Darshan University

A Project Report on

“Bank management system”

Under the subject

Software Engineering (2301CS405)

B. Tech, Semester –IV

Computer Science & Engineering Department

Submitted By

Student Name: Darshan Padsumbiya Enrollment No.: 23010101181

Academic Year

(2024-2025)

Internal Guide	Dean-DIET
Prof.R. B. Gondaliya	Dr. Gopi Sanghani
Darshan University	Darshan University

Computer Science & Engineering Department
Darshan University

DECLARATION

We hereby declare that the SRS, submitted along with the **Software Engineering (2301CS405)** for entitled "**Bank management system**" submitted in partial fulfilment for the Semester-4 of **Bachelor Technology (B. Tech)** in **Computer Science and Engineering (CSE)** Department to Darshan University, Rajkot, is a record of the work carried out at **Darshan University, Rajkot** under the supervision of R. B. Gondaliya and that no part of any of report has been directly copied from any students' reports, without providing due reference.

Darshan Padsumbiya

Student's Signature

Date: _____

Computer Science & Engineering Department
Darshan University

CERTIFICATE

This is to certify that the SRS on "Bank Management System" has been satisfactorily prepared by Darshan Padsumbiya (23010101181) under my guidance in the fulfillment of the course Software Engineering (2301CS405) work during the academic year 2024-2025.

Internal Guide	Dean-DIET
Prof. R. B. Gondaliya	Dr. Gopi Sanghani
Darshan University	Darshan University

ACKNOWLEDGEMENT

I wish to express my sincere gratitude to my project guide Prof. R. B. Gondaliya and all the faculty members for helping me through my project by giving me the necessary suggestions and advices along with their valuable co-ordination in completing this work.

I also thank my parents, friends and all the members of the family for their precious support and encouragement which they had provided in completion of my work. In addition to that, I would also like to mention the Darshan University personals who gave me the permission to use and experience the valuable resources required for the project from the University premises.

Thus, in conclusion to the above said, I once again thank the faculties and members of **Darshan University** for their valuable support in completion of the project.

Thanking You

Darshan Padsumbiya

ABSTRACT

The Bank Management System (BMS) is a software application designed to automate and streamline various banking processes, such as customer account management, transactions, loan management, deposits, withdrawals, and reporting. With the increasing demand for improved efficiency, security, and customer satisfaction in the banking industry, BMS provides an essential framework for modern banks to handle a wide range of banking operations seamlessly. As the global financial sector embraces digital transformation, the need for efficient and scalable systems that ensure smooth operations while reducing human error has never been greater.

A Bank Management System centralizes core banking functions in a single platform, offering a secure and user-friendly interface for both customers and employees. The system automates manual tasks, thereby reducing operational costs, saving time, and increasing overall productivity. By integrating essential functions such as account management, loan disbursements, interest calculations, transaction processing, and reporting into one system, BMS allows banks to deliver a seamless banking experience to their customers across various platforms, including online banking, mobile banking, and ATM networks.

Main purpose of this system is to reduce human efforts as much as possible.

Table of Contents

List of Figures

List of Tables

1 Introduction

1.1 Product perspective

A Bank Management System (BMS) is a software solution designed to automate, manage, and streamline the complex tasks involved in banking operations. In an era where technology plays an indispensable role in every sector, the financial industry, including banks, has adopted advanced software solutions to facilitate better customer service, ensure secure transactions, and improve overall operational efficiency. The Bank Management System, as a critical part of the banking infrastructure, serves as the backbone of a bank's internal systems, assisting in day-to-day banking operations and in managing the wide variety of services that banks offer.

The need for efficient, secure, and scalable management systems in banks has never been more pressing. With the advent of online banking, mobile applications, and increased regulatory scrutiny, the pressure to manage vast amounts of sensitive financial data while maintaining the highest levels of security and customer satisfaction has grown exponentially. A Bank Management System addresses these challenges by providing automated, integrated solutions that allow banks to deliver quality services to their customers while ensuring operational efficiency, security, and compliance.

1.2 Key Features and Functionalities of a Bank Management System

A Bank Management System typically includes a range of features aimed at simplifying the management of various banking operations. Some of the key functionalities include:

1. **Customer Management**: One of the fundamental features of a Bank Management System is the management of customer data. This involves storing and retrieving information such as customer profiles, transaction history, account types, and personal details. The system ensures that sensitive customer data is managed securely, in compliance with data privacy regulations.
2. **Account Management**: The system allows for the creation, modification, and deletion of accounts. This includes saving accounts, checking accounts, fixed deposits, and more. Bank employees can manage account information such as balance, transaction history, interest calculations, and other pertinent details.
3. **Transaction Management**: A Bank Management System handles various types of financial transactions, such as deposits, withdrawals, transfers, and bill payments. The system automates the transaction process, reducing the risk of human error and ensuring that all transactions are accurately recorded in real-time.
4. **Loan Management**: The system helps manage the loan application process, loan disbursements, repayments, and interest calculation. It also allows banks to track loan status and customer payments, and maintain a record of all loan-related information.
5. **Deposit and Withdrawal Processing**: The system manages deposit and withdrawal transactions for both customers and non-customers. Automated processing of deposits and withdrawals ensures accuracy, and reporting features help maintain a record of all activities.

6. **Interest Calculation**: For savings accounts, fixed deposits, and other interest-bearing accounts, the Bank Management System automatically calculates interest based on pre-configured rates. This feature enhances accuracy and saves time for both customers and bank staff.
7. **Reporting and Analytics**: The system generates reports on account activity, loan status, interest calculations, and other financial operations. This enables bank management to track performance, assess financial health, and make informed decisions. Analytics tools also help in identifying trends and improving customer service.
8. **Security Features**: Security is a paramount concern in banking, and Bank Management Systems are designed with multiple layers of protection. These include encryption for secure data storage, multi-factor authentication for user access, and audit trails for tracking user activities. Regular updates and compliance with security standards are also integral to ensuring that the system remains secure.
9. **Integration with External Systems**: A Bank Management System must integrate with various external systems, such as payment gateways, ATMs, and online banking platforms. This allows the bank to offer services like internet banking, mobile banking, and seamless transactions across various channels.
10. **Regulatory Compliance**: Compliance with national and international financial regulations is essential in banking. The Bank Management System helps ensure that all operations, including financial reporting, are in line with regulatory requirements. Features like anti-money laundering (AML) and know-your-customer (KYC) checks are often integrated into the system to help banks comply with legal standards.

1.3 Functional Requirement

1.3.1 Customer

- See Account Details:Customers should see their account details.
- Apply for documents: Customer can apply for documents like statement, chequebook etc.
- Login: Customer should login by their A/C No, Mobile No or Email.
- Online payment:Customer should do online payments from system.
- Apply for card renew: Customer should apply for card renew.
- View statement: Customer should view statements of their account.
- Change personal details: Customer should change their personal details like phone No, email, address, etc.
- Apply for loans: Customer should view and apply for loans.
- Pay credit card bills: Customer should view credit card purchases and Pay bill in the system.
- Change passwords and pin: Customer should change net banking password and ATM pin online.
- Track cheque status: Customer should track cheque status by cheque No.
- Stop payment of cheque: Customer should request for stop payment of any given cheque.
- Invest money in FDs: Customer should view and invest in different FDs.
- Apply for lockers: Customer should apply for bank locker service
- Disable payment on card: Customer should disable payment service on card.
- Open new account: Customer should open new account.

1.3.2 Bank staff

- Login: Staff members should login in system by their id's.
- Access of account:Casher should deposit and credit amount of account.
- Apply for leave: Staff should apply for leave in system.
- View daily cash collection: Casher should view daily cash collection of particular branch.

- View loan application: loan department should view loan application and documents.
- Search customers: Staff should search customers by their name, AC No and mobile No
- View document applications: Staff members should view that Customers would apply for which documents
- Approve document application: Staff members should approve document application of customer.
- Approve loan application: Loan department should approve loan application.
- View locker application: Locker department should view locker applications of customers.
- View list of securities: Loan department should view list of securities was given by loan takers.
- Open new account: Staff member should open a new account of person.
- Close account: Staff member should remove existing account.

1.3.3 Manager

- Login: Manager should login in system by their id's.
- View leave applications: Managers should view leave application of their respective department.
- Grant leave: Manager should grant a leave of staff member.
- View daily collection: Manager should view daily cash collection of branches.
- View loan details: Managers should view given loan details and their document.
- View Details of ATMs: Managers should view details of ATMs of their respective branch area.
- View details of loan EMIs: Managers should view EMI details of loan taker. Weather they are paying EMIs on time or not.
- View details of credit card bills: Managers should view pending and paid bill details of credit cards.
- View details of FDs: Managers should view that how many customers are Inves
- ting in FDs.
- View performance of employee: Managers should view overall performance of their staff.

1.3.4 Admin

- View performance branch: Admin should view details of branches in their region.
- Chek the cashflow: Admin should view overall cashflow of bank.
- View total assets: Admin should view total assets of the bank.
- View investment details: Admin should view investment details of the bank that bank are investing money in which bonds, schemes, stocks, etc.
- View performance of managers: Admin should view performance of branch manager.
- Promote manager: Admin should give promotion of the managers.

1.4 Non-Functional Requirement

1.4.1 Usability:

- The UI should be simple enough for everyone to understand and get the relevant information without any special training. Different languages can be provided based on the requirements.

1.4.2 Accuracy:

- The data stored about the books and the fines calculated should be correct, consistent, and reliable.

