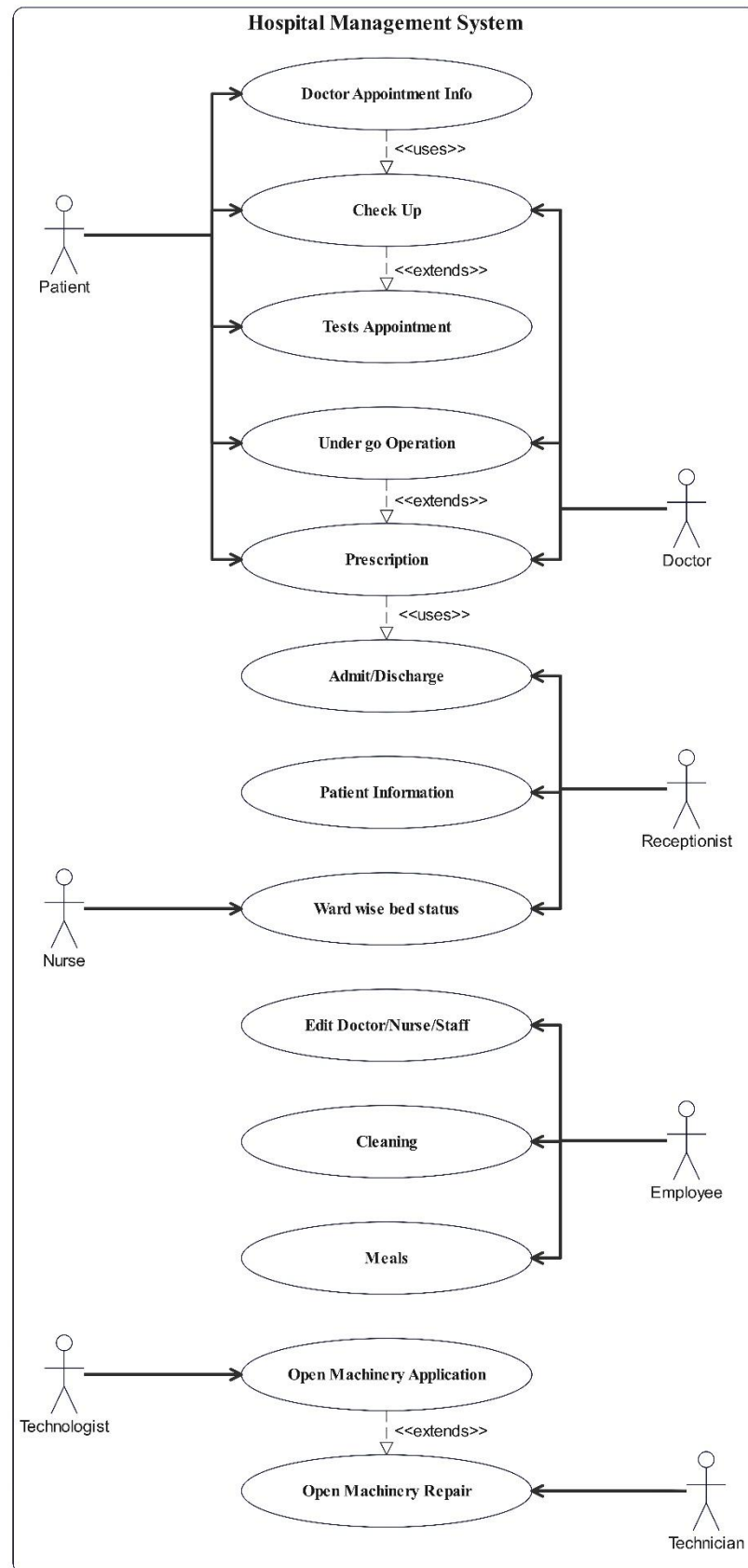
5. Undergo Operation: This could be a surgical procedure.

6. Prescription: This could be for medication after a diagnosis.

7. Admit/Discharge: This refers to admitting a patient to the hospital or discharging them.

8. Patient Information: This refers to registering new patients or updating a patient's medical record.
9. Ward wise bed status: This refers to updating the system on bed availability in different wards.
10. Edit Doctor/Nurse/Staff: This could be for updating staff information within the system.
11. Cleaning: This likely refers to logging cleaning tasks.
12. Meals: This likely refers to ordering meals for patients.
13. Open Machinery Application: This could be launching specific software to operate medical equipment.
14. Open Machinery Repair: This could be for reporting malfunctioning equipment.

## Use Case Diagram:





**Lab Report No.:** 03

**Lab Report Name:** Create a class diagram of an online processing system.

**Date of Submission:** 29 March 2024

**Instrument:**

- Computer
- EdrawMax Software, etc.

