**CMP 403 1st TEST**

**TEST QUESTIONS**

1. Compare and Contrast Plan-driven methods and Agile methods.

2. Write the requirements for the four software projects

**GROUP MEMEBERS**

1. AIYEWUMI DAVID OLUWOLE BHU/20/04/09/0011
2. ANOINTING EBUBE DAUDA BHU/20/04/05/0032
3. ORJI CHINONYEREM PEACE BHU/20/04/09/0007
4. CHUKWU DANIEL NONSO BHU/20/04/05/0010
5. AROMOLARAN ADENIKE ELIZABETH BHU/20/04/05/0079
6. JOLLY D JOSEPH BHU/20/04/05/0079
7. ABRAHAM BENJAMIN ORUME BHU/20/04/05/0145
8. AMBINKANME MBOSIRE MUSA BHU/20/04/05/0040
9. MAIGIDA GYENOM TALATU BHU/20/04/05/0013
10. DANJUMA KUYET DIVINE BHU/20/04/05/0011

**ANSWERS**

**Q1. Comparison Between Plan-driven and Agile methods**

a. Both plan-driven and agile methods aim to deliver value to the customer. In both cases, the end goal is to create software that meets the needs of users.

b. Both approaches involve some form of requirement analysis. Plan driven methods typically gathered and documented upfront, while agile methods may gather requirements incrementally throughout the project.

c. Both methodologies recognize the importance of effective communication and teamwork within the development team.

d. Both methodologies recognize that changes can occur during the development process.

e. Both methodologies prioritize testing and quality assurance

f. Plan-driven follows the sequential approach and agile follows the iterative and incremental approach.

**Q2. Write the requirements for the four software projects**

**Project 1 - Employee Management System (EMS)**

**User Requirements**

1. Users (Admins and Employee) should be able to access the system through secure authentication.

2. Admins should have the capability to add and edit employee profiles

3. Employees should be able to check their leave status, qualification, promotion history, and yearly holiday list.

**System Requirements**

1. It should provide features for leave management

2. The system should have a database to store employee information and payroll data.

3. The system must have secure user authentication mechanisms

**Functional Requirements**

1. The system should support the submission and tracking of grievances and resignations

2. Admins should be able and modify employee details, like name, job

3. Admins should be able to access and modify salary details

4. Employees should have a dashboard to check their leave status, qualifications and promotion history.

**Non-Functional Requirements**

1. User-friendly

2. Secure and Privacy

3. High availability

**Project 2 - Payroll Management System**

**User Requirements**

1. Admins should be able to login and track employee payroll and attendance

2. Employee should have access to their own attendance and payroll details

**System Requirements**

1. The system should provide real-time attendance tracking.

2. It should enable admins, to manage employee profiles and salary details

3. It should support secure user authentication for admins and employees

**Functional Requirements**

1. Admins should be able to track attendance data in real-time

2. Employees should be able to access their attendance and payroll information

**Non-Function Requirements**

1. The system should be highly available

2. It should be responsivve and user-friendly

3. The system should have data security and privacy

**Project 3 - Fingerprint-Based ATM Card**

**User Requirements**

1. Users should be able to login using their fingerprint for authentication

2. Users should be able to perform various banking transaction including cash withdrawal, money transfer, and balance inquires.