1.4.3 Availability:

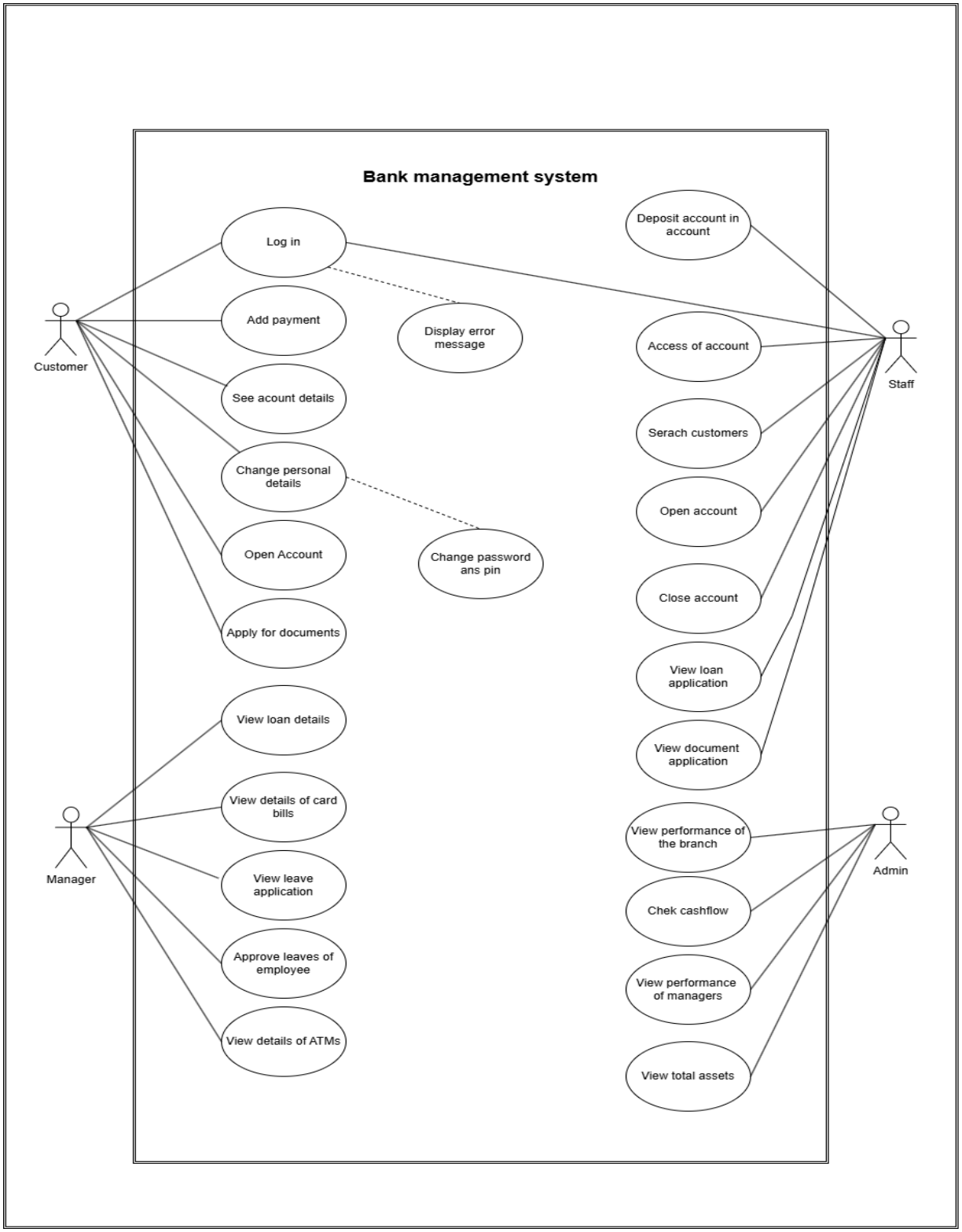
- The System should be available for the duration when the library operates and must be recovered within an hour or less if it fails. The system should respond to the requests within two seconds or less.

1.4.4 Maintainability:

- The software should be easily maintainable and adding new features and making changes to the software must be as simple as possible. In addition to this, the software must also be portable.