**Procedure:**

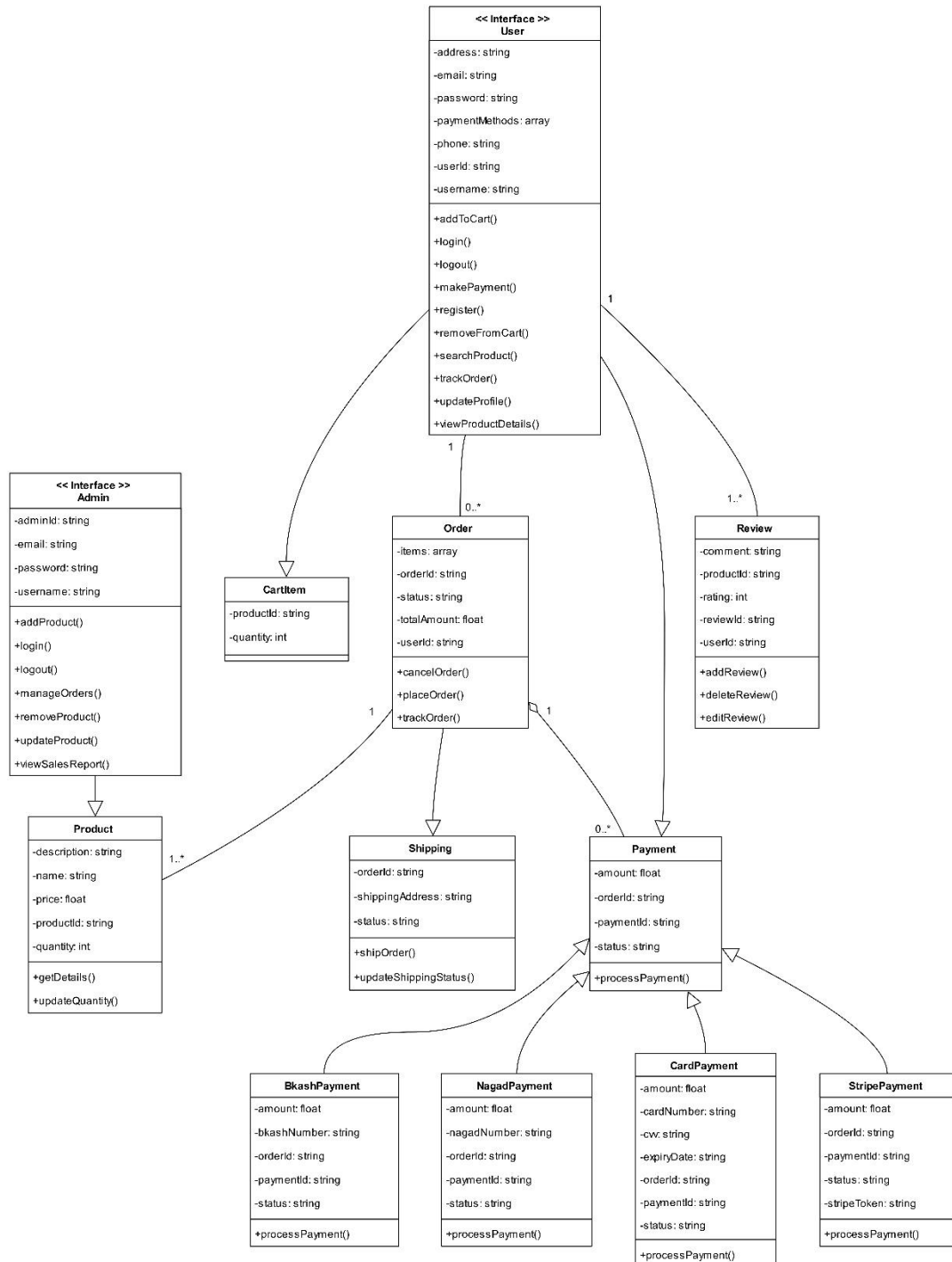
- At first, we have to learn about online processing system.
- Now we have to power on the computer.
- When the computer is ready to use, we have to open EdrawMax software.
- After opening EdrawMax software, we have to create the class diagram of online processing system.
- Finally, after drawing the class diagram, we have to ensure the class diagram is correct.

**Online Processing System:**

1. Admin: Represents an administrator of the online processing system.
  - Attributes
    - i. adminId: Unique identifier for the admin.
    - ii. email: Email address of the admin.
    - iii. password: Password of the admin.
    - iv. username: Username of the admin.
2. User: Represents a user of the online processing system.
  - Attributes
    - i. Address: Address of the user.
    - ii. Email: Email address of the user.
    - iii. Password: Password of the user.
    - iv. Payment Methods: Array of payment methods associated with the user.
    - v. Phone: Phone number of the user.
    - vi. User Id: Unique identifier of the user.
    - vii. User Name: Username of the user.
3. Product: Represents a product available for purchase in the system.
  - Attributes
    - i. Description: Description of the product.
    - ii. Name: Name of the product.