**System Requirements**

1. The system should capture and verify user fingerprints for authentications

2. The system should allow various banking transactions and account balance checks

3. It should store and retrieve the last 5 transactions for each users

4. It should support secure user pins for additional security.

**Functional Requirements**

1. The system should display the last 5 transactions for users

2. Users should be able to conduct various banking transaction

3. Users should be able to login using their fingerprint and PIN

**Non-functional Requirements**

1. The system should ensure high security and data privacy

2. It should be available 24/7

3. It provides fast and reliable fingerprint recognition

**Project 4 - Android Local Ticketing System**

**User Requirements**

1. Users should be able to login using admin or user account

2. users should have access to a ticket booking form

3. Admins should be able to recharge user account balances and view processed tickets

**System Requirements**

1. The system should secure user authentication for admin and user accounts

2. It should provide a user-friendly ticket booking form

3. The system should store user account balances and ticket data

**Functional Requirements**

1. Users should be able to select source, destination, and travel class and journey type in the booking form

2. Users should be able to book and receive online tickets

3. Admins should be able to recharge user account balances and view processed tickets

**Non Functional Requirements**

1. The system should ensure data security and privacy

2. It should be user-friendly and responsive

3. It should be available 24/7 for ticket booking and account management.

**CMP 403 2nd TEST**

**TEST QUESTION**

A. Draw the below diagram our above 4 software projects

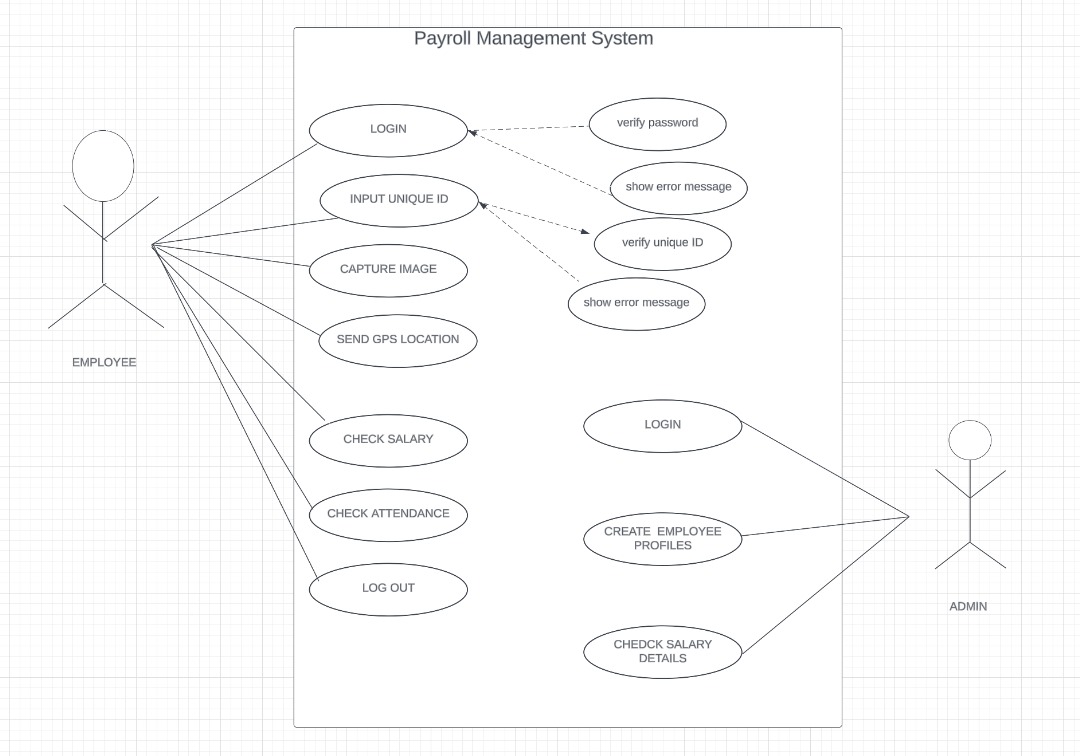
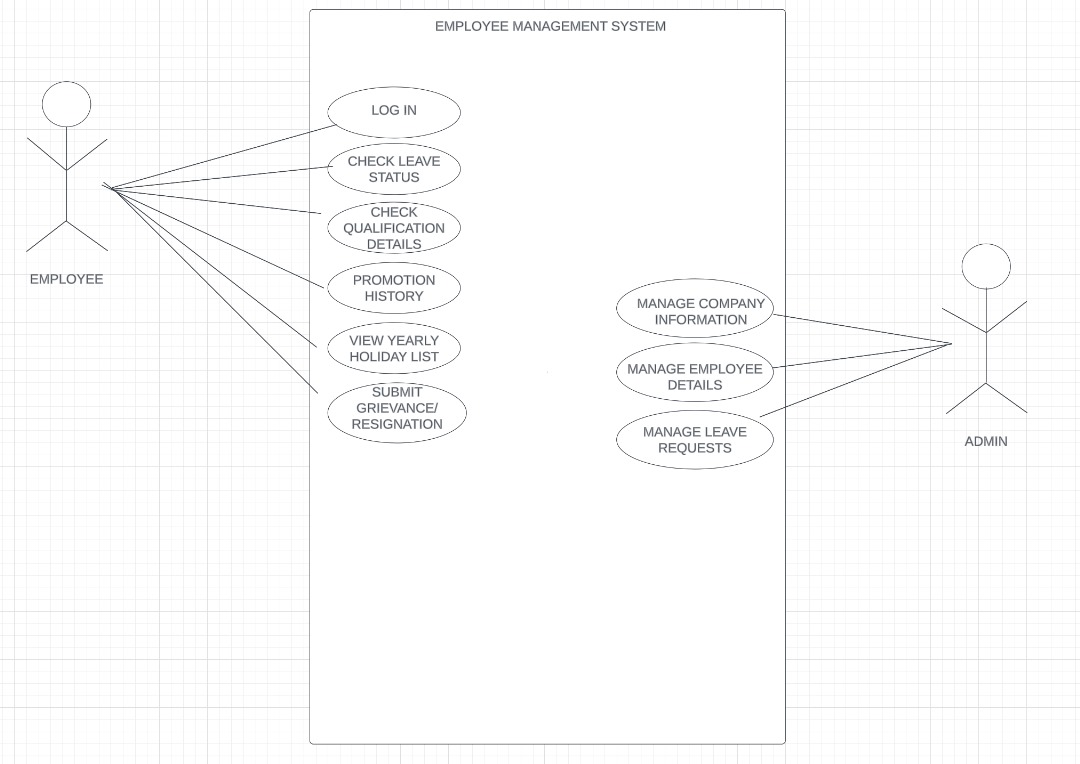
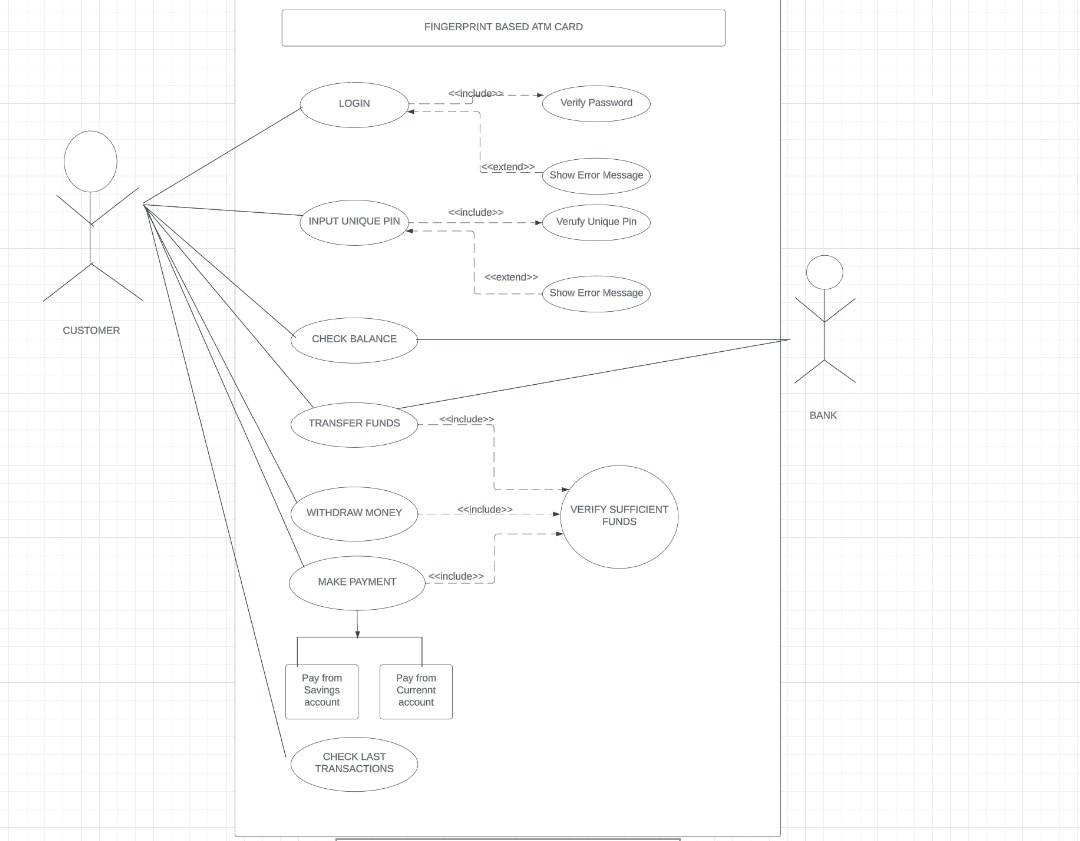
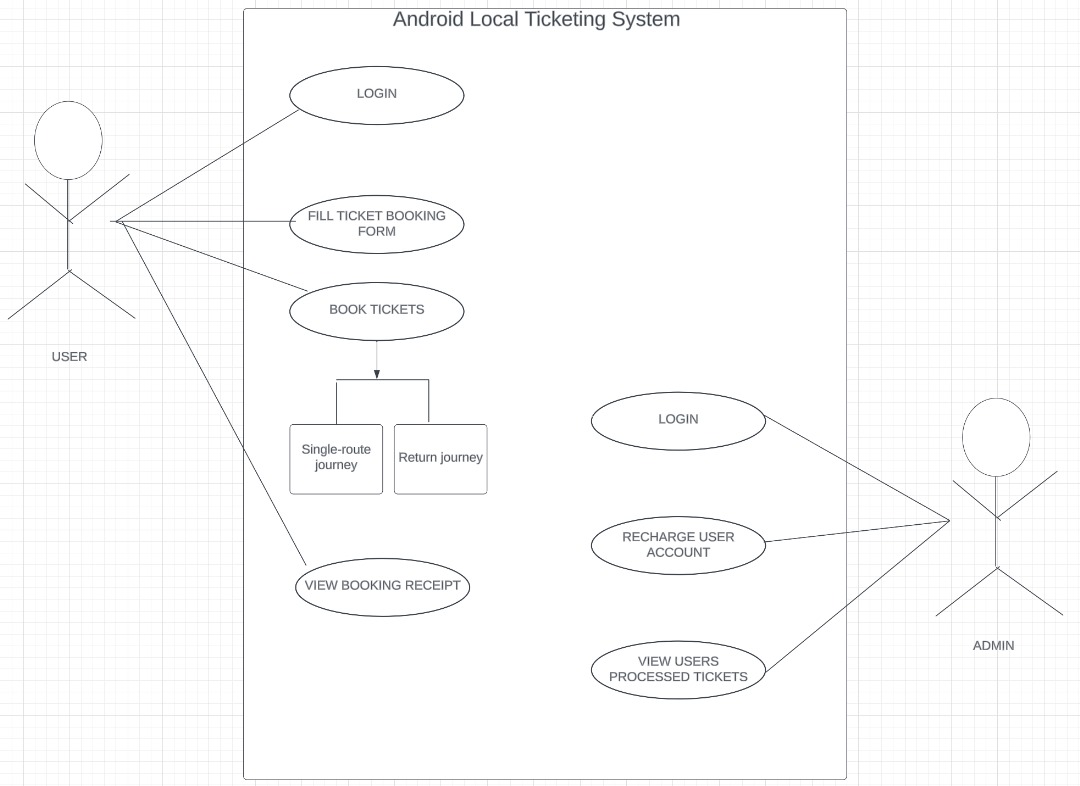
1. use case diagram
2. sequence diagram
3. activity diagram
4. class diagram
5. entity relationship diagram