2 Design and Implementation Constraints

2.1 Use case diagram



+Figure 2.1-1 Use case diagram for Bank management system

2.2 Activity diagram and Swimlane diagram

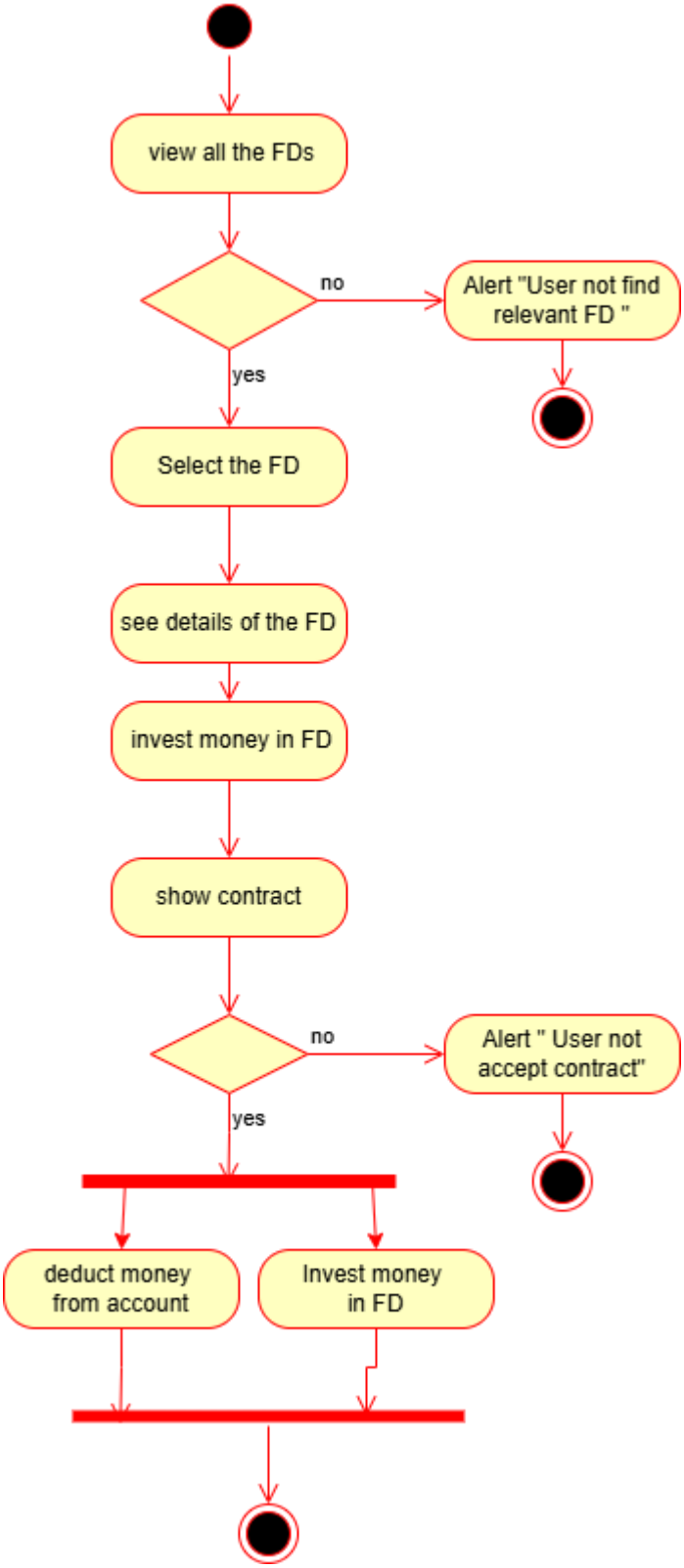


Figure 2.2-1 Activity diagram for Invest in FDs

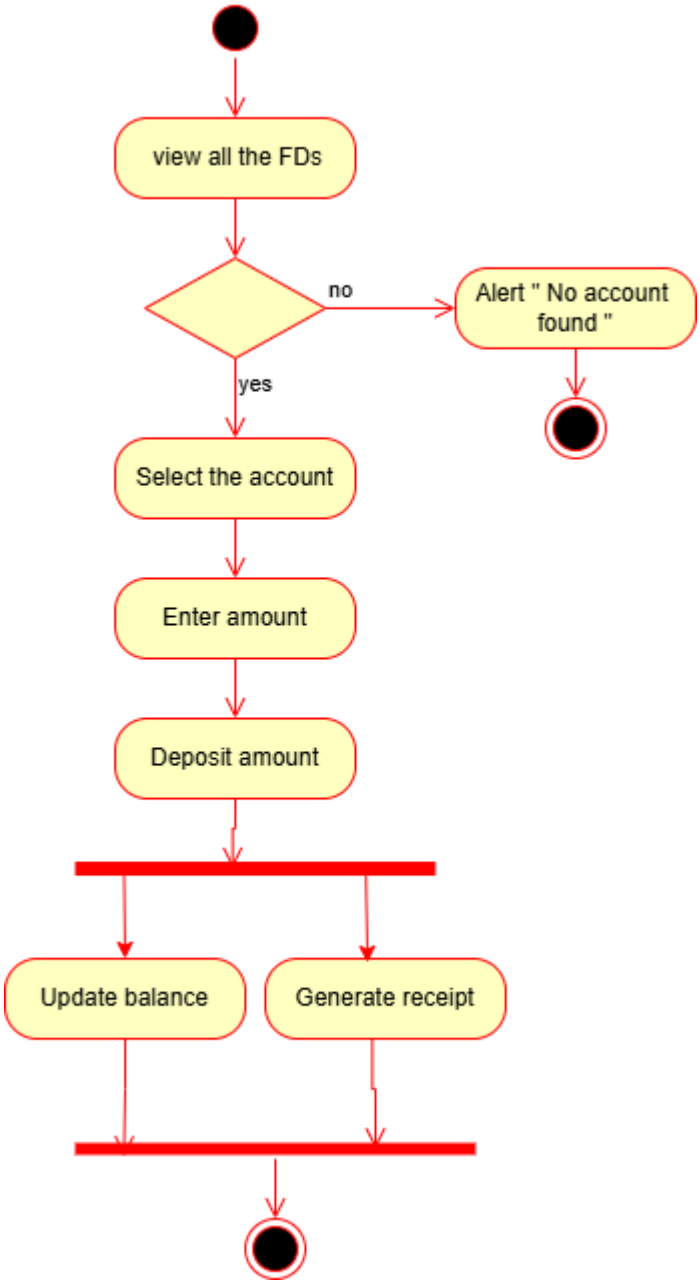


Figure 2.2-2 Activity diagram for deposit money

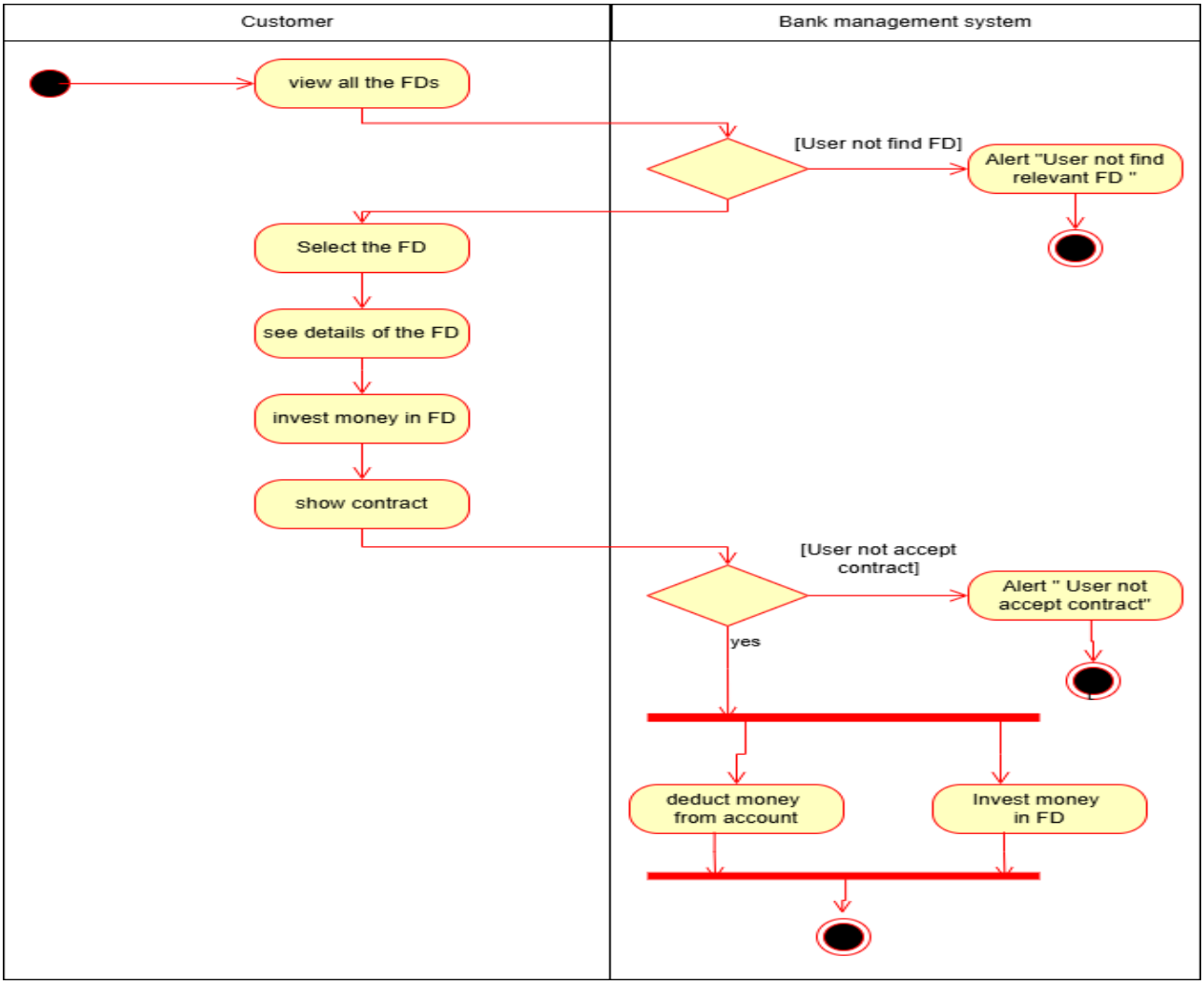


Figure 2.2-3 Swimlane diagram for Invest money in FDs

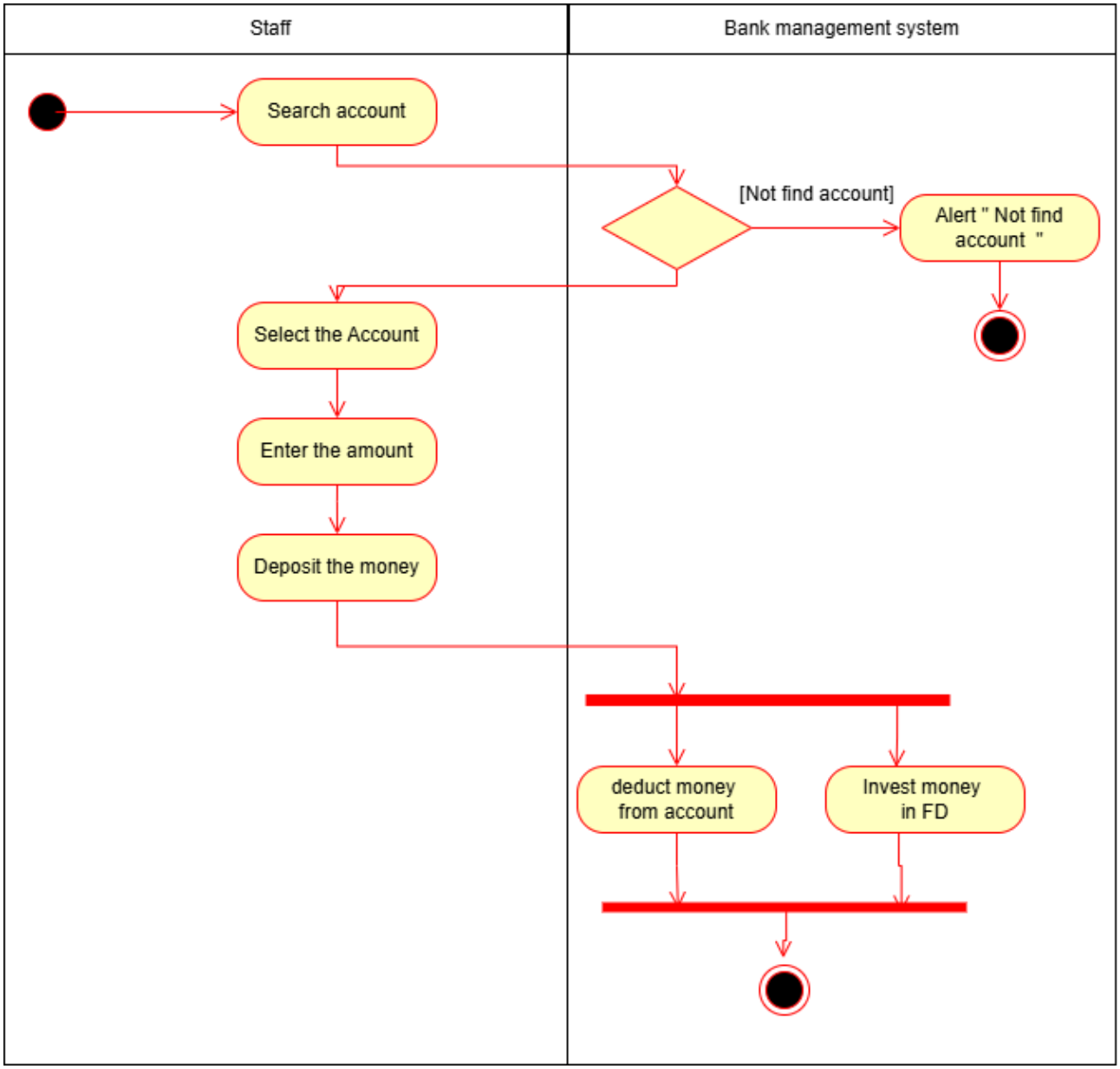


Figure 2.2-4 Swimlane diagram for deposit money

2.3 Sequence diagram

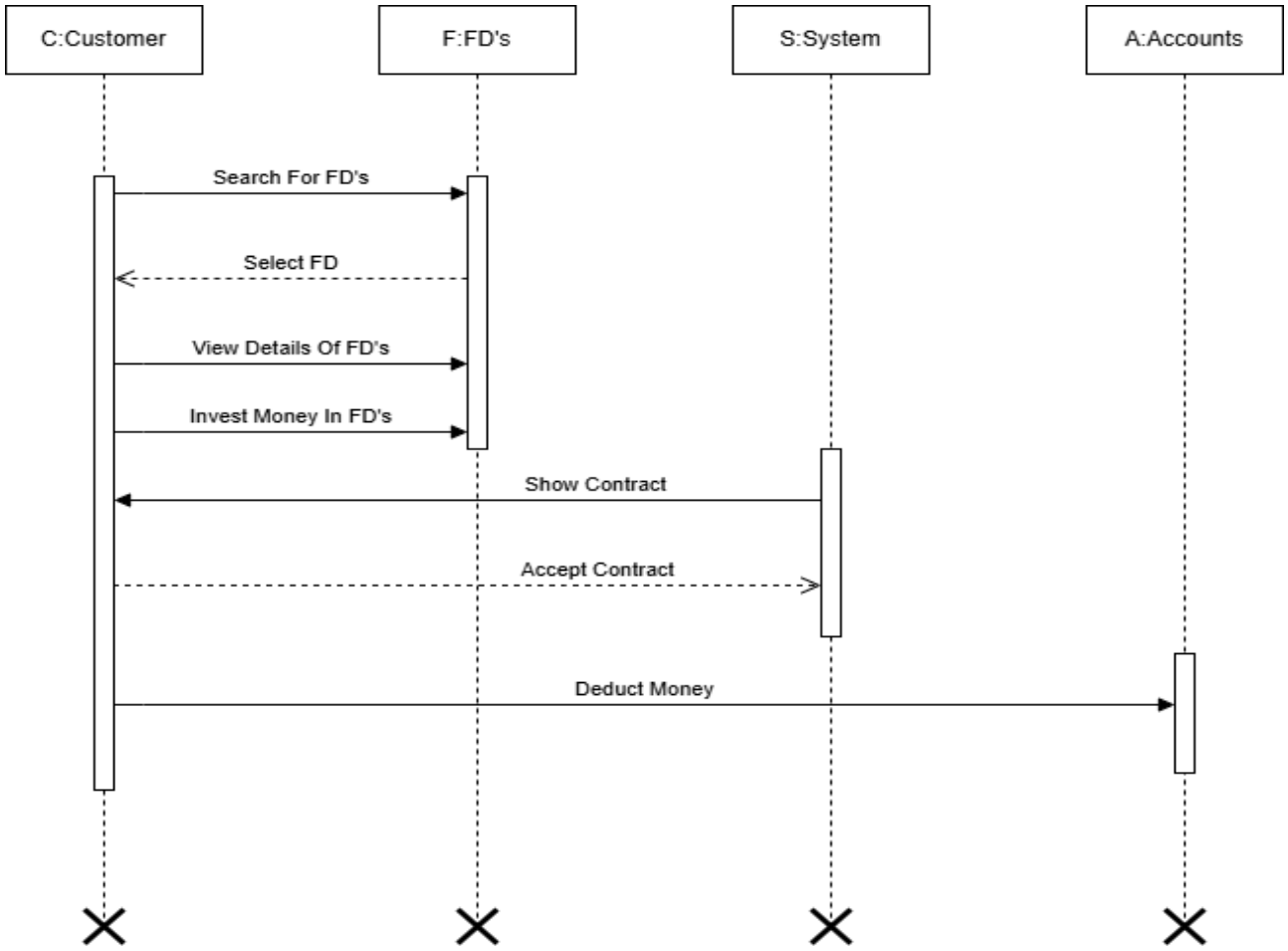


