Bank E-Statement Application.

Documentation Index:

- **Project Completion Status**

Design	Yes
Use case diagram	Yes
Class Diagram	Yes
Design Pattern	2 design patterns
Sequence Diagram	Yes
State Chart Diagram	Yes

Accept CSV Data	Yes
Store CSV data in Database	Yes
Calculate running balance	Yes
Downloadable CSV data	Yes
Admin Login Functionality	Yes

1. Project Overview:

Admin side functionalities:

A dashboard which has functionalities for the admin to login and upload the data as **.Csv** files .Using this dashboard ,the admin can upload csv files which will be stored in **Mysql5 database.** All the stored data should be processed and the running balance must be calculated. The admin should be able to view all the records.

Customer/End-User Functionalities:

The customer or the end-user will be able to view their transaction details and will be able to download them as .csv file.

2.Requirement Analysis:

- **User Requirements Analysis:**
  - The proposed system should have admin login functionality.
  - The admin, after logging should be able to upload the **.Csv files** which should be stored in the database.
  - Running balance should be calculated against each record upon uploading.
  - The proposed system should be able to handle data with **multiple account numbers.**
  - The end-user or bank customers should be able to view their data based on **account number.**

- The end-users/customers must be able to download the transaction details as **.Csv file.**

- **Performance requirements:**
  - Peak load performance for admin functionality must be less than 2 seconds
  - Peak load performance for e-commerce must be less than 3 seconds during festival days.
  - Peak load performance of e-commerce must be less than 2 seconds for normal days.

- **Libraries and dependencies used:**

1.spring-security: spring security secures the web application by providing the basic authentication requirement for all **HTTP endpoints.** All the pages in the web applications will require a basic authentication other than those for which the **urls** are whitelisted. The security can be scaled up to higher levels if required.

2.Spring-data-jpa : Spring Data JPA handles most of the complexity of JDBC-based database access and ORM (Object Relational Mapping). It reduces the boilerplate code required by JPA. It makes the implementation of your persistence layer easier and faster.

3.spring-boot-thymleaf : Thymeleaf is a Java-based library used to create a web application. It provides a good support for serving a XHTML/HTML5 in web applications.

4. Mysql5 : MySQL is the most popular Open Source Relational SQL Database Management System. MySQL is one of the best RDBMS being used for developing various web-based software applications.

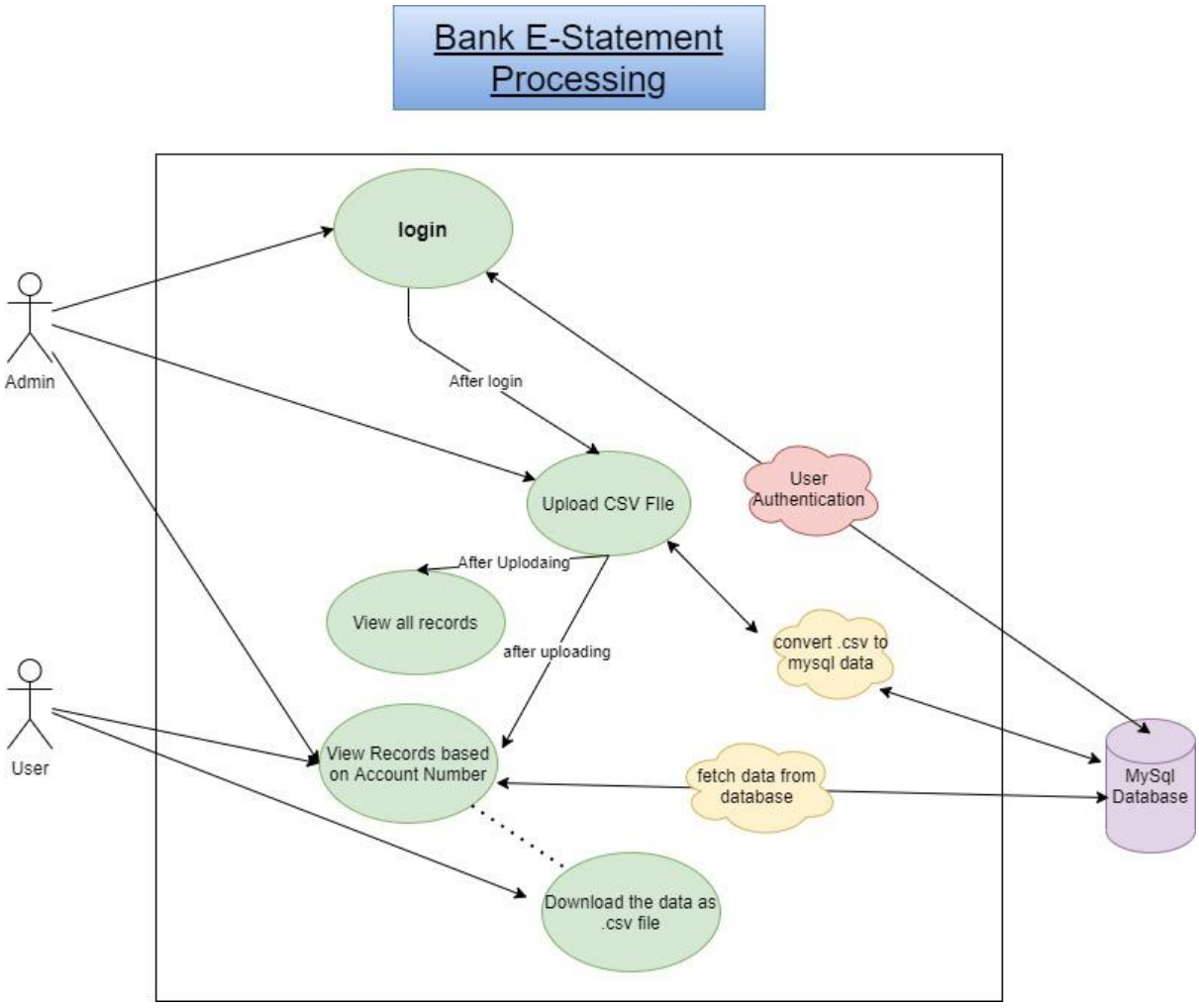
5.Bootstrap: Bootstrap is the most popular front end framework in the recent time. It is sleek, intuitive, and powerful mobile first front-end framework for faster and easier web development. It uses HTML, CSS and Javascript.