- iii. Price: Price of the product.
  - iv. Product Id: Unique identifier of the product.
  - v. Quantity: Quantity of the product available in stock.
4. Cart Item: Represents an item added to the user's shopping cart.
- Attributes:
    - i. Product Id: Unique identifier of the product.
    - ii. Quantity: Quantity of the product added to the cart.
5. Order: Represents an order placed by a user.
- Attributes:
    - i. Items: Array of items included in the order.
    - ii. Order Id: Unique identifier of the order.
    - iii. Status: Status of the order (e.g., processing, shipped).
    - iv. Total Amount: Total amount of the order.
    - v. User Id: Unique identifier of the user who placed the order.
6. Payment: Represents a generic payment made for an order.
- Attributes:
    - i. Amount: The amount of the payment.
    - ii. Order Id: Unique identifier of the associated order.
    - iii. Payment Id: Unique identifier of the payment.
    - iv. Status: Status of the payment (e.g., pending, completed).
7. Shipping: Represents the shipping process for an order.
- Attributes:
    - i. Order Id: Unique identifier of the order being shipped.
    - ii. Shipping Address: Address to which the order will be shipped.
    - iii. Status: Status of the shipping process (e.g., pending, shipped).
8. Review: Represents a review submitted by a user for a product.
- Attributes:
    - i. Comment: Textual comment or review content.
    - ii. Product Id: Unique identifier of the product being reviewed.
    - iii. Rating: Numeric rating given by the user for the product.
    - iv. Review Id: Unique identifier of the review.
    - v. User Id: Unique identifier of the user who submitted the review.

## Class Diagram:



**Lab Report No.:** 04

**Lab Report Name:** Create a sequence diagram of Dhaka Metrorail and the experiences of commuters.

**Date of Submission:** 29 March 2024

**Instrument:**

- Computer
- EdrawMax Software, etc.

**Procedure:**

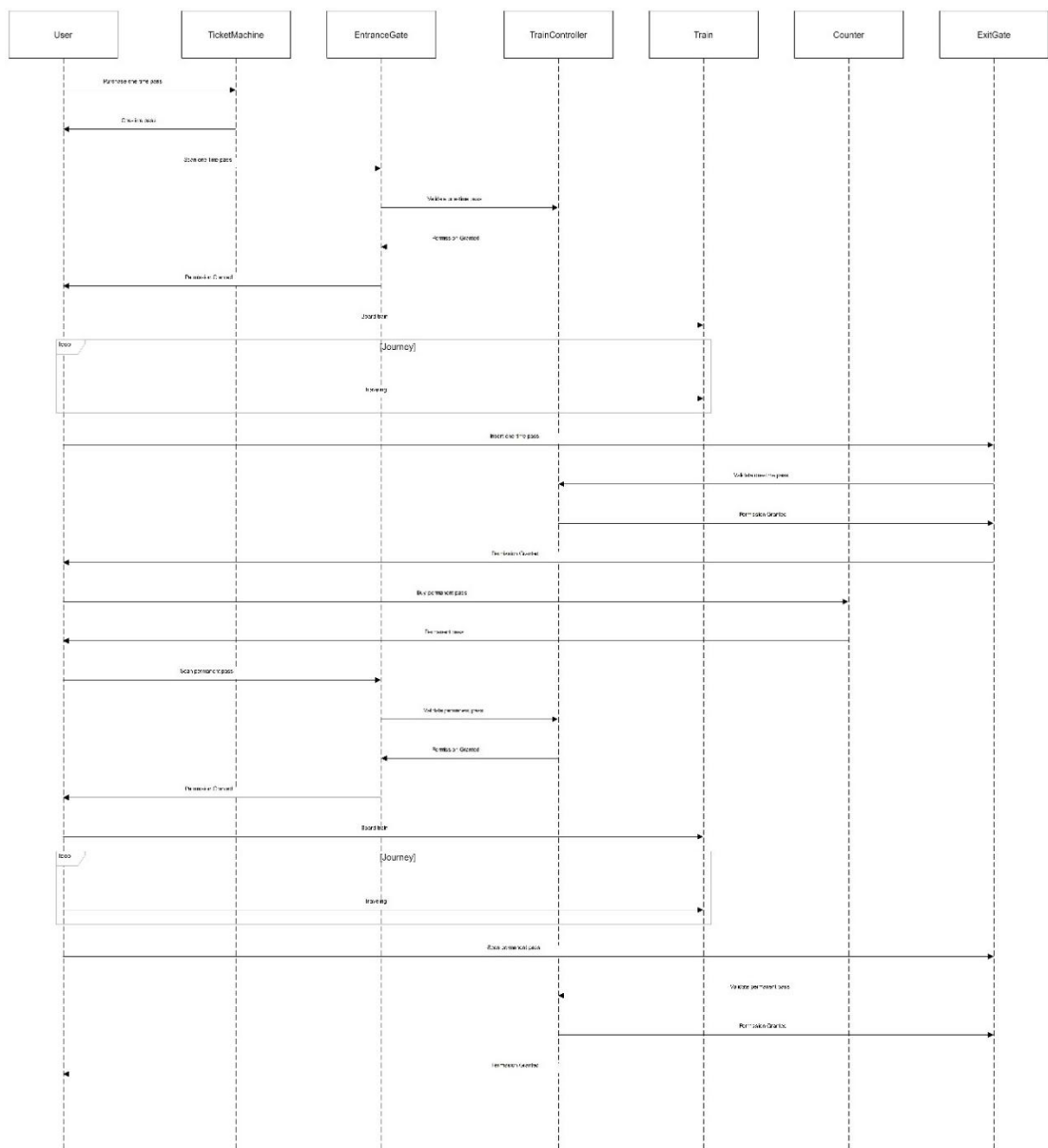
- At first, it is essential to become familiar with the Dhaka Metrorail and the experiences of commuters who use it.
- Now we have to draw a sequence diagram from what we have learned. First, power on the computer.
- When the computer is ready to use, we have to open EdrawMax software.
- After opening EdrawMax software, we have to create the sequence diagram.
- Finally, after drawing the sequence diagram, we have to ensure the sequence diagram is drawn correct.