B. What is the purpose of system modelling

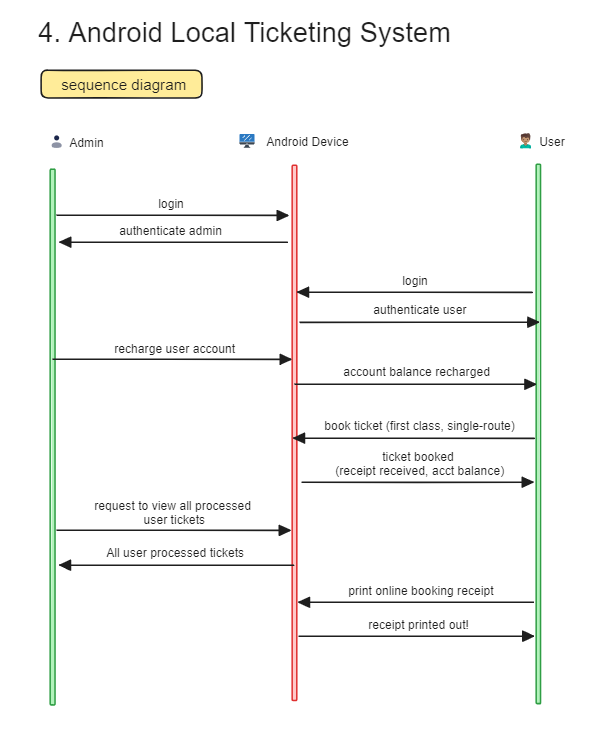
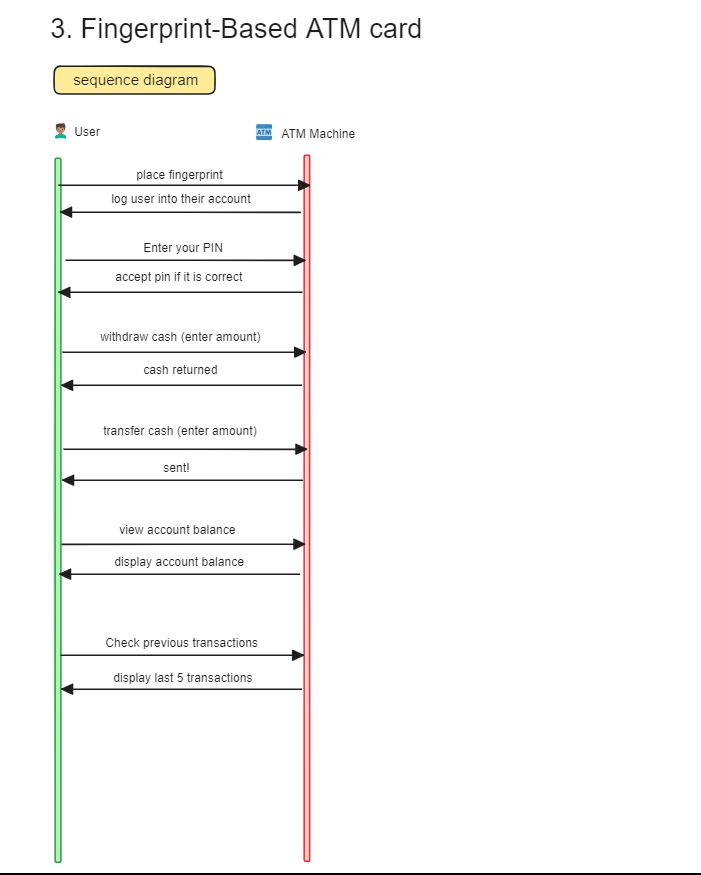
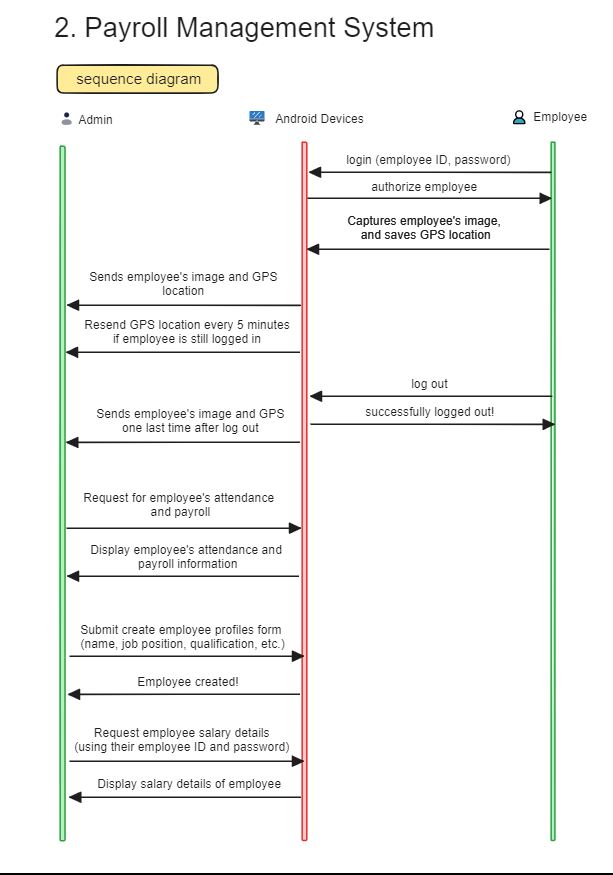
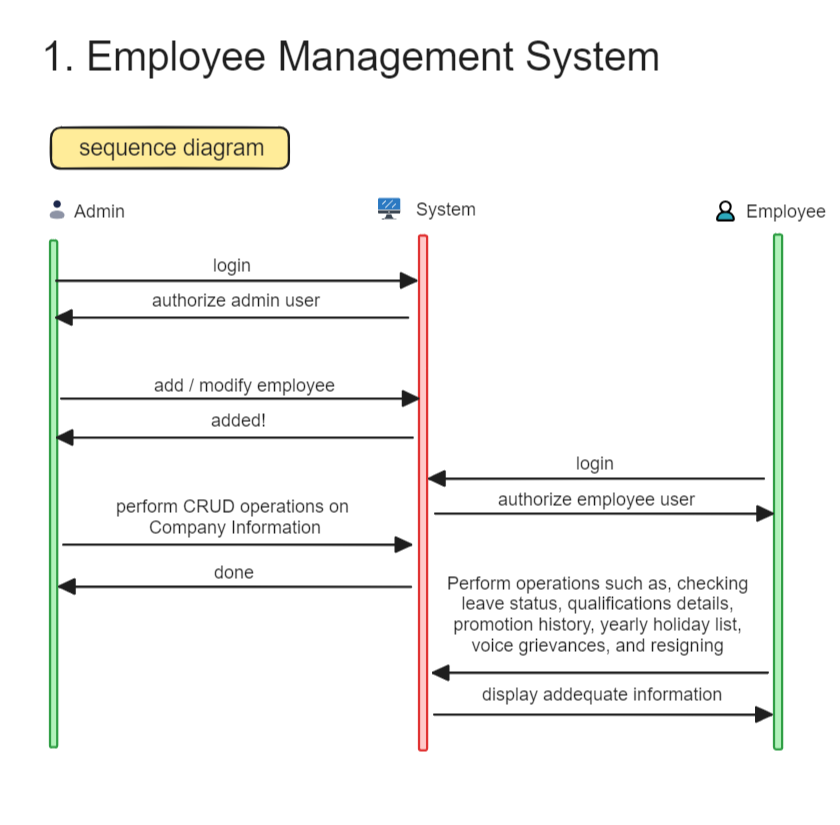
**ANSWERS**