3.Features List :

1. Admin login functionality
2. uploading data in .Csv format.
3. Display all the data.
4. Calculate the running balance
5. Filter datasets based on Account Number
6. Downloading the data as .Csv files.

4.Use Case:

4.1. Usecase Diagram:



4.2Fully Dressed Use Case:

- **Uploading data as .Csv File:**

Use Case Headings	Explanation
Name	Loading of Csv files
Scope	Storing data in database
Level	Admin Goal
Primary Actor	Admin
Pre-Conditions	The dataset should be .CSV format
Success Guarantee	The data is available in the database
Frequency of occurrence	Quite often occurring use case(High-priority)

- **Calculating the running balance**

Use Case Headings	Explanation
Name	Calculating the running balance
Scope	Calculating and storing running balance
Level	Admin Goal
Primary Actor	Admin
Pre-Conditions	The dataset should be loaded
Success Guarantee	The running balance should be calculated correctly for each account
Frequency of occurrence	Quite often occurring use case(High-priority)

- **Displaying/Loading All data**

Use Case Headings	Explanation
Name	Displaying the data
Scope	Data should be displayed in table/grid format
Level	Admin Goal
Primary Actor	Admin
Pre-Conditions	The dataset should be loaded
Success Guarantee	The data is displayed in grid format
Frequency of occurrence	Quite often occurring use case(High-priority)

- **Filtering Data by Account Number:**

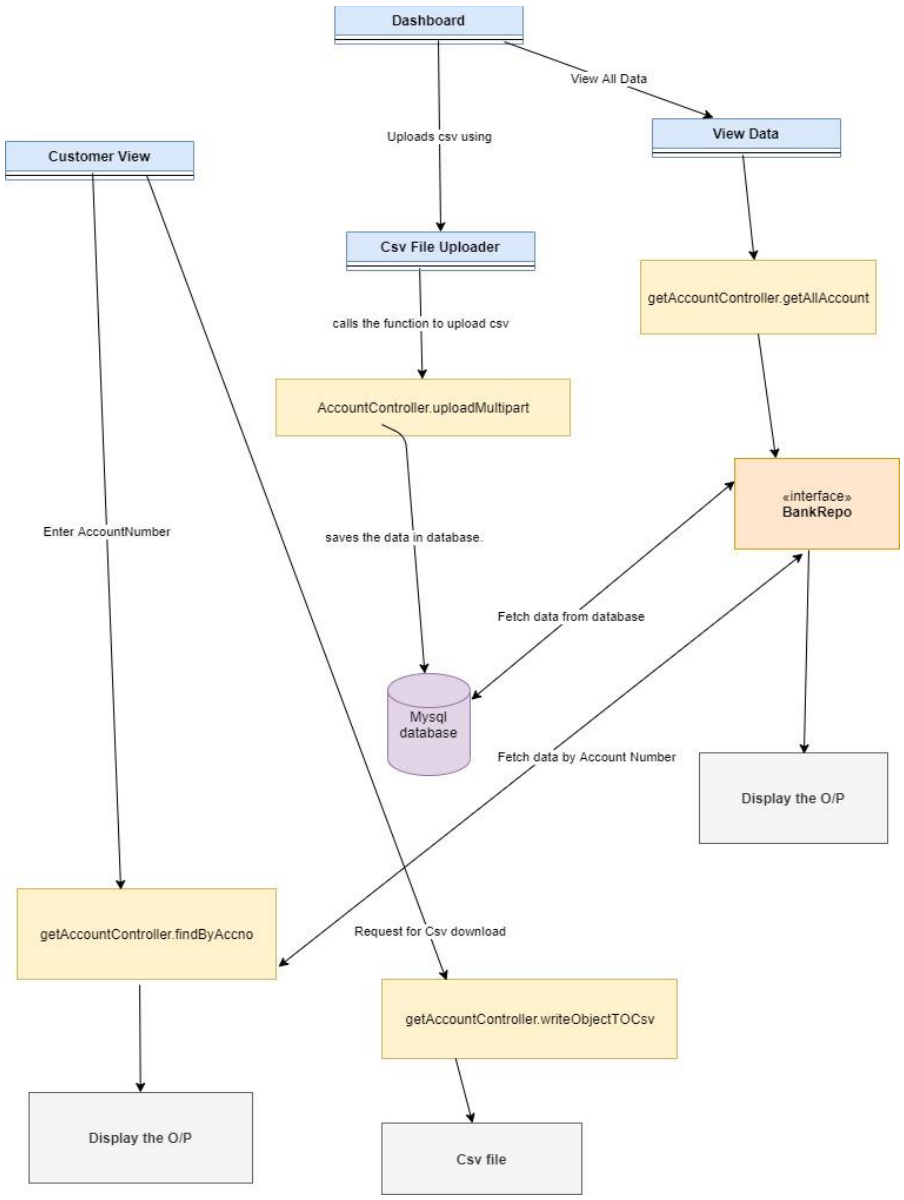
Use Case Headings	Explanation
Name	Filtering data by Account Number
Scope	Displaying data
Level	Admin Goal, User Goal
Primary Actor	Admin,Customer
Pre-Conditions	The dataset should be loaded and stored in database. The account number should be valid
Success Guarantee	The data is displayed
Frequency of occurrence	Quite often occurring use case(High-priority)