**Dhaka Metrorail:**

1. User Purchases One-time Pass: The user interacts with the ticket machine to purchase a one-time pass for their journey.
2. Ticket Machine Dispenses One-time Pass: The ticket machine dispenses the one-time pass to the user.
3. One-time pass holder:
  - 3.1. User Scans One-time Pass at Entrance Gate: The user scans their one-time pass at the entrance gate of the train station.
  - 3.2. Entrance Gate Validates One-time Pass: The entrance gate validates the one-time pass with the train controller.
  - 3.3. Permission Granted: Upon successful validation, the entrance gate grants permission to the user to enter and board the train.
  - 3.4. User Boards Train: The user boards the train and begins their journey.
  - 3.5. Journey Loop: This loop represents the user's journey on the train. It's a repetitive process until they reach their destination.

- 3.6. User Reaches Exit Gate: Once the user reaches their destination, they approach the exit gate.
- 3.7. User Inserts One-time Pass at Exit Gate: One-time pass holders insert their pass into the exit gate machine for validation.
- 3.8. Exit Gate Validates One-time Pass: The exit gate validates the one-time pass with the train controller.
- 3.9. Permission Granted to Exit: Upon successful validation, the exit gate grants permission to the user to exit the train station.
- 4. Permanent pass holder:
  - 4.1. User Buys Permanent Pass: The user purchases a permanent pass from the counter.
  - 4.2. Counter Dispenses Permanent Pass: The counter dispenses the permanent pass to the user.
  - 4.3. User Scans Permanent Pass at Entrance Gate: The user scans their permanent pass at the entrance gate to board the train.
  - 4.4. Permission Granted: Similar to before, upon successful validation, the entrance gate grants permission to the user to enter and board the train.
  - 4.5. User Boards Train: The user boards the train and starts their journey.
  - 4.6. Journey Loop: The user's journey continues until they reach their destination.
  - 4.7. User Reaches Exit Gate: Once the user reaches their destination, they approach the exit gate.
  - 4.8. User Scans Permanent Pass at Exit Gate: Permanent pass holders scan their pass at the exit gate for validation.
  - 4.9. Exit Gate Validates Permanent Pass: The exit gate validates the permanent pass with the train controller.
  - 4.10. Permission Granted to Exit: Upon successful validation, the exit gate grants permission to the user to exit the train station.

Sequence Diagram:



# **Lab Problem**

## **Lab Problem No.: 01**

**Lab Report Name:** Sequence Diagram of a Phone Call.

### **Instrument:**

- Computer
- EdrawMax Software, etc.