Figure 2.3-1 Sequence diagram Invest in FDs

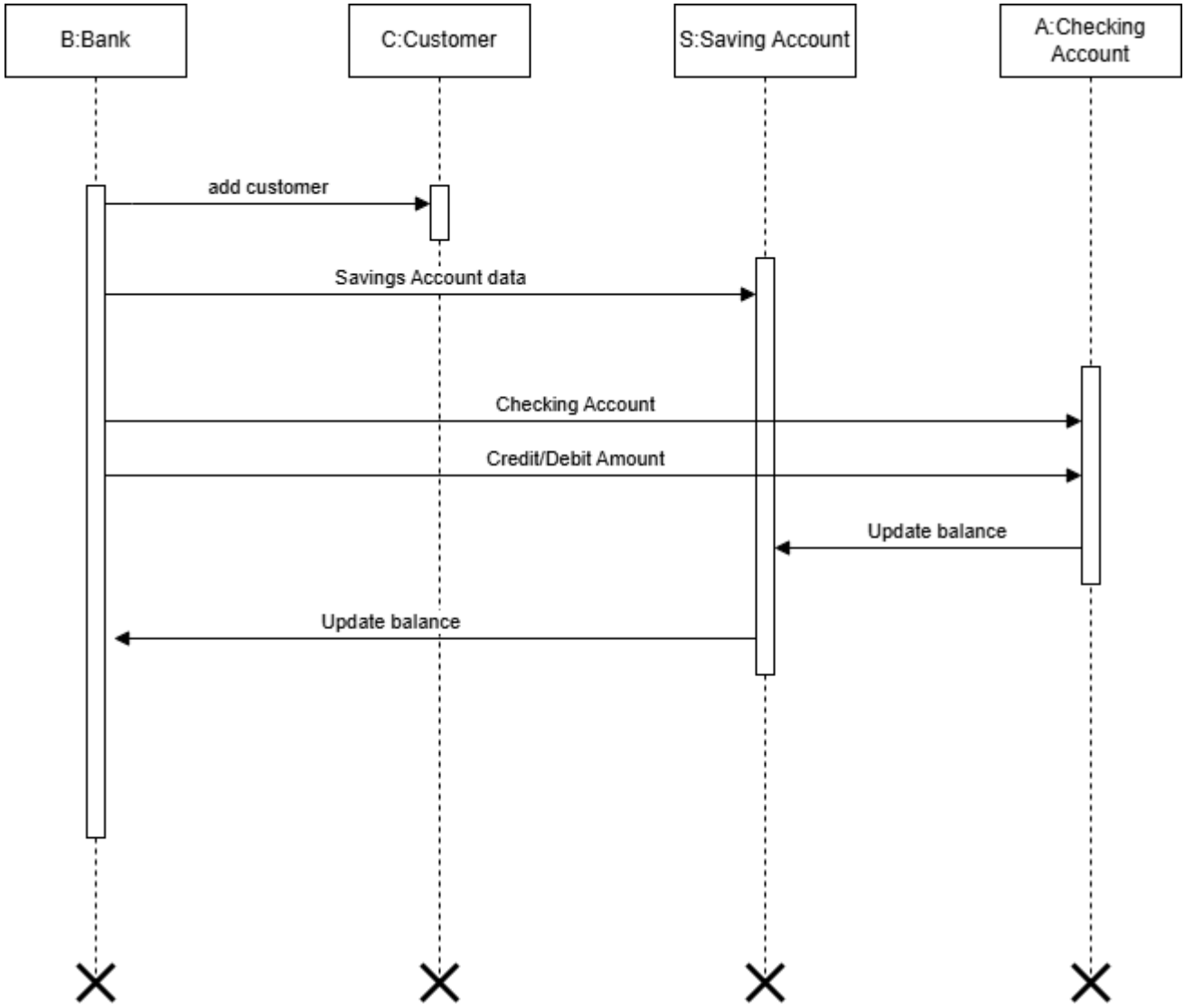


Figure 2.3-2 Sequence diagram Manage Account

2.4 State diagram

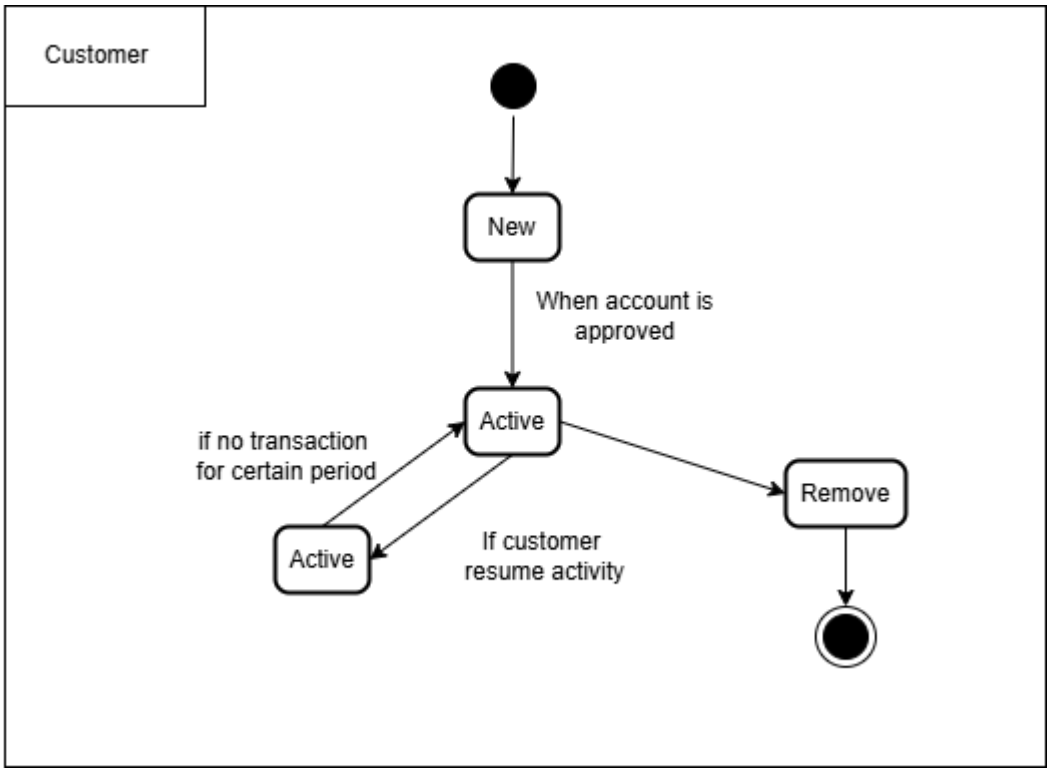


Figure 2.4-1 State diagram of Customer

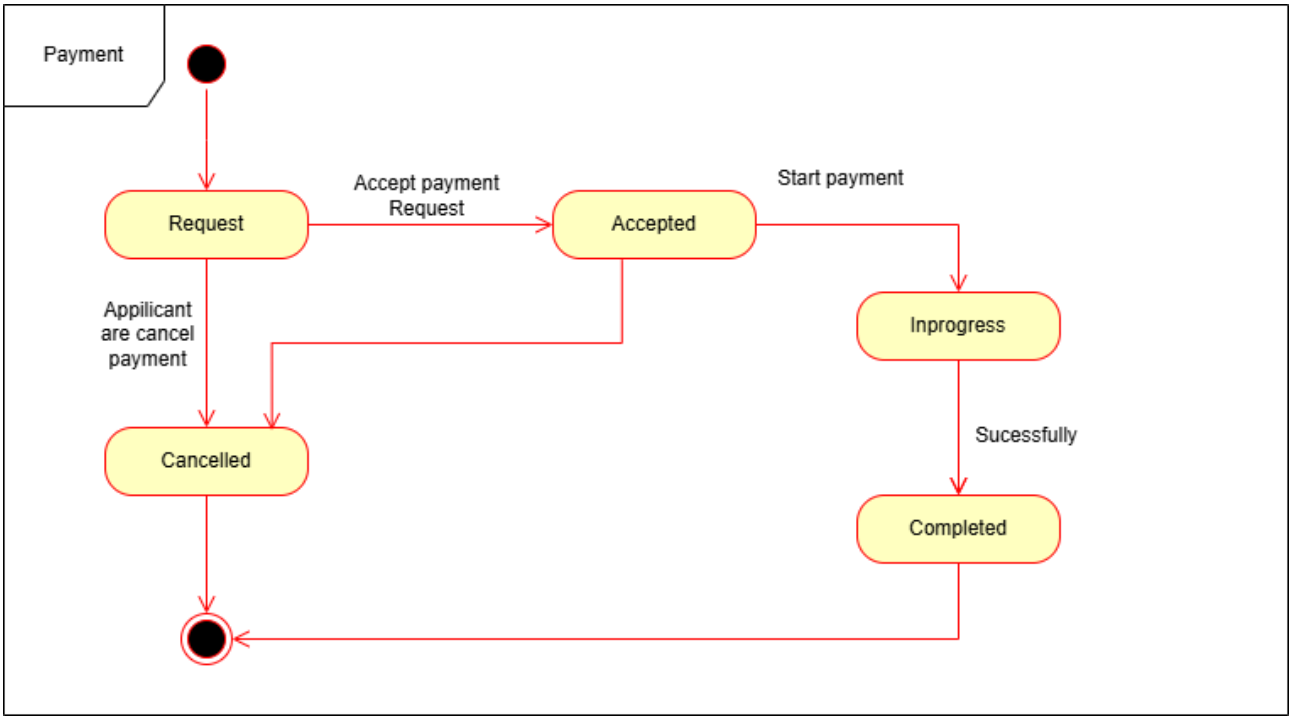


Figure 2.4-2 State diagram of Payment

2.5 Class diagram

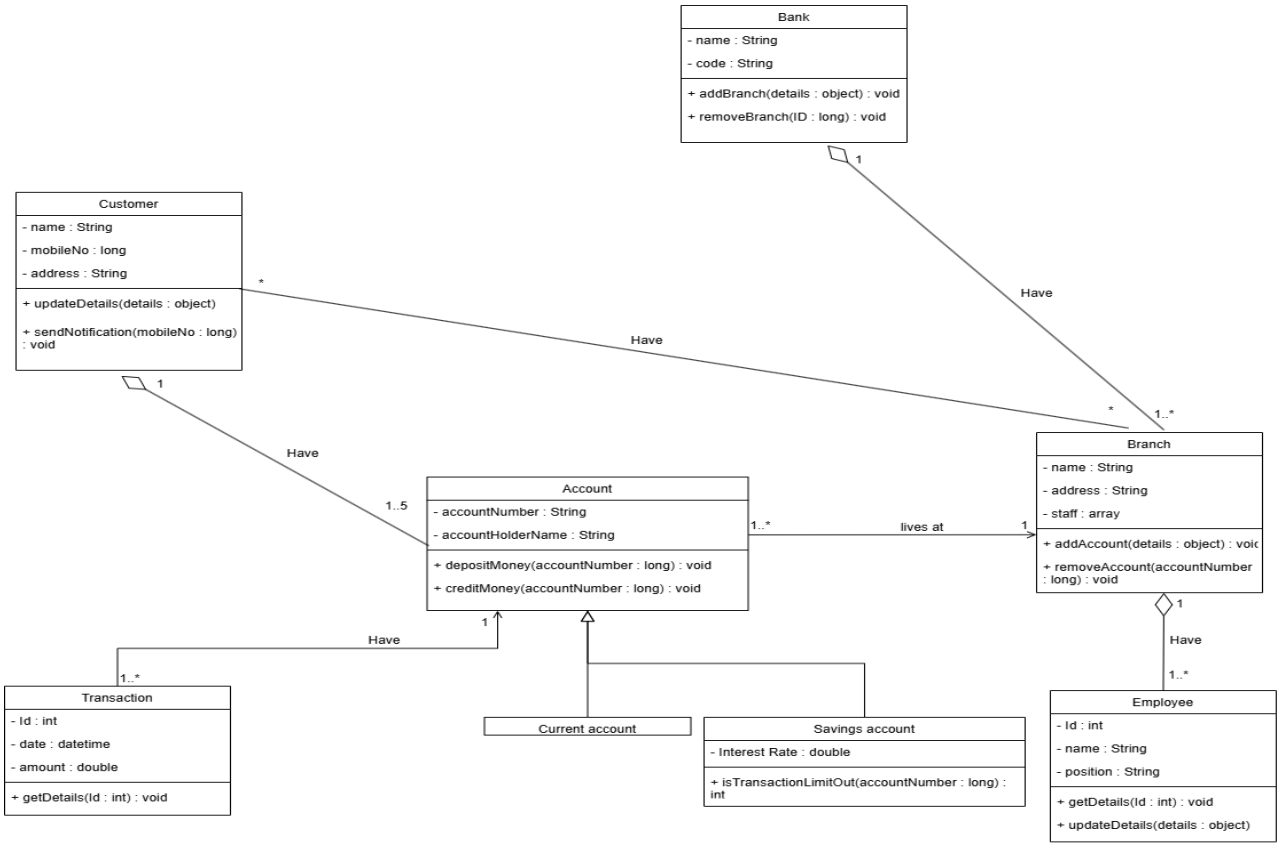


Figure 2.5-1 Class diagram for Bank management system

2.6 Data flow diagram

2.6.1 Context diagram (level-0)

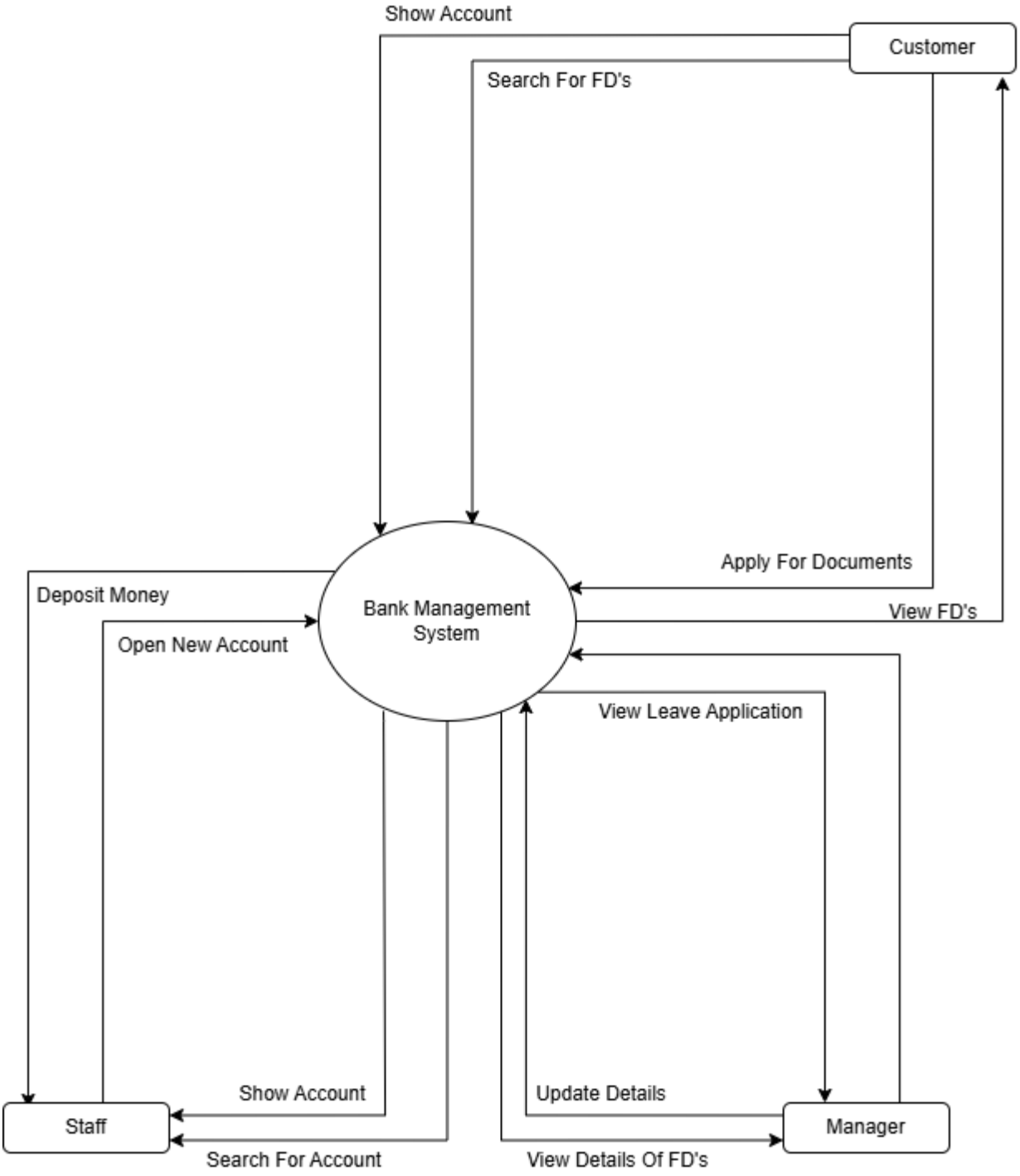


Figure 2.6-1 Context diagram for Bank management system