- **Downloading data as .Csv Format:**

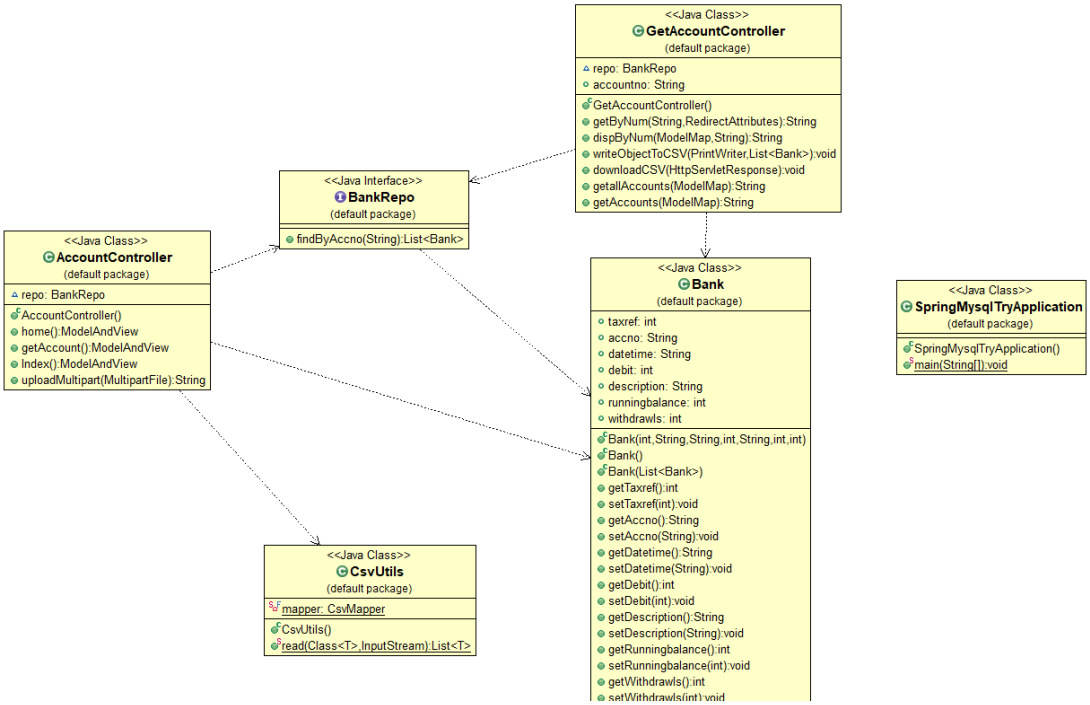
Use Case Headings	Explanation
Name	Downloading data as .Csv files
Scope	Downloading data
Level	Customer goal
Primary Actor	Customer
Pre-Conditions	The dataset should be available in database
Success Guarantee	The data is downloaded as .csv file
Frequency of occurrence	Low priority