### **Procedure:**

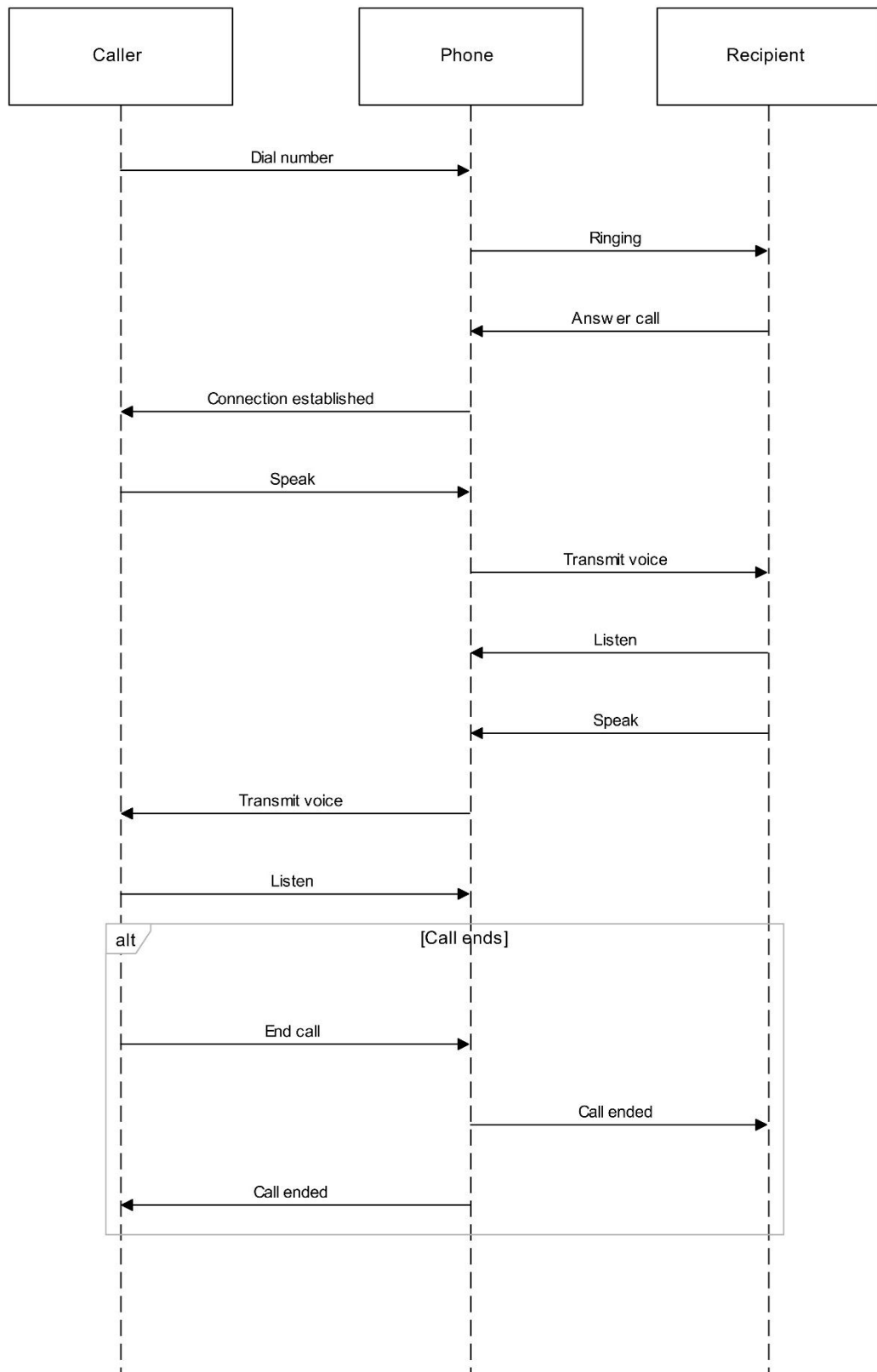
- At first, it is essential to become familiar with the phone call system and how it works.
- Now we have to draw a sequence diagram from what we have learned. First, power on the computer.
- When the computer is ready to use, we have to open EdrawMax software.
- After opening EdrawMax software, we have to create the sequence diagram.
- Finally, after drawing the sequence diagram, we have to ensure the sequence diagram is drawn correct.

### **Diagram Analysis:**

1. Call Initiation:
  - The Caller initiates the call by dialing the Recipient's phone number.
  - The Phone system establishes a connection between the two parties.
2. Call Connection:
  - The Phone system signals the Recipient with a ringing sound.
  - The Recipient answers the call by picking up the phone.
3. Conversation:
  - The Caller and Recipient engage in a conversation.
  - The Phone system facilitates the transmission of audio signals between the two parties.
4. Call Termination:
  - One of the parties hangs up the call (either the Caller or the Recipient).
  - The Phone system disconnects the call, ending the communication.



## Sequence Diagram:



**Lab Problem No.: 02**

**Lab Report Name:** Use Case Diagram of ATM System.

**Instrument:**

- Computer
- EdrawMax Software, etc.