**A. Draw the below diagram our above 4 software projects**

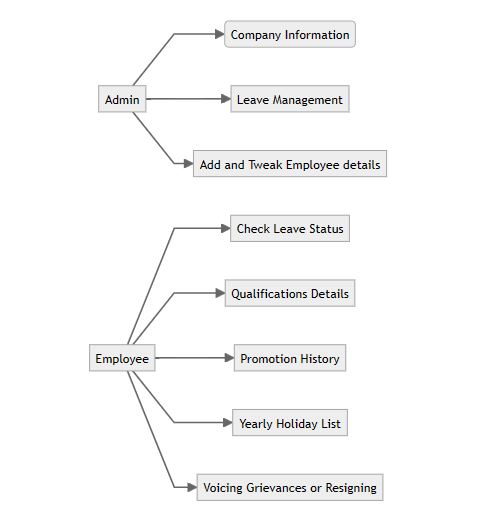
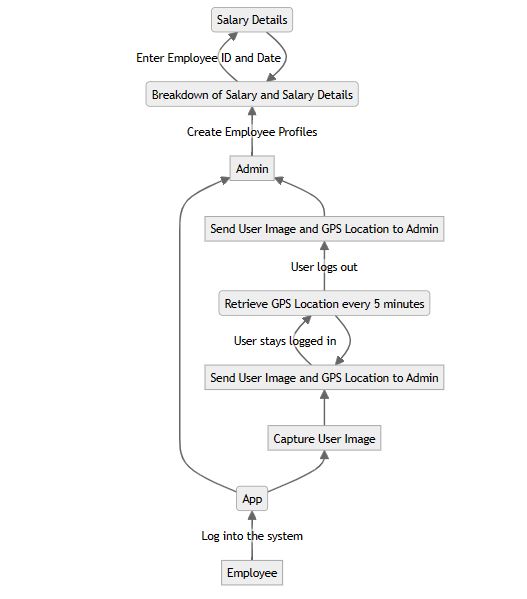
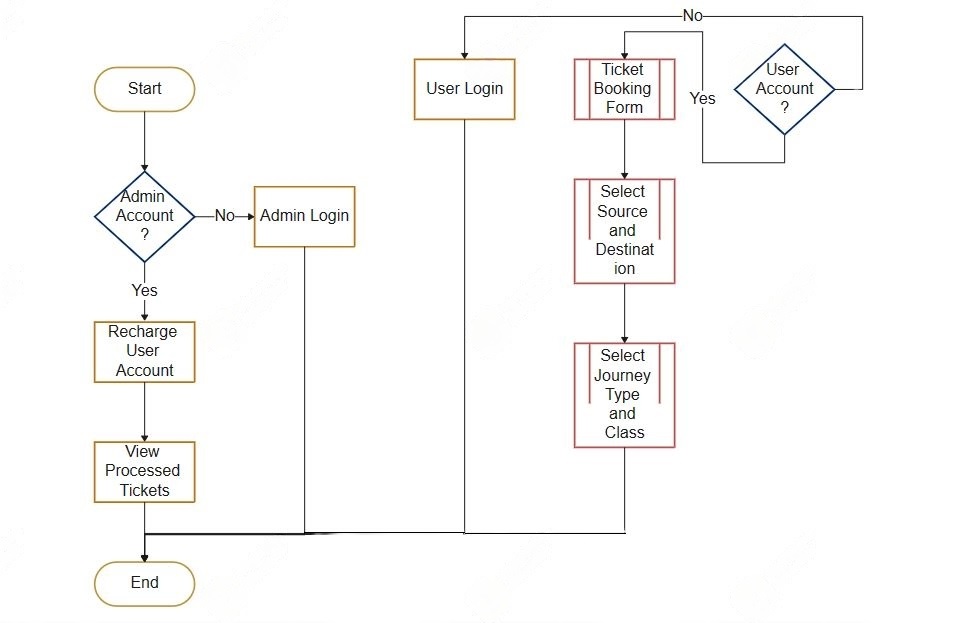
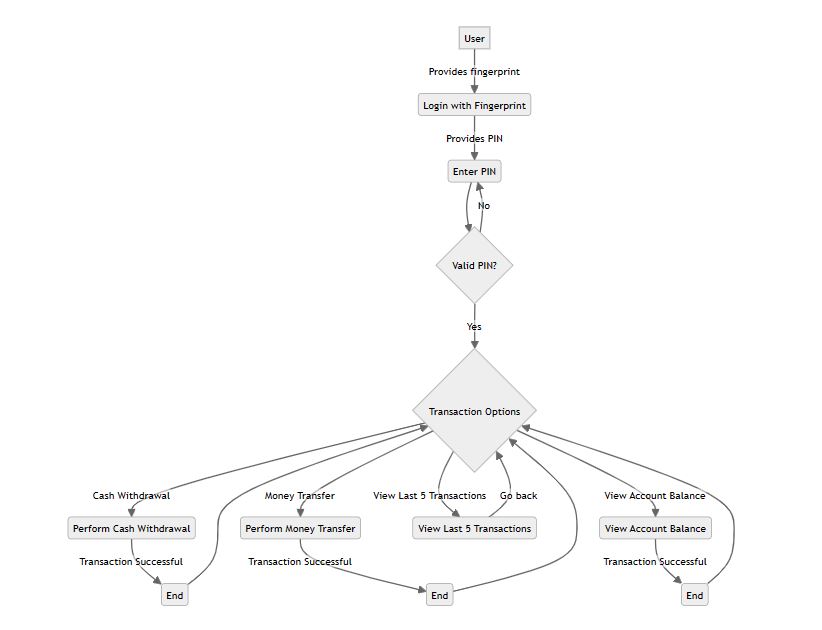
**1. Use case diagram**