5 Class Diagram :

5.1 Conceptual model :

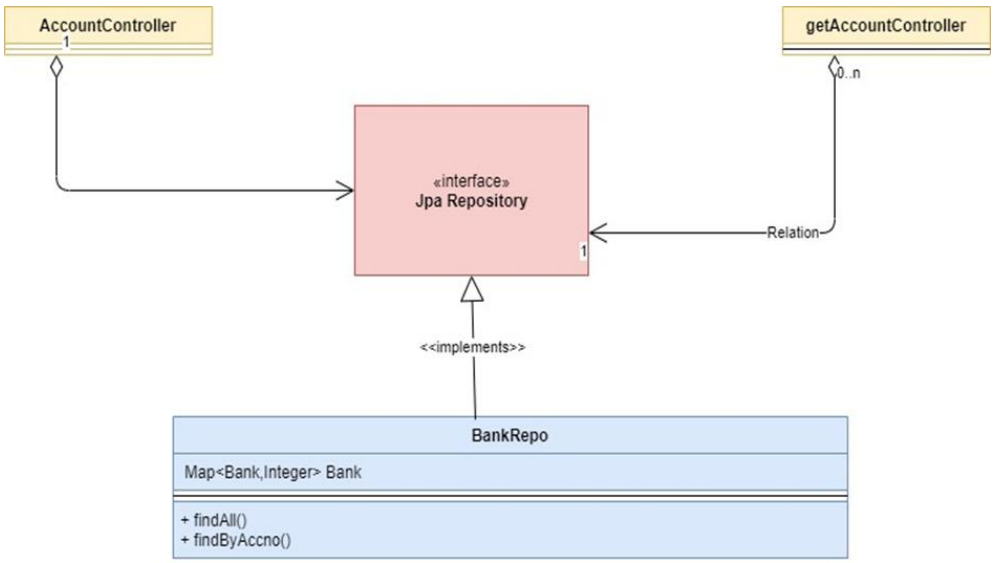


5.2 Design Model:

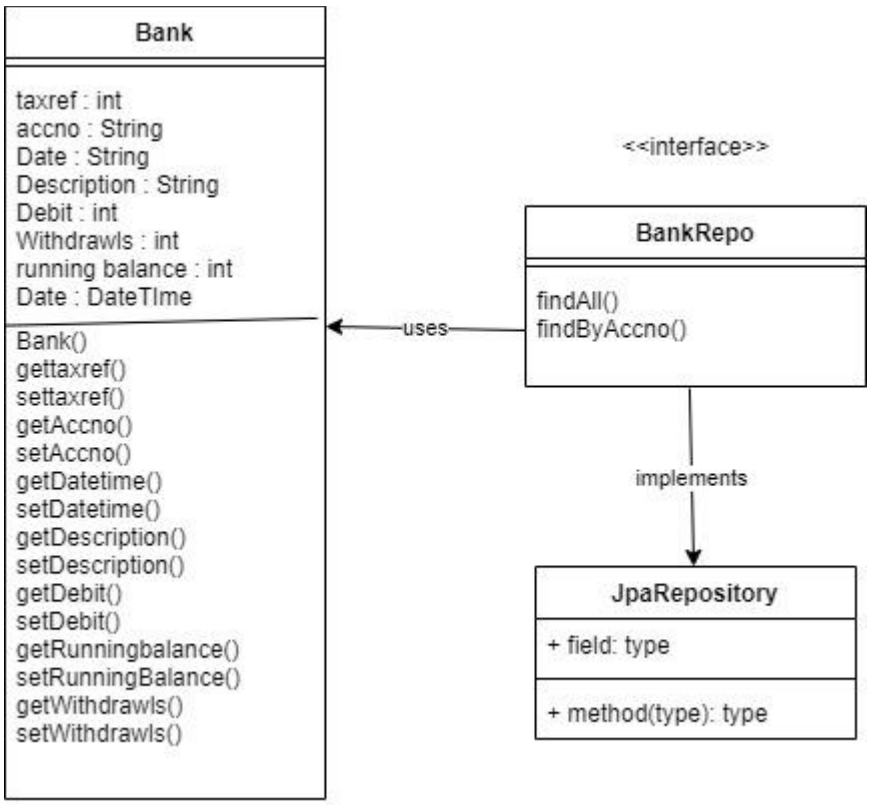


6 Design Patterns:

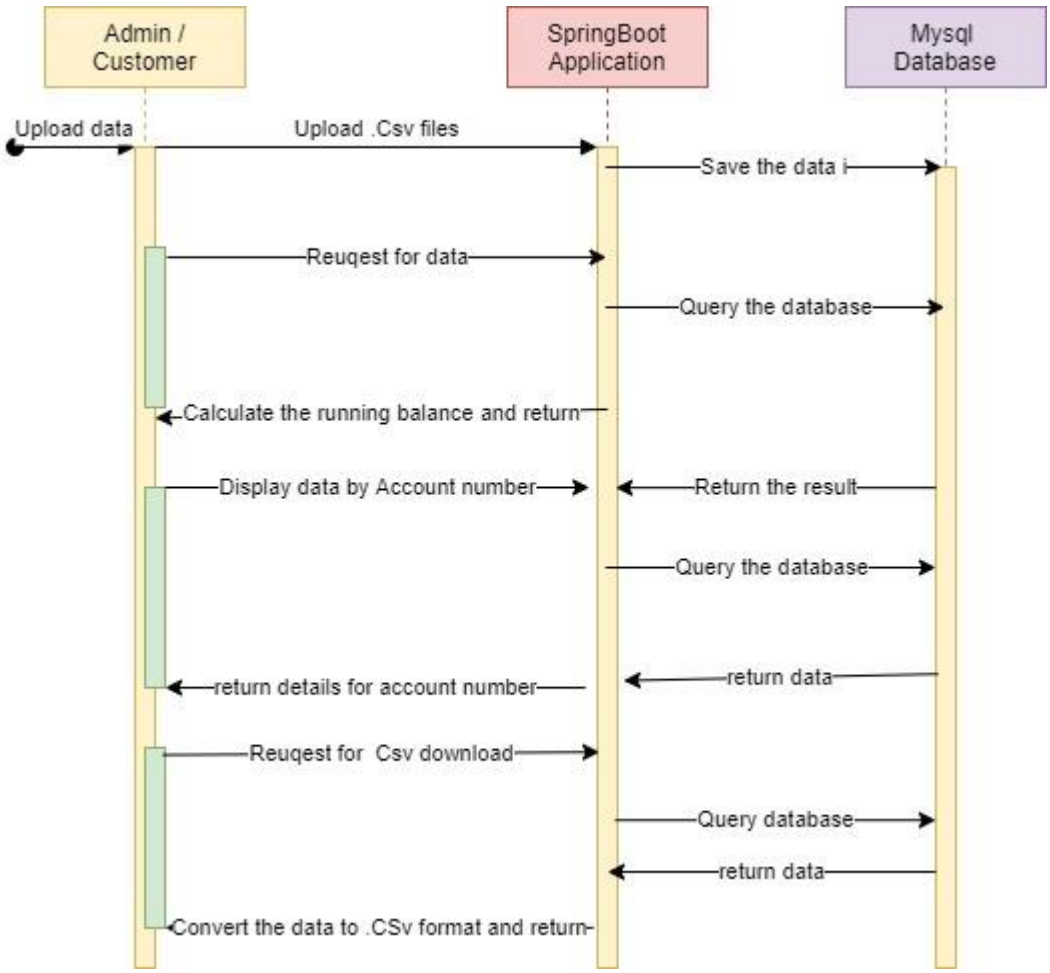
6.1 Dependency Injection



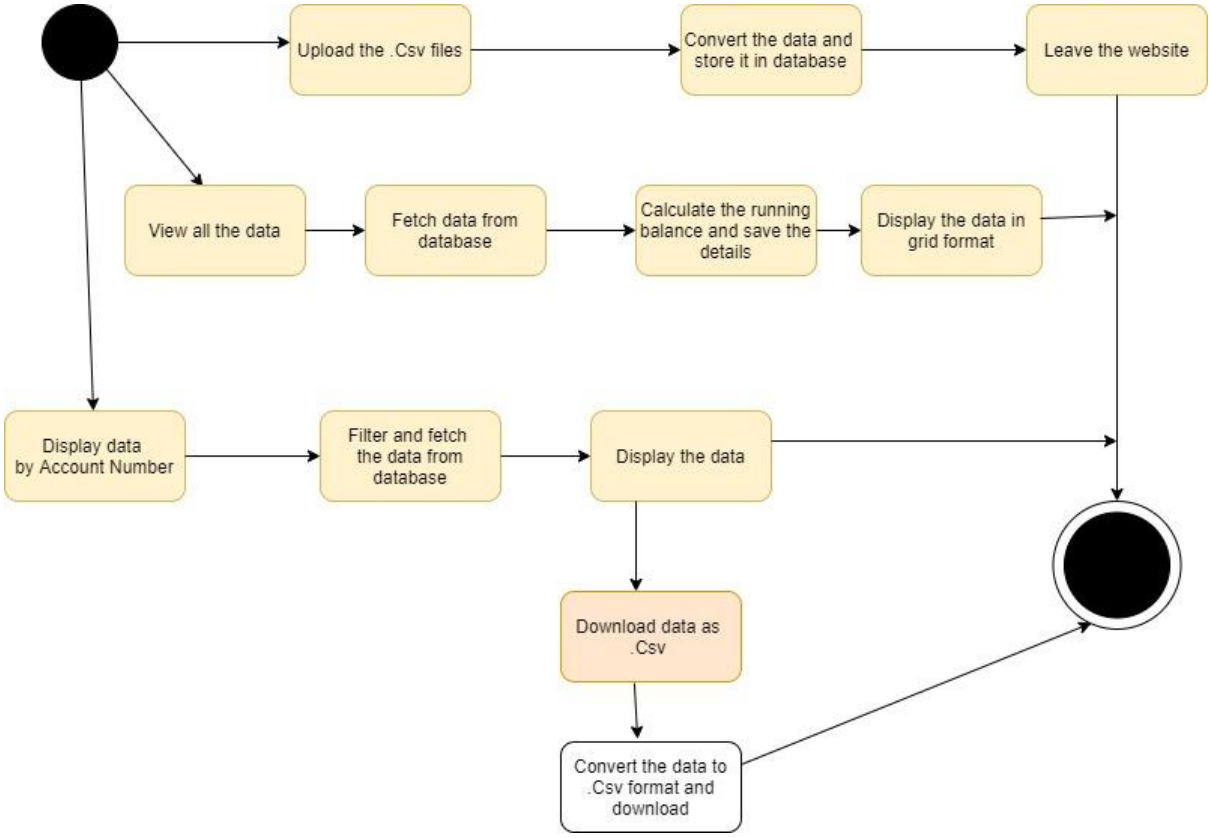
6.2 Data Access Object



7 Sequence Diagram:



8 State Chart Diagram:



9 Advantages and utilities:

- The project provides an user-friendly dashboard for maintaining all the transaction related data.