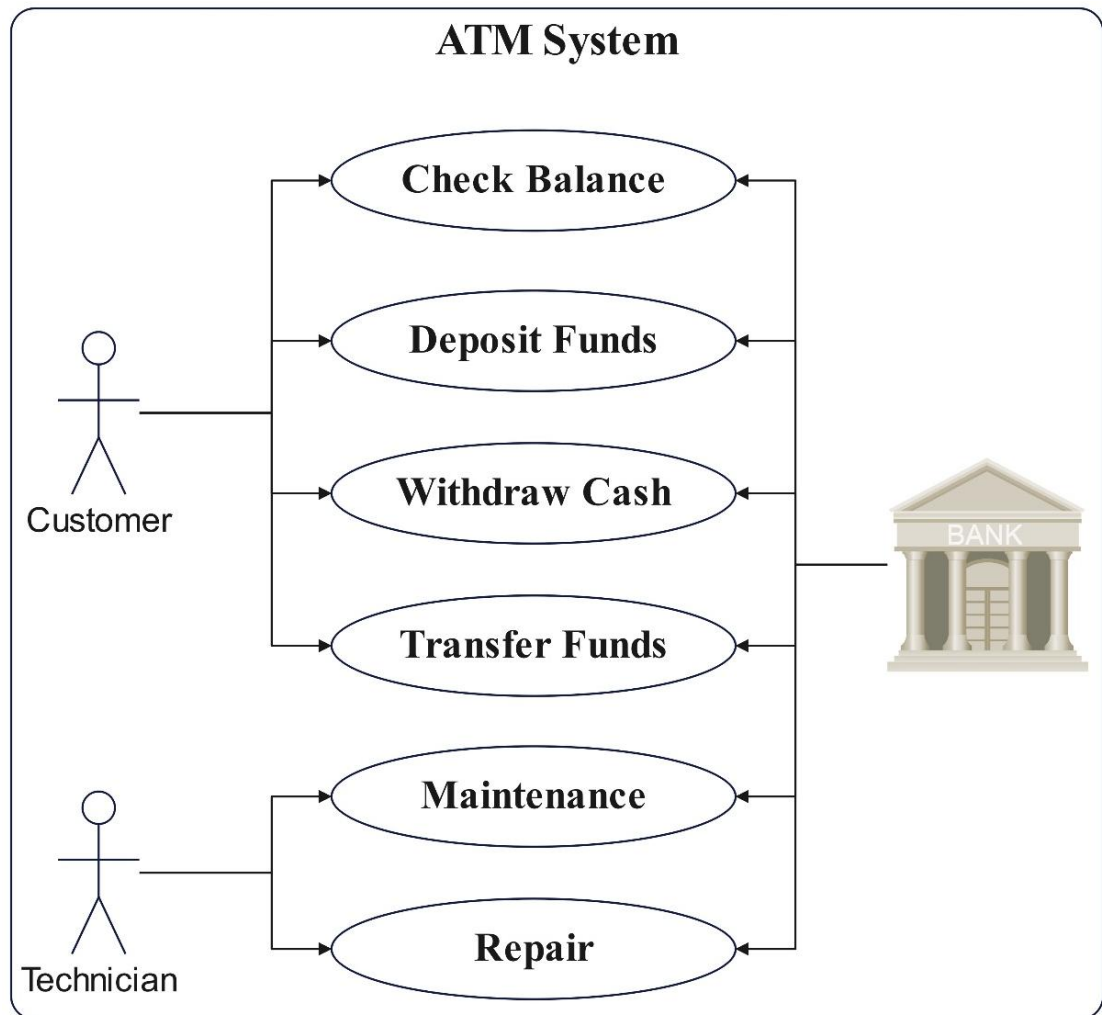
**Procedure:**

- At first, it is essential to become familiar with the ATM system and how it works.
- Now we have to draw a use case diagram from what we have learned. First, power on the computer.
- When the computer is ready to use, we have to open EdrawMax software.
- After opening EdrawMax software, we have to create the use case diagram.
- Finally, after drawing the use case diagram, we have to ensure the use case diagram is drawn correct.

**Diagram Analysis:**

1. Identify the actors (entities that interact with the system):
  - 1.1. Customer
  - 1.2. Technician
2. Identify the use cases (functions or services the system provides):
  - 2.1. Check Balance
  - 2.2. Deposit Funds
  - 2.3. Withdraw Cash
  - 2.4. Transfer Funds
  - 2.5. Maintenance
  - 2.6. Repair
3. Understand the relationships between actors and use cases.

**Use Case Diagram:**



## **Lab Problem No.: 03**

**Lab Report Name:** Class Diagram of a company management system.

### **Instrument:**

- Computer
- EdrawMax Software, etc.

### **Procedure:**

- At first, it is essential to become familiar with the management system of a company and how it works.
- Now we have to draw a class diagram from what we have learned. First, power on the computer.
- When the computer is ready to use, we have to open EdrawMax software.
- After opening EdrawMax software, we have to create the class diagram.
- Finally, after drawing the class diagram, we have to ensure the class diagram is drawn correct.

### **Diagram Analysis:**

#### **1. Company Class**

##### **1.1. Attributes**

- 1.1.1. name (string)
- 1.1.2. departments (Department)

##### **1.2. Methods**

- 1.2.1. addDepartment(department: Department)
- 1.2.2. removeDepartment(department: Department)
- 1.2.3. getDepartments(): Department
- 1.2.4. getDepartmentsCount(): int

#### **2. Department Class**

##### **2.1. Attributes**

- 2.1.1. name (string)
- 2.1.2. employees (Employee)
- 2.1.3. manager (Employee)
- 2.1.4. offices (Office)
- 2.1.5. company (Company)

## 2.2. Methods

- 2.2.1. addEmployee(employee: Employee)
- 2.2.2. removeEmployee(employee: Employee)
- 2.2.3. getEmployees(): Employee
- 2.2.4. getEmployeesCount(): int
- 2.2.5. getManager(): Employee
- 2.2.6. getOffices(): Office
- 2.2.7. removeOffice(office: Office)
- 2.2.8. setManager(manager: Employee)
- 2.2.9. setName(name: string)

## 3. Employee Class

### 3.1. Attributes

- 3.1.1. name (string)
- 3.1.2. department (Department)

### 3.2. Methods

- 3.2.1. getDepartment(): Department
- 3.2.2. getName(): string
- 3.2.3. setDepartment(department: Department)
- 3.2.4. setName(name: string)