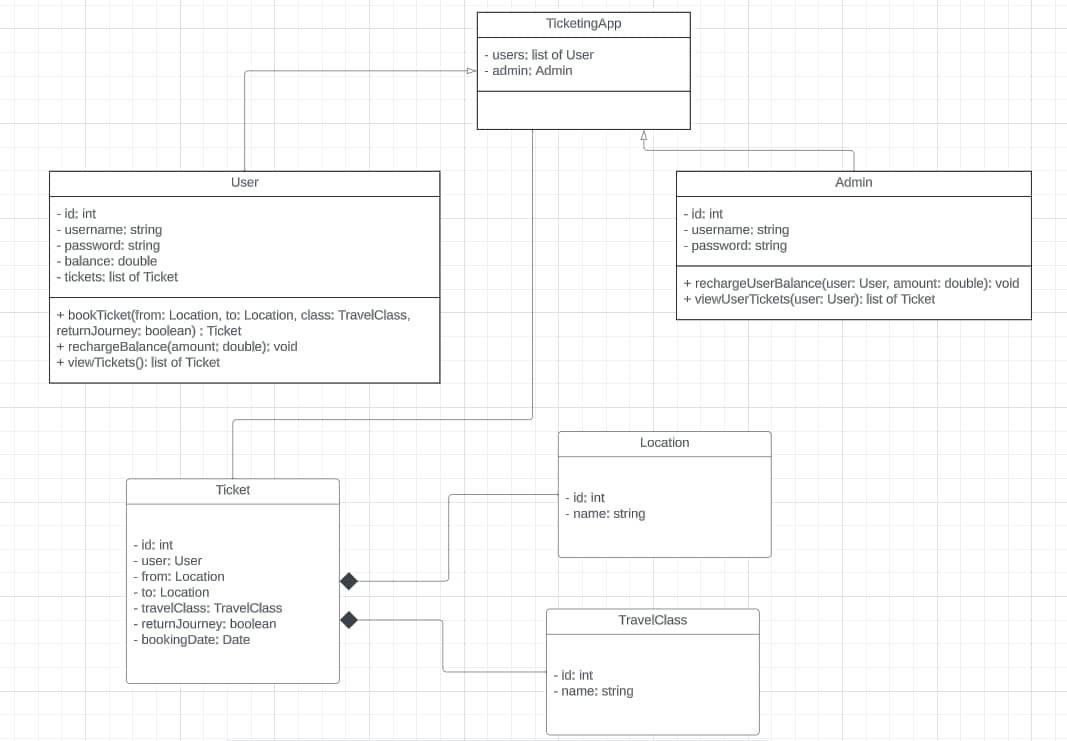
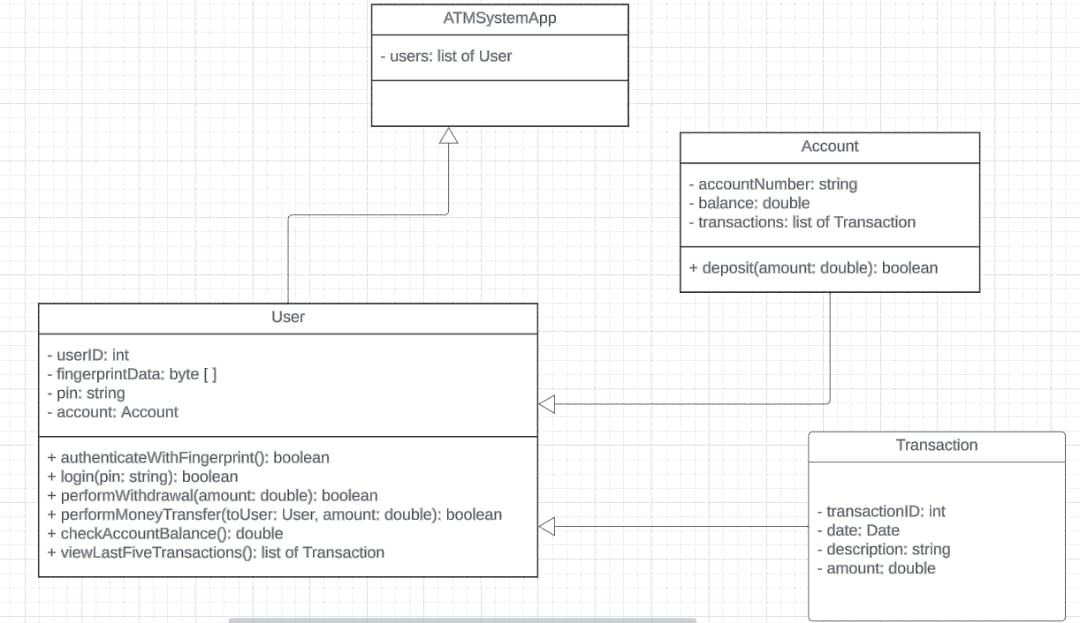
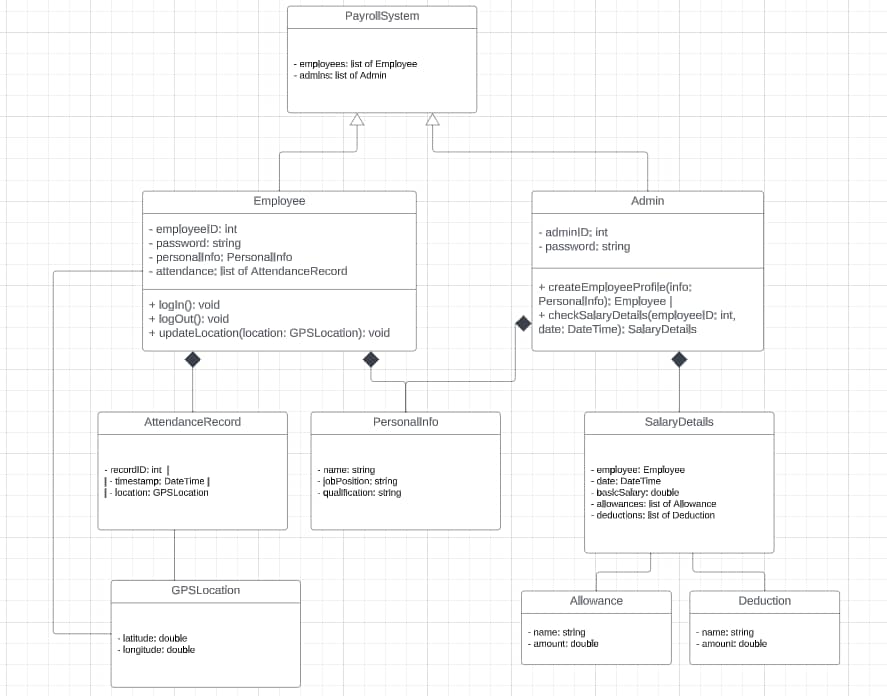
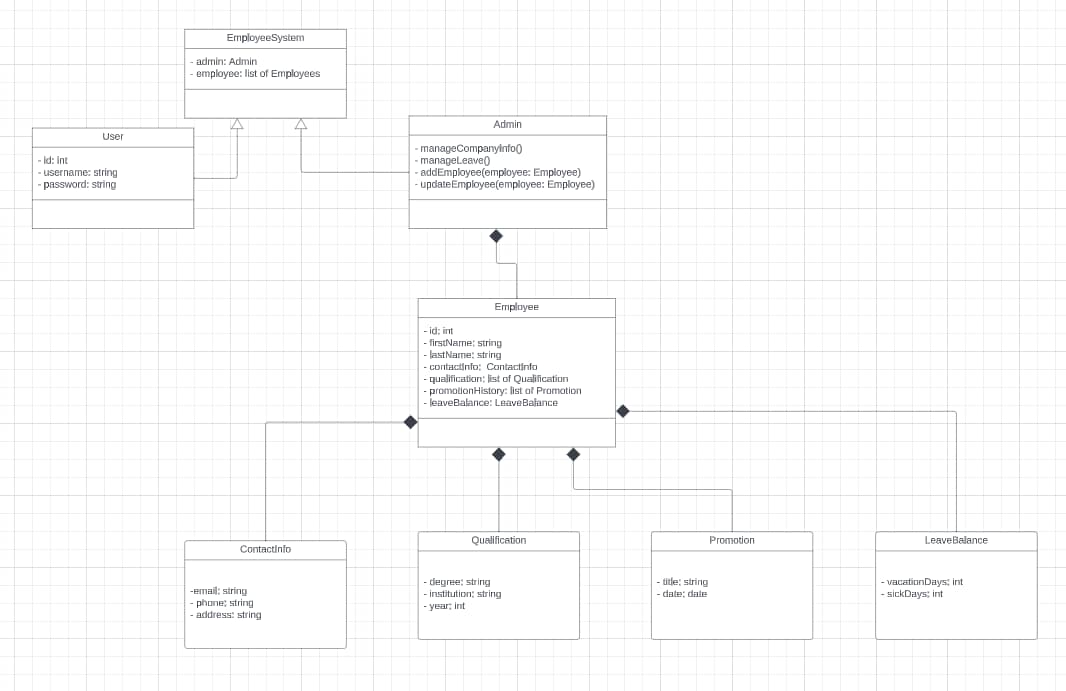
**2. Sequence diagram**