- The data can be uploaded in raw format as **.Csv File.**
- The running balance is calculated in a very efficient manner using the **HashMap data structure** available in java.
- The login facility provides additional security and restricts the access of certain pages only for the admin.
- The end-user on the other hand, can view their transaction details at any point of time.
- Also, the end-user can download their transaction details as **.Csv files** for reference.

10. Existing Systems:

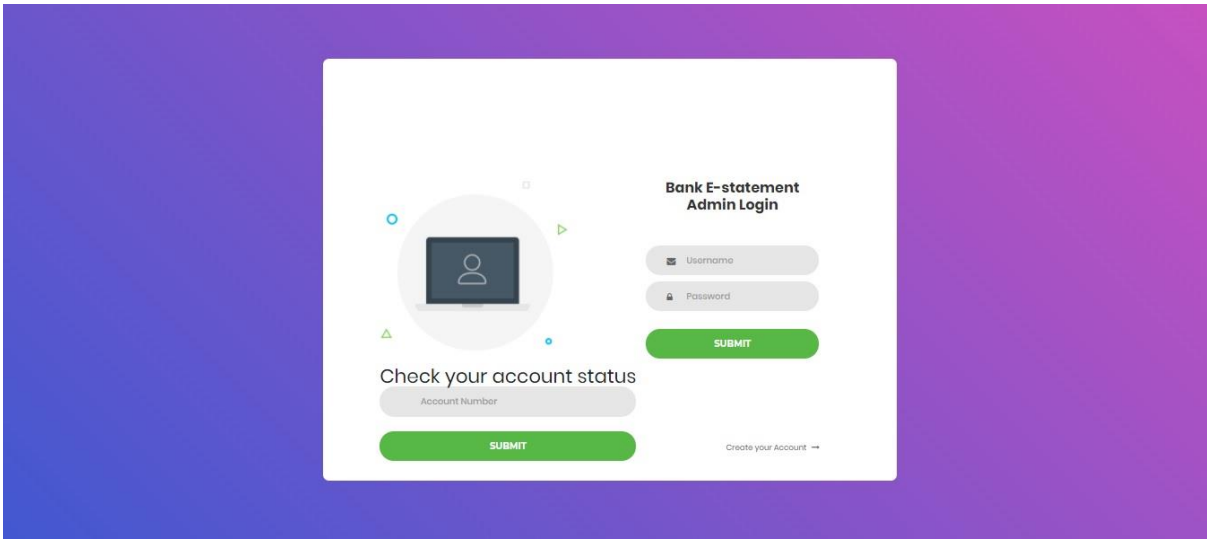
- **ICICI online banking:** the ICICI oneline banking web portal provides a way for the customers to check their transaction details as well as provides other banking facilities such as online transaction,etc.
- **SBI Net Banking :** The SBI net Banking facility allows the users to check their banking and transaction details online as well as keeps the users connected by sending the transaction details as message to the registered mobile number.