2.6.2 DFD Level-1

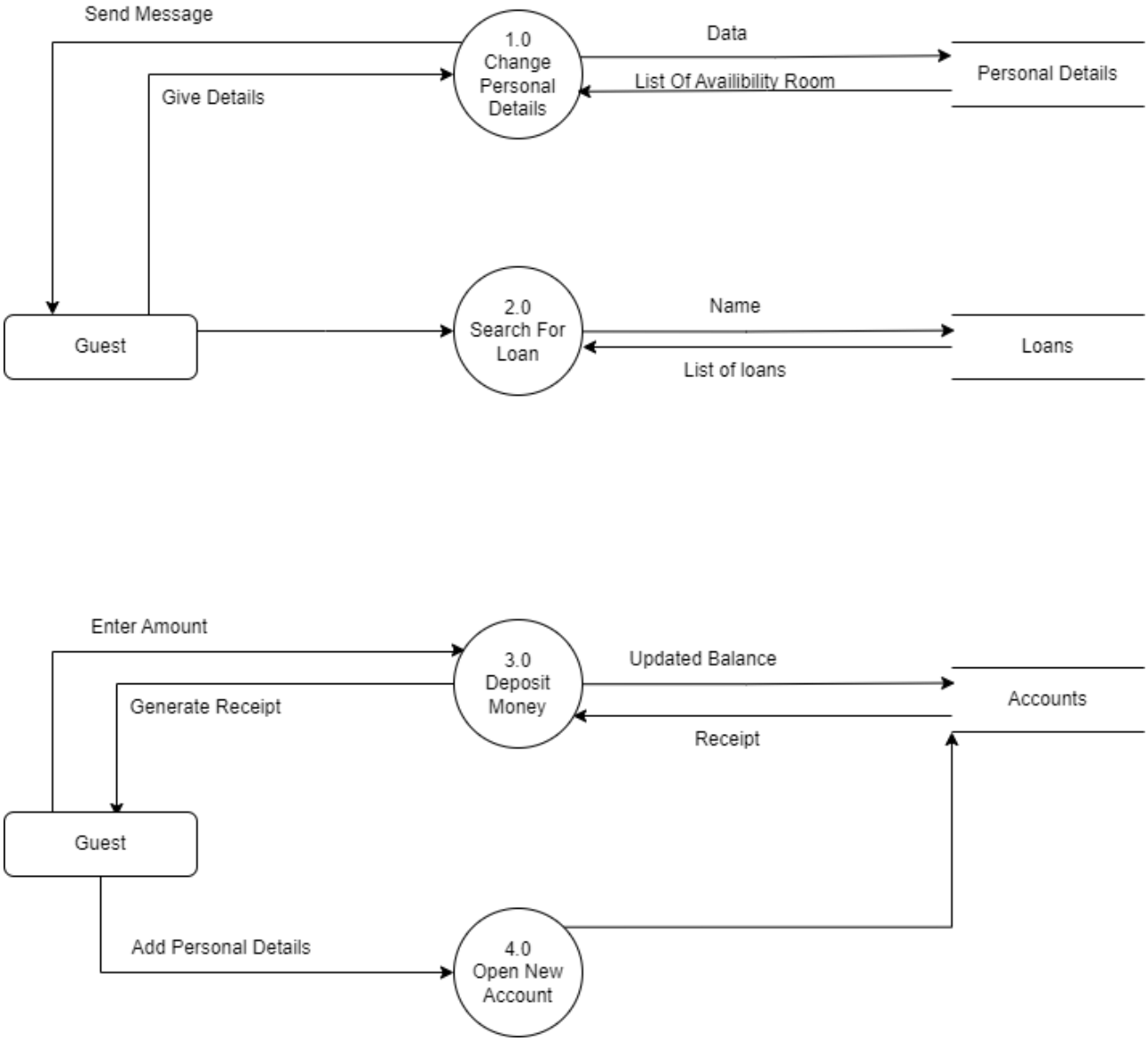


Figure 2.6-2 DFD level-1 for Bank management system

3 External interface requirement (Screens)

3.1 Screen-1: See Account Details

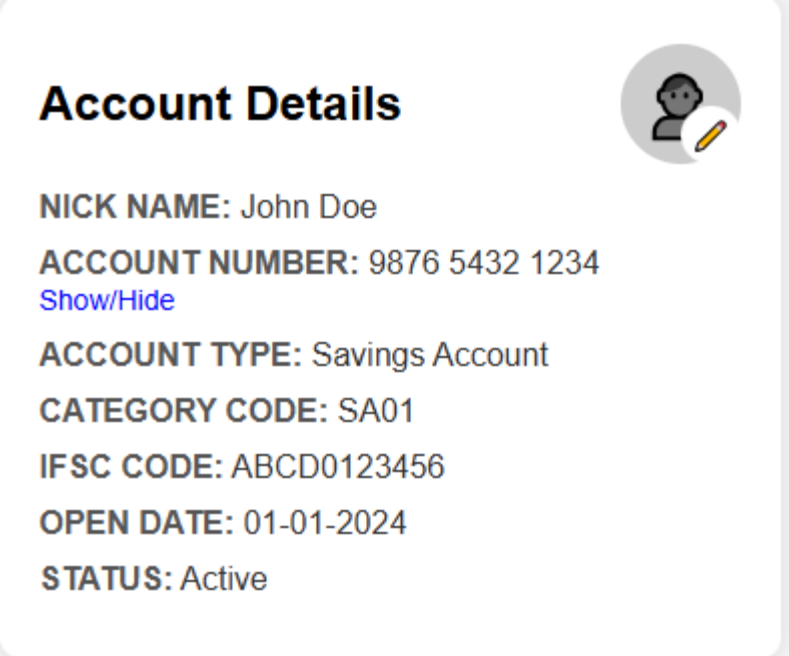


Figure 3.1-1 Screen-1: See Account Details

Purpose: The purpose of this screen is to display account details for a bank user. It includes essential information such as the account holder's nickname, account number (with a show/hide option for security), account type, category code, IFSC code, account opening date, and current status.