## 4. Office Class

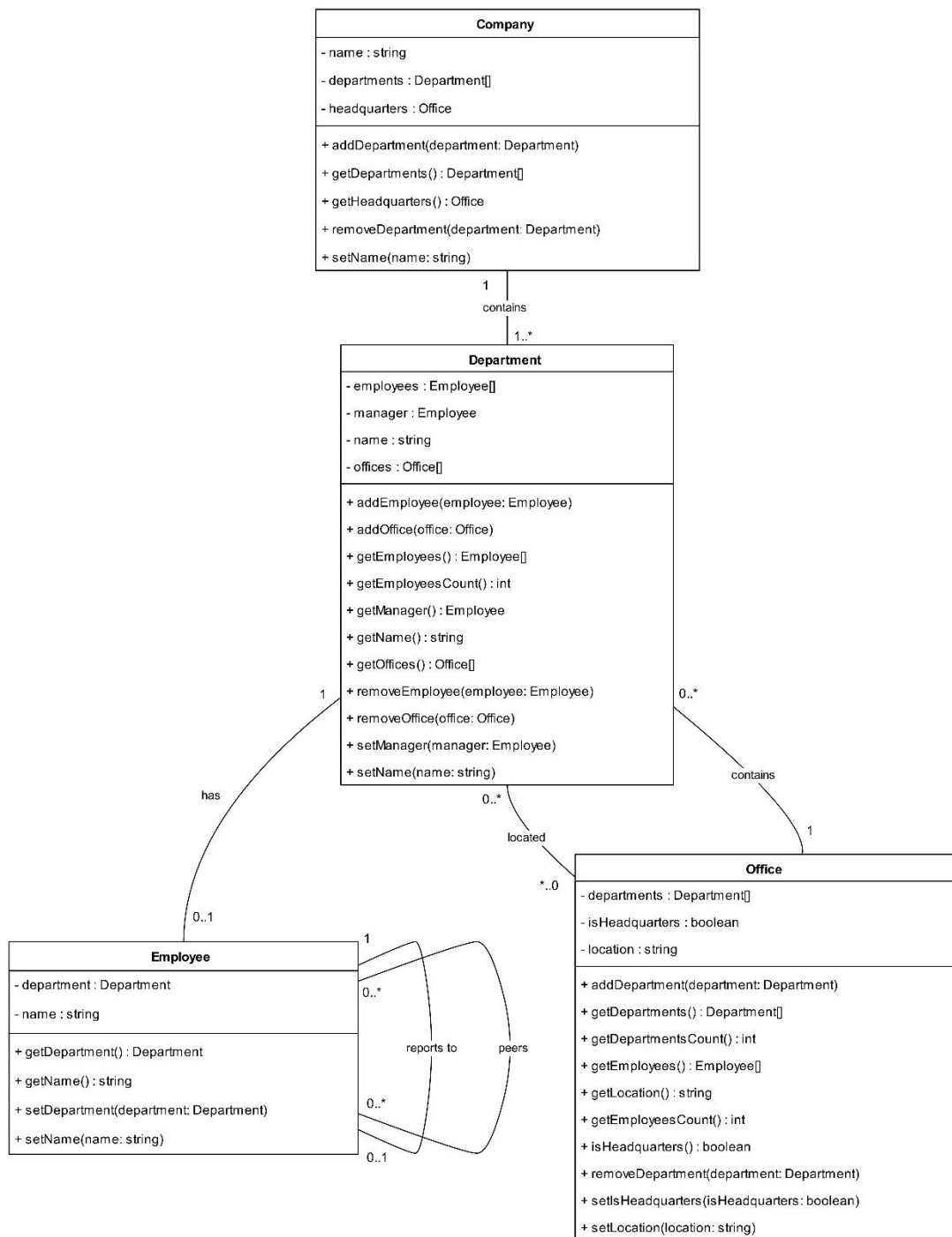
### 4.1. Attributes

- 4.1.1. department (Department)
- 4.1.2. location (string)
- 4.1.3. isHeadquarters (boolean)

### 4.2. Methods

- 4.2.1. addDepartment(department: Department)
- 4.2.2. getDepartments(): Department
- 4.2.3. getDepartment(): Department
- 4.2.4. getDepartmentsCount(): int
- 4.2.5. getLocation(): string
- 4.2.6. getEmployeesCount(): int
- 4.2.7. isHeadquarters(): boolean
- 4.2.8. setHeadquarters(isHeadquarters: boolean)
- 4.2.9. setLocation(location: string)

## Class Diagram:



## **Lab Problem No.: 04**

**Lab Report Name:** Level 1 Data Flow Diagram of an Airline Reservation System.

### **Instrument:**

- Computer
- EdrawMax Software, etc.

### **Procedure:**

- At first, it is essential to become familiar with the reservation system of an airline and how it works.
- Now we have to draw a level 1 data flow diagram (DFD) from what we have learned. First, power on the computer.
- When the computer is ready to use, we have to open EdrawMax software.
- After opening EdrawMax software, we have to create the level 1 data flow diagram.
- Finally, after drawing the level 1 data flow diagram, we have to ensure the level 1 data flow diagram is drawn correct.