11 Improvements that can be made in the proposed system:

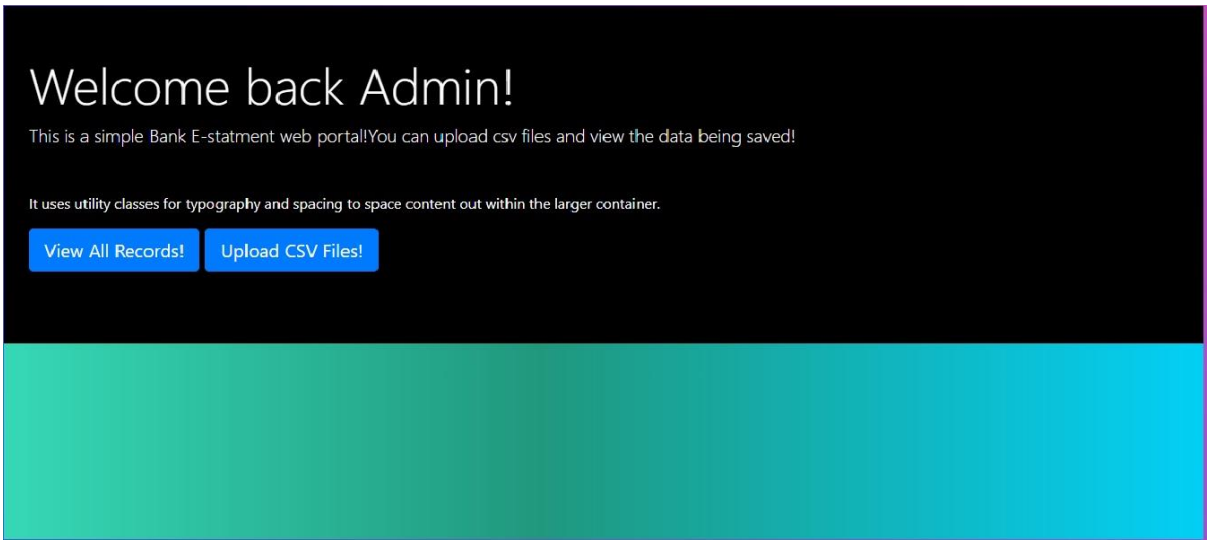
- The password details of each user can be made more secure by using a good encrypting method.
- The date range filter can be added to make it more user-friendly.
- Multiple data upload has to be implemented to make it more efficient and easy to use.