Table 3.1-1 See Account Details

Sr.	Screen Element	Input Type	O/M	1/N	Description
1	Nick Name	Textbox	M	1	Username field should be editable and accept the Username.
2	Account Number	Textbox	M	1	Account number field should be editable and masked for security.
3	Account Type	Textbox	M	1	Display account type (e.g., Savings, Checking).
4	Category Code	Textbox	M	1	Display category code assigned to the account.
5	IFSC Code	Textbox	M	1	Display IFSC code for transactions.
6	Open Date	Date Picker	M	1	Display the date when the account was opened.
7	Status	Textbox	M	1	Display account status (e.g., Active, Inactive).

3.2 Screen-2: Apply For Documents

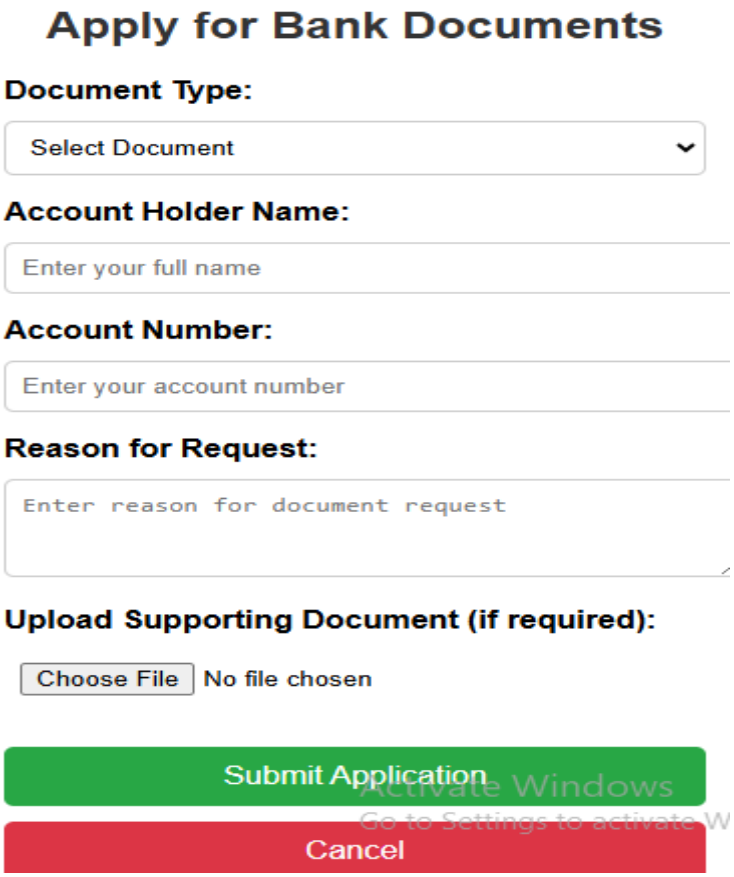


Figure 3.2-1 Screen-2: Apply For Documents

Purpose: The purpose of this screen is to allow users to apply for bank documents. It provides fields for entering necessary details such as document type, account holder name, account number, and the reason for the request.

Table 3.2-1 Apply For Documents

Sr.	Screen Element	Input Type	O/M	1/N	Description
1	Document Type	Dropdown	M	1	Select the type of document you want to request.
2	Account Holder Name	Textbox	M	1	Enter the full name of the account holder.
3	Account Number	Textbox	M	1	Enter the account number.
4	Reason for Request	Textbox	M	1	Provide a reason for requesting the document.
5	Upload Supporting Document	File Upload	O	N	Optional: Upload a document to support your request.
6	Submit Application	Button	M	1	Submit the completed application form.
7	Cancel	Button	O	N	Cancel the application process.

3.3 Screen-3:View Loan Application

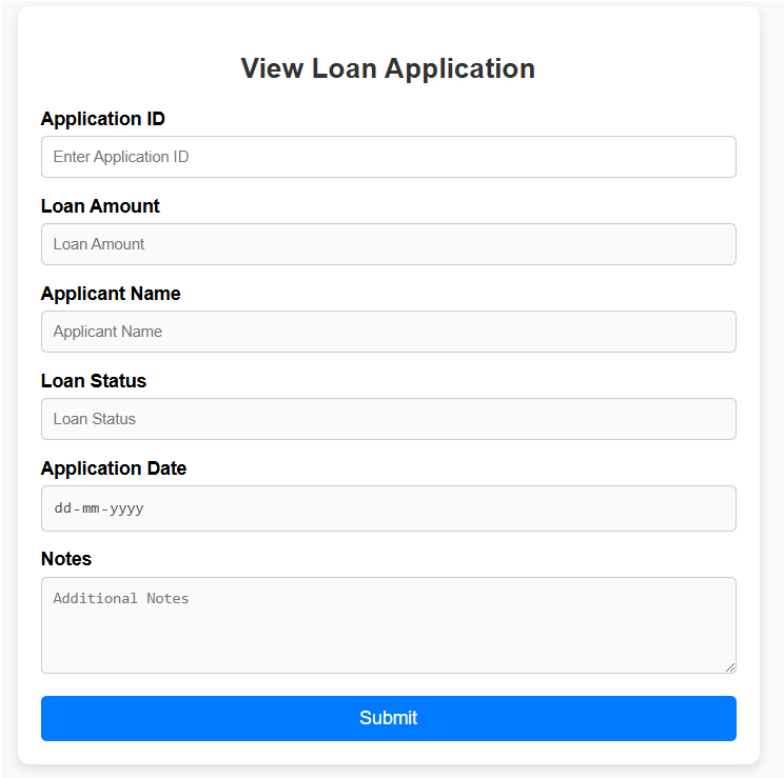


Figure 3.33-1 Screen-3: View Loan Application

Purpose: The purpose of this screen is to allow users to view and manage loan applications. It provides input fields for entering an application ID, loan amount, applicant name, loan status, application date, and additional notes

Table 3.3-1 View Loan Application

Sr.	Screen Element	Input Type	O/M	1/N	Description
1	Application ID	Textbox	M	1	Field to enter the application ID.
2	Loan Amount	Textbox	M	1	Field to display or edit the loan amount.
3	Applicant Name	Textbox	M	1	Field to display or edit the applicant's name.
4	Loan Status	Dropdown	M	1	Dropdown to select or display the loan status.
5	Application Date	Textbox	M	1	Field to display or edit the application date in the format `dd-mm-yyyy`.
6	Notes	Textbox	O	N	Field to provide additional notes related to the application.
7	Submit	Button	------	------	Button to submit the form data.

3.4 Screen-4: Leave Application Form

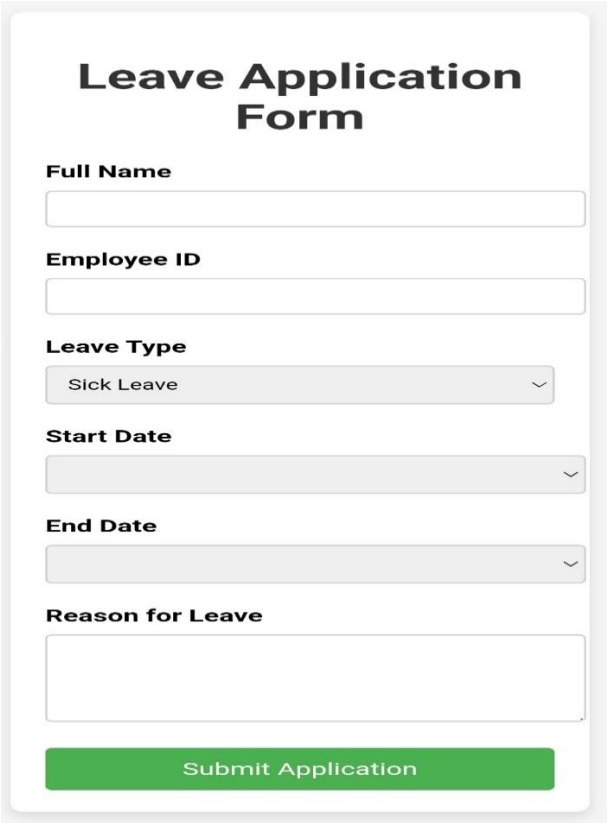


Figure 3.44-1 Screen-4: Leave Application Form

Purpose: The purpose of this screen is to allow employees to apply for leave. It includes fields for entering the employee's full name, employee ID, leave type (such as sick leave), start and end dates, and a reason for leave.

Table 3.4-1 Leave Application Form

Sr.	Screen Element	Input Type	O/M	1/N	Description
1	Full Name	Textbox	M	1	Field to input the full name of the applicant.
2	Employee ID	Textbox	M	1	Field to input the employee's ID.
3	Leave Type	Dropdown	M	1	Dropdown to select the type of leave (e.g., Sick Leave, Vacation, etc.).
4	Start Date	Date Picker	M	1	Field to select the start date of the leave.
5	End Date	Date Picker	M	1	Field to select the end date of the leave.
6	Reason for Leave	Textbox	O	1	Field to provide the reason for the leave application.
7	Submit Application	Button	------	------	Button to submit the leave application form.