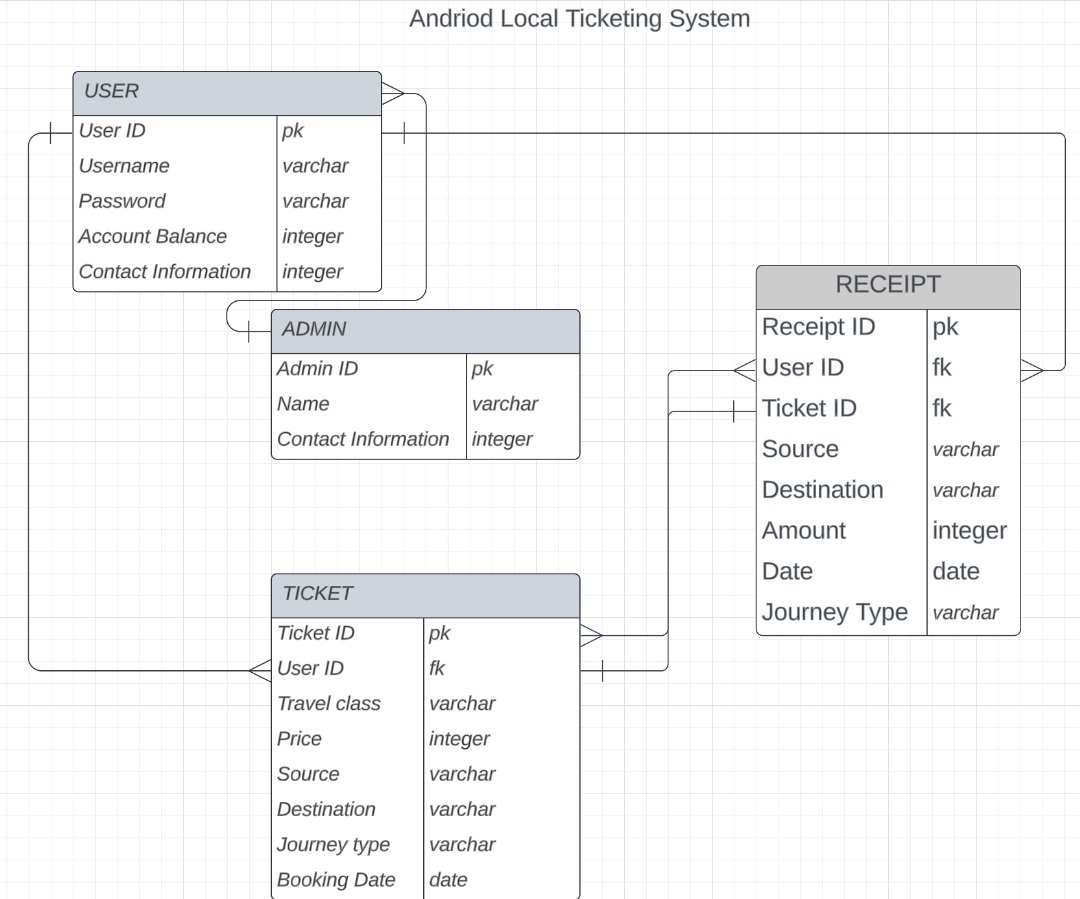
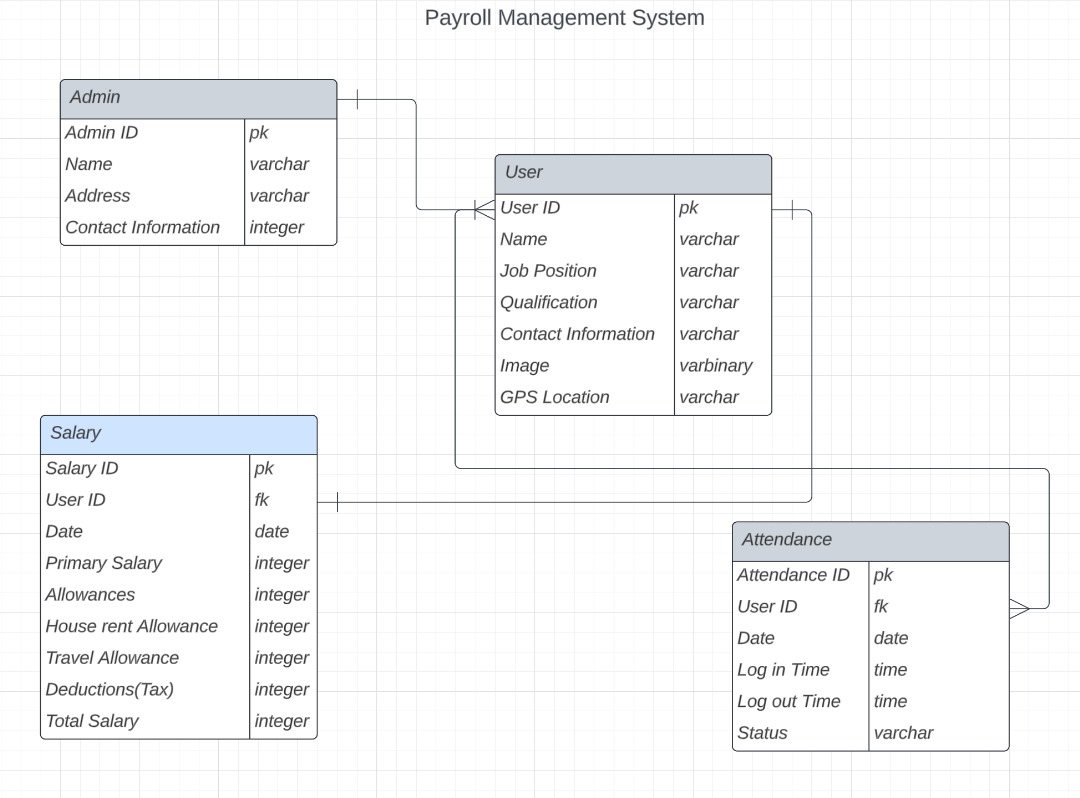
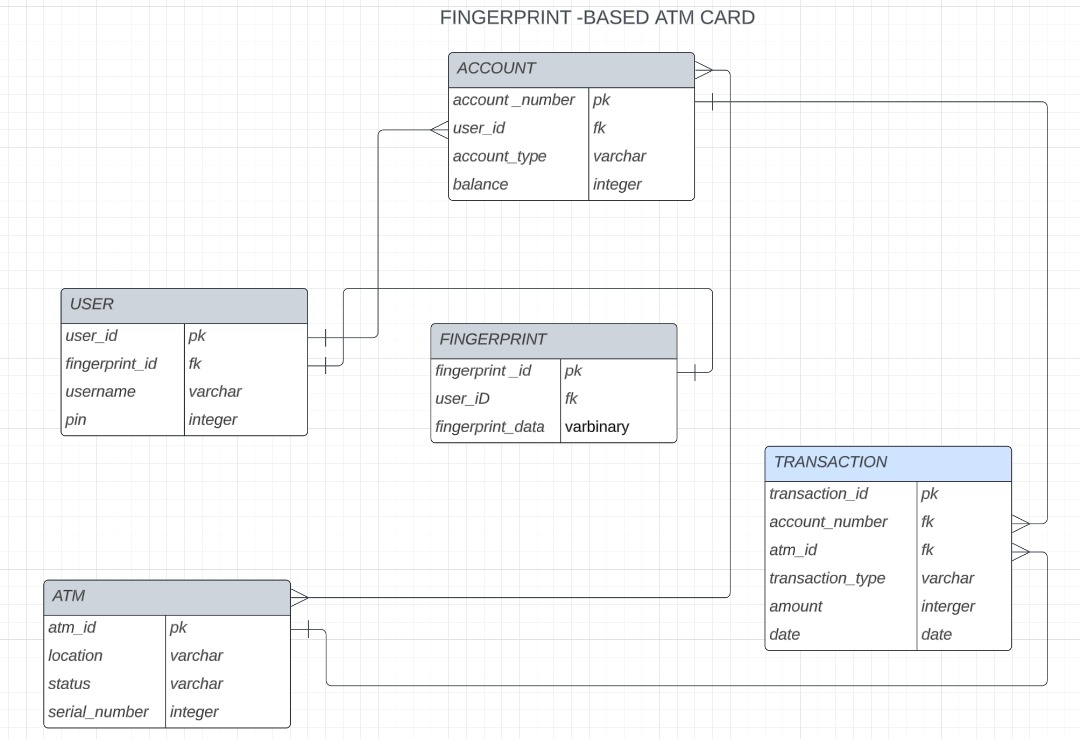
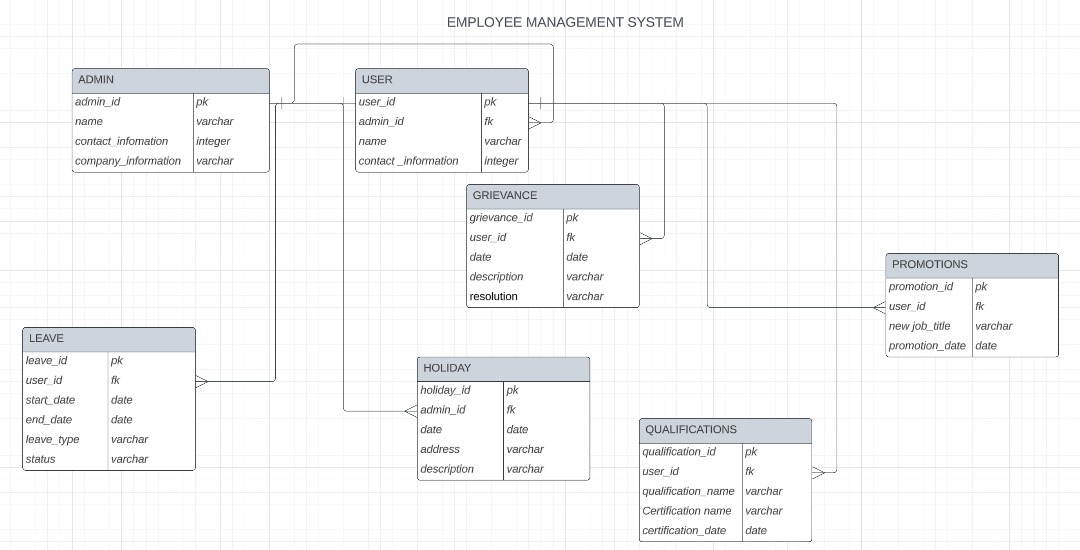
**3. Activity diagrams**