12 SnapShots of working:

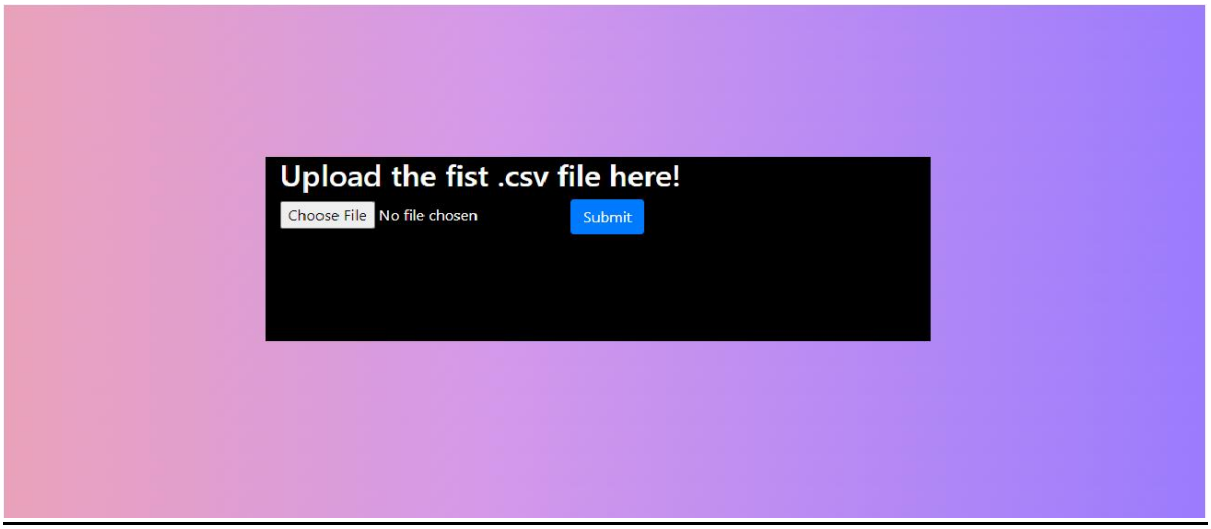
- **Admin Login:**



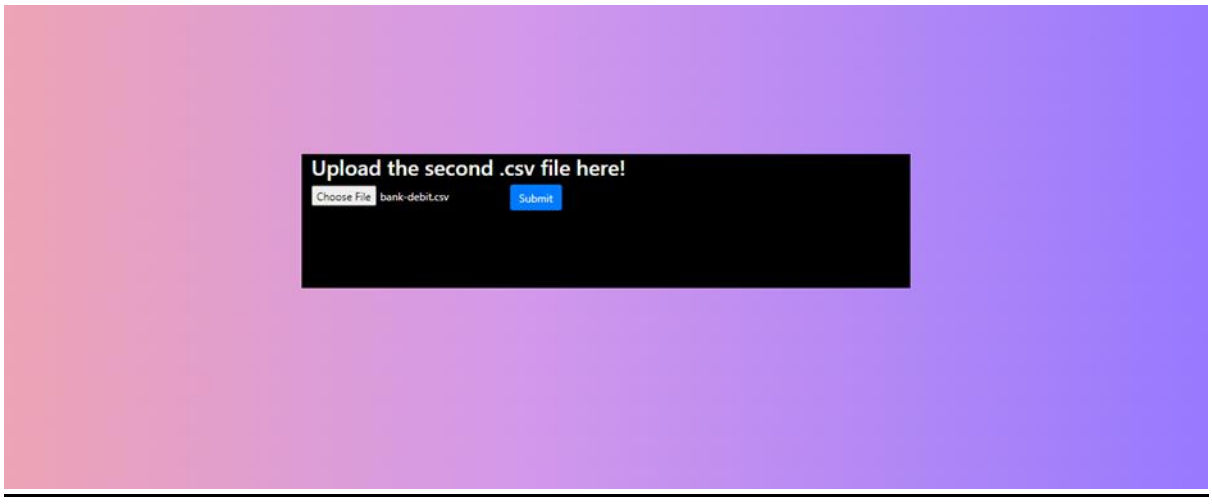
- **Admin Dashboard:**



- **CSV file upload-1:**



- **CSV File upload-2:**

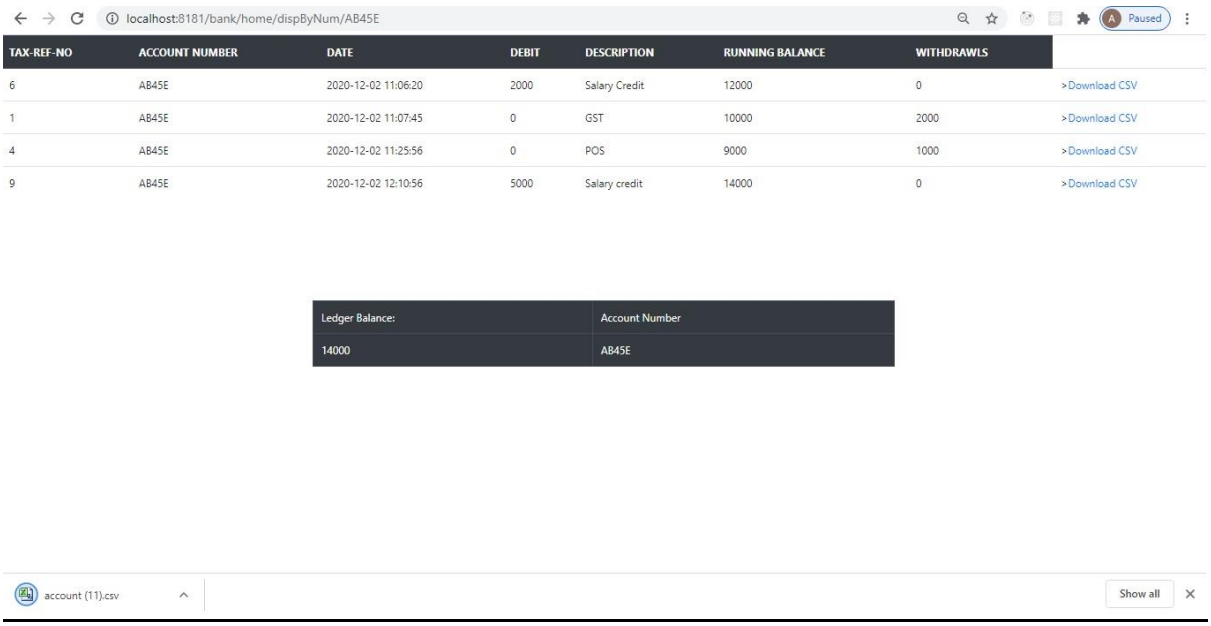


- **All the records with calculated running balance:**

All Record Details!

TAX-REF-NO	ACCOUNT NUMBER	DATE	DEBIT	DESCRIPTION	WITHDRAWLS	RUNNING BALANCE
6	AB45E	2020-12-02 11:06:20	2000	Salary Credit	0	12000
1	AB45E	2020-12-02 11:07:45	0	GST	2000	10000
7	BCD45A	2020-12-02 11:08:56	4000	Banc credit	0	24000
2	BCD45A	2020-12-02 11:11:56	0	WithDraw	3000	21000
4	AB45E	2020-12-02 11:25:56	0	POS	1000	9000
8	EFg456D	2020-12-02 12:06:56	0		0	15000
9	AB45E	2020-12-02 12:10:56	5000	Salary cred t	0	14000
5	BCD45A	2020-12-02 12:11:36	0	POS	500	20500

- **Displaying data according to account number:**



13 Conclusion: Thus the proposed application can be used in many banking sectors for displaying the transaction details and acts as a dashboard for both admin as well as customers . Furthermore, the application can be developed into an **API** service and can be used in various banking firms by changing the application necessity according to the firm. The application achieves most of the functionalities stated.

Github Link : https://github.com/Aditya-sairam/Bank-Application