3.5 Screen-5: View Branch Performance

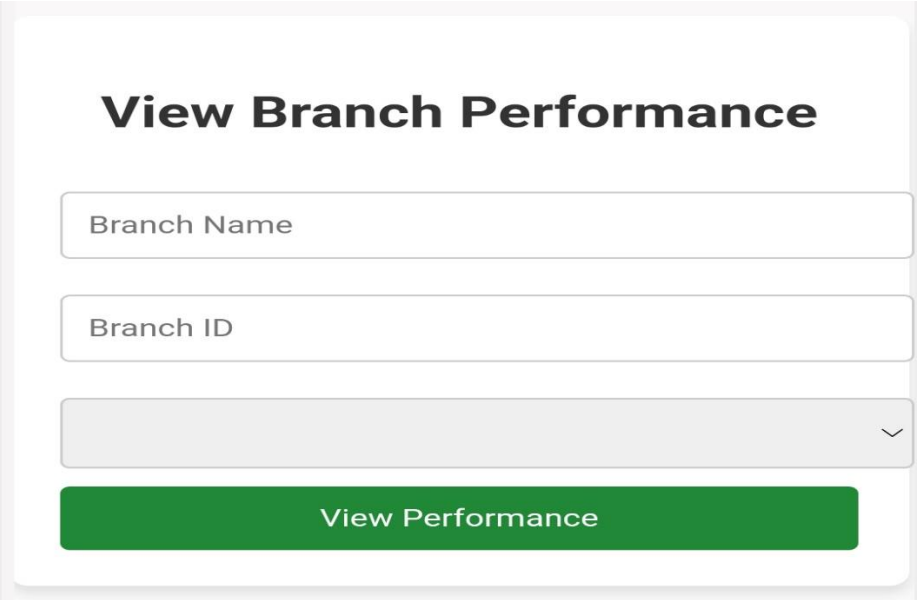


Figure 3.55-1 Screen-5: View Branch Performance

Purpose: The purpose of this screen is to allow users to view the performance of a specific branch. It includes fields to enter the branch name and branch ID, along with a dropdown (possibly for selecting performance criteria).

Table 3.5-1 View Branch Performance

Sr.	Screen Element	Input Type	O/M	1/N	Description
1	Branch Name	Textbox	O	1	Optional field to input the name of the branch.
2	Branch ID	Textbox	M	1	Mandatory field to input the unique ID of the branch.
3	Date	Dropdown	M	1	Field to select the date of the Performance.
4	View Performance	Button	------	------	Button to submit the details and view the performance metrics.

4 Database design

4.1 List of Tables

- Admin
- Staff
- Customer
- Manager
- Loan

Table 4.1-1 Table: Admin

Column Name	Data Type	Null	Constraints	Description
Admin_ID	INT	NOT NULL	PRIMARY KEY	Unique Identifier for each admin.
Name	VARCHAR(255)	NOT NULL		Name of the admin.
Email	VARCHAR(255)	NOT NULL	UNIQUE	Email address of the admin.
Password	VARCHAR(255)	NOT NULL		Password for the admin account.
Additional_Info	TEXT	NULL		Any additional information about the admin.
Contact_Number	VARCHAR(20)	NULL		Contact number of the admin.

Table 4.1-2 Table: Staff

Column Name	Data Type	Null	Constraints	Description
Staff_ID	INT	NOT NULL	PRIMARY KEY	Unique Identifier for each staff member.
Name	VARCHAR(255)	NOT NULL		Name of the staff member.
Email	VARCHAR(255)	NOT NULL	UNIQUE	Email address of the staff member.
Password	VARCHAR(255)	NOT NULL		Password for the staff account.
Position	VARCHAR(100)	NOT NULL		Job position of the staff member.
Contact_Number	VARCHAR(20)	NULL		Contact number of the staff member.

Table 4.1-3 Table: Customer

Column Name	Data Type	Null	Constraints	Description
Customer_ID	INT	NOT NULL	PRIMARY KEY	Unique Identifier for each customer.
Name	VARCHAR(255)	NOT NULL		Name of the customer.
Email	VARCHAR(255)	NOT NULL	UNIQUE	Email address of the customer.
Password	VARCHAR(255)	NOT NULL		Password for the customer account.
Address	TEXT	NULL		Address of the customer.
Contact_Number	VARCHAR(20)	NULL		Contact number of the customer.
Account_Number	VARCHAR(20)	NOT NULL	UNIQUE	Bank account number of the customer.
Balance	DECIMAL(10,2)	NOT NULL		Current balance of the customer's account.

Table 4.1-4 Table: Manager

Column Name	Data Type	Null	Constraints	Description
Manager_ID	INT	NOT NULL	PRIMARY KEY	Unique Identifier for each manager.
Name	VARCHAR(255)	NOT NULL		Name of the manager.
Email	VARCHAR(255)	NOT NULL	UNIQUE	Email address of the manager.
Password	VARCHAR(255)	NOT NULL		Password for the manager account.
Department	VARCHAR(100)	NOT NULL		Department the manager oversees.
Contact_Number	VARCHAR(20)	NULL		Contact number of the manager.

Table 4.1-5 Table: Loan

Column Name	Data Type	Null	Constraints	Description
Loan_ID	INT	NOT NULL	PRIMARY KEY	Unique Identifier for each Loan.
Loan_Type	VARCHAR(255)	NOT NULL		Type of the Loans.
Loan_Amount	DECIMAL(10,2)	NOT NULL		Amount of the Loan.
InterestRate	DECIMAL(10,2)	NOT NULL		InterestRate for the Loan.
Term	INT	NOT NULL		TermYears of the loan.

5 Stories and Scenario

5.1 Story-1: View my account details

*Story # **S1***	**:**	**As a** Customer, **I want to** view my account details **So that** I can track my balance and recent transactions easily..
Priority	**:**	High
Estimate	**:**	S
Reason	**:**	This is a core functionality that allows customers to track their account status, including their balance and transactions. It needs to be quick, accessible, and reliable, as customers frequently need to view their account details for various banking activities.

5.1.1 Scenario# S1.1

*Scenario# **S1.1***	**:**	Viewing Account Details with Valid Credentials
Prerequisite	**:**	The customer is registered and logged into the Bank Management System
Acceptance Criteria	**:**	**Given:** The customer is logged in to the system and navigates to the "Account Details" section. **When:** The customer selects the "View Account Details" option. **Then** The system should display the customer's account number, balance, recent transactions, and other relevant details

5.1.2 Scenario# S1.2

*Scenario# **S1.2***	**:**	Attempting to View Account Details with Invalid Credentials
Prerequisite	**:**	The customer is not logged into the system or has entered invalid login credentials.
Acceptance Criteria	**:**	**Given:** The customer tries to access their account details without logging in or by using invalid credentials. **When:** The customer enters incorrect login credentials or attempts to bypass login. **Then t**he system should display an error message such as "Invalid username or password" and prevent access to account details.

5.1.3 Scenario# S1.3

*Scenario# **S1.3***	**:**	Viewing Account Details During System Maintenance
Prerequisite	**:**	The Bank Management System is undergoing scheduled maintenance
Acceptance Criteria	**:**	**Given**: The customer tries to access their account details during a period when the system is unavailable due to maintenance. **When**: The customer logs in and attempts to view their account details. **Then**: The system should display a maintenance notification or a message like "Service is temporarily unavailable" and prevent access to account details. **Then**: Generate unique book id, barcode and spine label for various book of same title.

5.2 Story-2: Apply For Loan

*Story # **S2***	:	**As a** Customer, **I want to** apply for a loan online, **So that** I can submit my application without visiting the branch.
Priority	**:**	Medium
Estimate	**:**	M
Reason	**:**	Applying for loans online is an essential feature for customers, especially for convenience and accessibility. However, since this feature involves form filling and approval processes, it requires more development and testing efforts compared to simply viewing account details

5.2.1 Scenario# S2.1

*Scenario# **S2.1***	**:**	Checking Loan Eligibility Before Applying
Prerequisite	**:**	The customer is logged into the Bank Management System.
Acceptance Criteria	**:**	**Given**: The customer navigates to the "Loan Eligibility Check" section. **When**: The customer enters required details such as income, credit score, and existing loans. **Then**: The system should analyze the information and display whether the customer is eligible for the loan, along with the maximum loan amount they can apply for.

5.2.2 Scenario# S2.2

*Scenario# **S2.2***	:	Tracking Loan Application Status
Prerequisite	:	The customer has already applied for a loan.
Acceptance Criteria	:	**Given** The customer logs into the system and goes to the "Loan Application Status" section. **When**: The customer selects their submitted loan application. **Then**: The system should display the current status of the application (e.g., Pending, Approved, Rejected) along with any additional instructions if needed.

5.2.3 Scenario# S2.3

*Scenario# **S2.3***	:	Canceling a Loan Application Before Approval
Prerequisite	:	The customer has submitted a loan application but it has not yet been approved.
Acceptance Criteria	:	**Given**: The customer logs in and navigates to their loan application details. **When**: The customer selects the "Cancel Application" option. **Then**: The system should allow the cancellation, display a confirmation message, and update the loan status as "Canceled."

5.3 Story-3: Leave Application

*Story # **S3***	:	**As** Customer, **I want to** apply for leave online **So that** I can submit my leave request without visiting HR.
Priority	:	High
Estimate	:	L
Reason	:	Applying for leave online is a crucial feature for employees, enabling a seamless and efficient process without paperwork. It enhances accessibility and transparency by allowing employees to track leave balances and approval statuses. Since this involves validation of leave balance, approval workflows, and notifications, it requires more development and testing efforts.

5.3.1 Scenario# S3.1

*Scenario# **S3.1***	:	Applying for Leave with Sufficient Leave Balance
Prerequisite	:	The employee is logged into the system and has sufficient leave balance.
Acceptance Criteria	:	**Given**: The employee navigates to the "Leave Application" section. **When**: The employee selects the leave type, duration, and reason, then submits the application. **Then**: The system should validate the details, deduct the leave balance, and send the request to the manager for approval. A confirmation message should be displayed.

5.3.2 Scenario# S3.2

*Scenario# **S3.2***	:	Canceling a Leave Application Before Approval
Prerequisite	:	The employee has already submitted a leave request, and it is pending approval.
Acceptance Criteria	:	**Given**: The employee navigates to the "Leave Requests" section. **When**: The employee selects an unapproved leave request and chooses the "Cancel Request" option. **Then**: The system should cancel the request, update the leave balance, and notify the manager.

5.3.3 Scenario# S3.3

*Scenario# **S3.3***	:	Manager Approving or Rejecting a Leave Application
Prerequisite	:	The employee has submitted a leave request, and the manager has access to pending approvals.
Acceptance Criteria	:	**Given**: The manager logs into the system and navigates to the "Pending Leave Approvals" section. **When**: The manager reviews the leave request and selects either "Approve" or "Reject". **Then**: The system should update the leave status accordingly and notify the employee of the decision via email or in-app notification.