**4. Class diagram**



**5. Entity relationship diagram**



**B. What is the purpose of system modeling**

The purpose of system modeling is to create simplified representations of a system in order to better understand, analyze, and communicate its various aspects and components. System modeling serves several important purposes:

**Visualization:** System models provide a visual representation of a system's components, relationships, and interactions. This makes it easier for stakeholders to grasp the system's structure and behavior.

**Analysis:** System models allow for the analysis of system performance, behavior, and characteristics. By creating mathematical or computational models, it becomes possible to simulate and analyze how the system will behave under different conditions, inputs, or changes.

**Design and Development:** System modeling is a crucial step in the design and development process. It helps in the specification of system requirements, architecture, and interfaces. Designers can use models to explore and refine their ideas before implementation.

**Communication:** Models provide a common language for communication among stakeholders, including developers, designers, project managers, and users. They help bridge the gap between technical and non-technical team members.

**Problem Solving:** Modeling can assist in problem-solving and decision-making. By representing the system's current state and potential changes, it becomes easier to identify issues, propose solutions, and evaluate the impact of those solutions.

**Documentation:** Models serve as documentation that captures the knowledge about the system. This is valuable for future reference, maintenance, and system evolution.

**Risk Management:** Modeling can help identify potential risks and uncertainties in a system. By exploring different scenarios and conducting sensitivity analysis, organizations can better prepare for contingencies.

**Cost Reduction:** Modeling can reduce costs by allowing for early detection and correction of design flaws or issues. This can prevent costly changes during the implementation phase.

**Requirements Validation:** System models can be used to validate system requirements. By simulating the model, it can be determined if the system meets the specified criteria.

Improving Efficiency: Through modeling, it's possible to optimize system performance and resource utilization. This can lead to efficiency improvements and resource savings.