### **Diagram Analysis:**

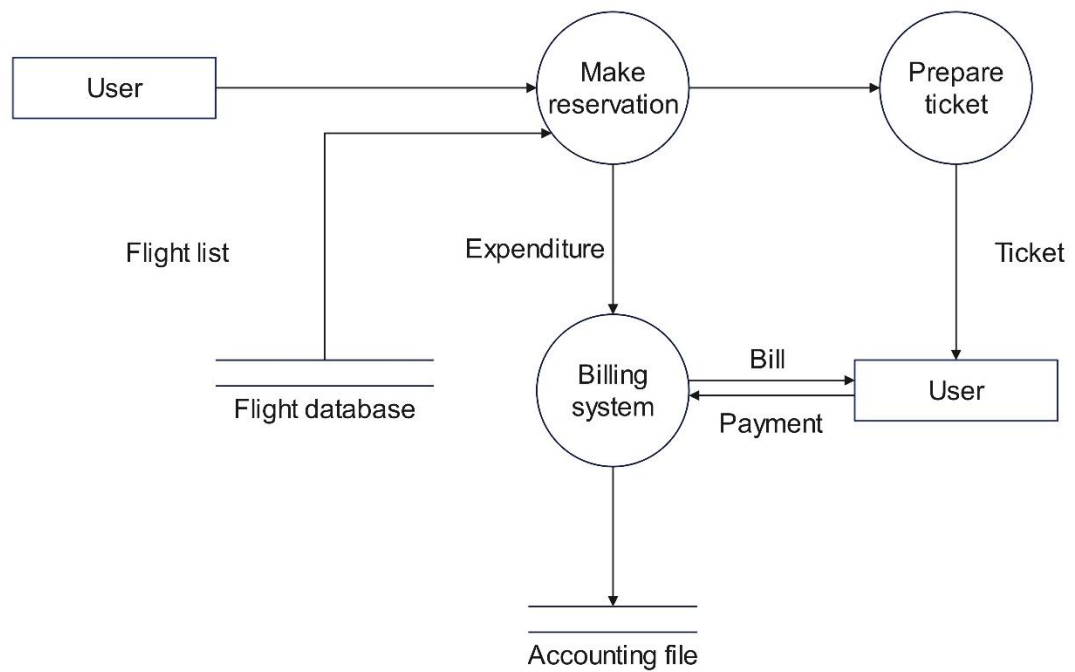
1. Diagram Analysis
  - 1.1. User: Represents the end-user who interacts with the system.
  - 1.2. Flight Database: Stores information about available flights.
2. Processes
  - 2.1. Make Reservation: Handles the process of booking a flight.
  - 2.2. Prepare Ticket: Generates the ticket for the booked flight.
  - 2.3. Billing System: Processes payments and generates bills.
3. Data Flows
  - 3.1. Flight List: Provides information about available flights to the user.
  - 3.2. Expenditure: Represents the cost of the flight.
  - 3.3. Ticket: The generated ticket for the booked flight.
  - 3.4. Bill: The bill generated after payment.
  - 3.5. Payment: The amount paid by the user.

### **Interpretation:**

- The system starts with the user requesting a flight list.

- Based on the list, the user makes a reservation.
- The system then prepares the ticket and sends it to the user.
- The billing system processes the payment and generates a bill.
- The accounting file is updated with the transaction details.

**Level 1 Data Flow Diagram:**





## **Lab Problem No.: 05**

**Lab Report Name:** Use Case Diagram of a Hotel Management System.

### **Instrument:**

- Computer
- EdrawMax Software, etc.

### **Procedure:**

- At first, it is essential to become familiar with the management system of a hotel and how it works.
- Now we have to draw a use case diagram from what we have learned. First, power on the computer.
- When the computer is ready to use, we have to open EdrawMax software.
- After opening EdrawMax software, we have to create the use case diagram.
- Finally, after drawing the use case diagram, we have to ensure the use case diagram is drawn correct.

### **Diagram Analysis:**

1. Actors: The individuals or entities that interact with the system.
  - 1.1. Waiter: Responsible for receiving and placing orders.
  - 1.2. Client: Places orders for food and wine, and pays for the service.
  - 1.3. Chef: Prepares and cooks the ordered food.
  - 1.4. Cashier: Handles payment for the food and wine.
2. Use Cases: The specific functionalities or services provided by the system.
  - 2.1. Order Food: The waiter receives and places an order for food.
  - 2.2. Confirm Order: The system confirms the order with the client.
  - 2.3. Cook Food: The chef prepares and cooks the ordered food.
  - 2.4. Serve Food: The waiter serves the prepared food to the client.
  - 2.5. Order Wine: The client orders wine along with the food.
  - 2.6. Serve Wine: The waiter serves the ordered wine to the client.
  - 2.7. Eat Food: The client consumes the food.
  - 2.8. Drink Wine: The client consumes the wine.
  - 2.9. Pay for Food: The client pays for the food.
  - 2.10. Pay for Wine: The client pays for the wine.

- 2.11. Accept Payment: The cashier accepts the payment.
- 3. Relationships: The connections between actors and use cases.
  - 3.1. Communication: Actors communicate with the system through use cases.
  - 3.2. Extension: Optional behaviors that can be added to a use case, such as ordering wine.
- 4. Extensions: Optional behaviors that can be added to a use case.
  - 4.1. Serve Wine
  - 4.2. Drink Wine
  - 4.3. Pay for Wine
- 5. Functions:
  - 5.1. Order Management: The system handles food and wine orders from clients.
  - 5.2. Food Preparation: The kitchen staff prepares and cooks the ordered food.
  - 5.3. Wine Service: If wine is ordered, it is served to the client.
  - 5.4. Payment Processing: The cashier handles payment for the food and wine.

**Use Case Diagram:**