6 Test cases

Project Name:	**Bank Management system**	**Test Designed by:**	**P. U. Jadeja**
Module Name:	**Apply for loan**	**Test Designed date:**	01-10-2023
Release Version:	**1.0**	**Test Executed by:**	**R. B. Gondaliya**
		Test Execution date:	15-01-2023

Pre-condition: **Web application should be accessible**				
Test Case ID	**Test Title**	**Test Type**	**Description**	**Test Case ID**
TC_001	Apply for a loan with valid data	Functional	Apply for loan in bank management system with valid credential	TC_001
TC_002	Apply for a loan with invalid data	Functional	Apply for loan in bank management system with invalid credential	TC_002

Test Case Title	Apply for a loan with valid credential
Test Type	Functional
Test Priority	High
Pre-condition	The user must be logged into the Bank Management System

Test Step	Test Case Description	Expected Result	Actual Result	Status	Comment	Data	Bug ID
1	Navigate to the "Apply for Loan" section	The loan application form should be displayed	Form displayed	Pass			
2	Enter valid loan amount, duration, and purpose	Fields should accept valid inputs and display them correctly	Inputs accepted	Pass		Loan amount: 100000 Duration: 12 months Purpose: Home renovation	
3	Upload valid required documents (e.g., proof of income)	The system should accept the uploaded files	Documents uploaded	Pass		File: income_proof.pdf	
4	Click on the "Submit" button properly or not	The system should process the application and display a confirmation	Application submitted	Pass	A confirmation message should appear: "Your loan application has been submitted successfully."		

Test Case Title	Apply for a loan with invalid credential
Test Type	Functional
Test Priority	High
Pre-condition	The user must be logged into the Bank Management System

Test Step	Test Case Description	Expected Result	Actual Result	Status	Comment	Data	Bug ID
1	Navigate to the "Apply for Loan" section	The loan application form should be displayed	Form displayed	Pass			
2	Enter valid loan amount, duration, and purpose	The system should reject invalid inputs and display an error message	Error message displayed	Pass		Loan amount: -5000 Duration: 0 months Purpose: N/A	
3	Upload valid required documents (e.g., proof of income)	The system should reject invalid or missing files and display an error	Error message displayed	Pass		File: empty	
4	Click on the "Submit" button with invalid data	The system should not process the application and should display an error message	Application not submitted	Pass	A validation error message should appear: "Invalid loan details or missing documents."		

Project Name:	**Bank Management system**	**Test Designed by:**	**P. U. Jadeja**
Module Name:	**Apply for leave**	**Test Designed date:**	01-10-2023
Release Version:	**1.0**	**Test Executed by:**	**R. B. Gondaliya**
		Test Execution date:	15-01-2023

Pre-condition: **Web application should be accessible**				
Test Case ID	**Test Title**	**Test Type**	**Description**	**Test Case ID**
TC_001	Apply for a leave with valid data	Functional	Apply for leave in bank management system with valid credential	TC_001
TC_002	Apply for a leave with invalid data	Functional	Apply for leave in bank management system with invalid credential	TC_002

Test Case Title	Apply for a leave with valid credential
Test Type	Functional
Test Priority	High
Pre-condition	The user must be logged into the Bank Management System

Test Step	Test Case Description	Expected Result	Actual Result	Status	Comment	Data	Bug ID
1	Navigate to the "Apply for Leave" section	The leave application form should be displayed	Form displayed	Pass			
2	Enter valid leave type, duration, and reason	Fields should accept valid inputs and display them correctly	Inputs accepted	Pass		Leave type: Sick Leave Duration: 3 days Reason: Medical Rest	
3	Upload required documents (if applicable, e.g., medical certificate)	The system should accept the uploaded files	Documents uploaded	Pass		File: medical_certificate.pdf	
4	Click on the "Submit" button	The system should process the leave request and display a confirmation	Leave request submitted	Pass	A confirmation message should appear: "Your loan application has been submitted successfully."		

Test Case Title	Apply for a leave with invalid credential
Test Type	Functional
Test Priority	High
Pre-condition	The user must be logged into the Bank Management System

Test Step	Test Case Description	Expected Result	Actual Result	Status	Comment	Data	Bug ID
1	Navigate to the "Apply for Leave" section	The leave application form should be displayed	Form displayed	Pass			
2	Enter invalid leave details (e.g., invalid dates or exceeding allowed leave balance)	The system should reject invalid inputs and display an error message	Error message displayed	Pass		Leave dates: Start - 01/01/2025, End - 01/01/2024 (invalid date range)	
3	Upload invalid or missing required documents (if applicable)	The system should reject invalid or missing files and display an error	Error message displayed	Pass		File: empty	
4	Click on the "Submit" button with invalid data	The system should not process the application and should display an error message	Application Not Submitted	Pass	A validation error message should appear: "Invalid leave request details or missing documents."		

7 References

- http://www.w3schools.com/html/html_intro.asp
- https://www.w3schools.com/php/default.asp
- https://www.javatpoint.com/uml

BankSystem

GET New Request

https://api.openweathermap.org/data/3.0/onecall?lat={33.44}&lon={-94.04}&exclude={part}&appid={1e6c6bae89347497632a818f549a79c9}

StartFragment

Product concept

Get essential weather data, short-term and long-term forecasts and aggregated weather data is easy with our OpenWeather **One Call API 3.0**. This product designed to ensure easy migration from the Dark Sky API.

One Call API 3.0 contains 4 endpoints and provides access to various data:

- and government weather alerts
    - minute forecast for 1 hour
    - hourly forecast for 48 hours
    - daily forecast for 8 days
- **Weather data for any timestamp** for 46+ years historical archive and 4 days ahead forecast
- **Daily aggregation** of weather data for 46+ years archive and 1.5 years ahead forecast
- **Weather overview** with a human-readable weather summary for today and tomorrow's forecast, utilizing OpenWeather AI technologies

EndFragment

PARAMS

lat	{33.44} Latitude, decimal (-90; 90). If you need the geocoder to automatic convert city names and zip-codes to geo coordinates and the other way around, please use our Geocoding API
lon	{-94.04} Longitude, decimal (-180; 180). If you need the geocoder to automatic convert city names and zip-codes to geo coordinates and the other way around, please use our Geocoding API

exclude	{part} By using this parameter you can exclude some parts of the weather data from the API response. It should be a comma-delimited list (without spaces). Available values: current minutely hourly daily alerts
appid	{1e6c6bae89347497632a818f549a79c9} Your unique API key (you can always find it on your account page under the "API key" tab)

POST New Request

https://api.openweathermap.org/data/3.0/onecall?lat={33.44}&lon={-94.04}&exclude={part}&appid={1e6c6bae89347497632a818f549a79c9}

StartFragment

Professional collections

For professionals and specialists with middle sized project, we recommend our Professional collections, which included Current & Forecasts collection, Historical weather data collection, Weather Maps collection and other APIs.

For Enterprise level projects we provide Enterprise license, which is included all forecast products and current state, along with alerts, maps, and other products. Learn more

You can read the How to Start guide and enjoy using our powerful weather APIs right now.

EndFragment

PARAMS

lat	{33.44} Latitude, decimal (-90; 90). If you need the geocoder to automatic convert city names and zip-codes to geo coordinates and the other way around, please use our Geocoding API
lon	{-94.04} Longitude, decimal (-180; 180). If you need the geocoder to automatic convert city names and zip-codes to geo coordinates and the other way around, please use our Geocoding API
exclude	{part} By using this parameter you can exclude some parts of the weather data from the API response. It should be a comma-delimited list (without spaces). Available values:

current minutely hourly daily alerts

appid {1e6c6bae89347497632a818f549a79c9}

Your unique API key (you can always find it on your account page under the "API key" tab)

PUT New Request

https://api.openweathermap.org/data/3.0/onecall?lat={33.44}&lon={-94.04}&exclude={part}&appid={1e6c6bae89347497632a818f549a79c9}

StartFragment

Product concept

Get essential weather data, short-term and long-term forecasts and aggregated weather data is easy with our OpenWeather **One Call API 3.0**. This product designed to ensure easy migration from the Dark Sky API.

One Call API 3.0 contains 4 endpoints and provides access to various data:

- and government weather alerts
  - minute forecast for 1 hour
  - hourly forecast for 48 hours
  - daily forecast for 8 days
- **Weather data for any timestamp** for 46+ years historical archive and 4 days ahead forecast
- **Daily aggregation** of weather data for 46+ years archive and 1.5 years ahead forecast
- **Weather overview** with a human-readable weather summary for today and tomorrow's forecast, utilizing OpenWeather AI technologies

EndFragment

PARAMS

lat {33.44}

Latitude, decimal (-90; 90). If you need the geocoder to automatic convert city names and zip-codes to geo coordinates and the other way around, please use our Geocoding API

lon {-94.04}

Longitude, decimal (-180; 180). If you need the geocoder to automatic convert city names and zip-codes to geo coordinates and the other way around, please use our Geocoding API

exclude {part}

By using this parameter you can exclude some parts of the weather data from the API response. It should be a comma-delimited list (without spaces).

Available values:

current minutely hourly daily alerts

appid {1e6c6bae89347497632a818f549a79c9}

Your unique API key (you can always find it on your account page under the "API key" tab)

PATCH New Request

https://api.openweathermap.org/data/3.0/onecall?lat={33.44}&lon={-94.04}&exclude={part}&appid={1e6c6bae89347497632a818f549a79c9}

StartFragment

Daily Aggregation

To learn about how get access to aggregated weather data for a particular date from 2nd January 1979 till long-term forecast for 1,5 years ahead, please use this section of the documentation.

If you are interested in other functionality on One Call API 3.0, please check Product concept to follow the right section.

EndFragment

PARAMS

lat {33.44}

Latitude, decimal (-90; 90). If you need the geocoder to automatic convert city names and zip-codes to geo coordinates and the other way around, please use our Geocoding API

lon {-94.04}

Longitude, decimal (-180; 180). If you need the geocoder to automatic convert city names and zip-codes to geo coordinates and the other way around, please use our Geocoding API

exclude {part}

By using this parameter you can exclude some parts of the weather data from the API response. It should be a comma-delimited list (without spaces). Available values:

current minutely hourly daily alerts

appid {1e6c6bae89347497632a818f549a79c9}

Your unique API key (you can always find it on your account page under the "API key" tab)

DELETE **New Request**

https://api.openweathermap.org/data/3.0/onecall?lat={33.44}&lon={-94.04}&exclude={part}&appid={1e6c6bae89347497632a818f549a79c9}

StartFragment

Weather overview

This section describes how to get weather overview with a human-readable weather summary for today and tomorrow's forecast, utilizing OpenWeather AI technologies.

If you are interested in other functionality on One Call API 3.0, please check Product concept to follow the right section.

EndFragment

PARAMS

lat	{33.44} Latitude, decimal (-90; 90). If you need the geocoder to automatic convert city names and zip-codes to geo coordinates and the other way around, please use our Geocoding API
lon	{-94.04} Longitude, decimal (-180; 180). If you need the geocoder to automatic convert city names and zip-codes to geo coordinates and the other way around, please use our Geocoding API
exclude	{part} By using this parameter you can exclude some parts of the weather data from the API response. It should be a comma-delimited list (without spaces). Available values: current minutely hourly daily alerts
appid	{1e6c6bae89347497632a818f549a79c9} Your unique API key (you can always find it on your account page under the "API key" tab)