INSTITUTE FOR ADVANCED COMPUTING AND SOFTWARE DEVELOPMENT AKURDI, PUNE

Documentation On

**“E-BANKING”**

PG-DAC SEP 2022

*Submitted By:*

**Group No: 44**

# Vaishnavi Ingole (229110)

# Sandhya Negi (229082)

**Mr. Rohit Puranik Mrs. Geeta Darunte**

**Centre Coordinator Project Guide**

# Table of Contents

1. [Introduction 3](#_TOC_250013)

[Problem Statement 3](#_TOC_250012)

Aim & Objectives 3

1. [Overall Description 4](#_TOC_250011)

Proposed Methodology 4

[Operating Environment 4](#_TOC_250010)

[Design and Implementation Constraints](#_bookmark0) 5

1. [Requirements Specification 6](#_TOC_250009)

External Interface Requirements 6

1. System Diagram 7

[Activity Diagram 7](#_TOC_250008)

[Data Flow Diagram 9](#_TOC_250007)

[Class Diagram 12](#_TOC_250006)

[Use Case Diagram 13](#_TOC_250005)

[ER Diagram 15](#_TOC_250004)

1. [Table Structure 16](#_TOC_250003)

User 16

Primary account 16

Savings account 16

Primary Transaction 17

Savings Transaction 17

Appointment 17

Recipient 18

1. [Conclusion 19](#_TOC_250002)

[Future Scope 19](#_TOC_250001)

1. [References 20](#_TOC_250000)

**List of Figures**

Figure 1 Admin Activity Diagram 7

Figure 2 User Activity Diagram 8

Figure 3 Level 0 Data Flow Diagram 9

Figure 4 Level 1 Data Flow Diagram 9

Figure 5 Level 2 Data Flow Diagram for Admin 10

Figure 6 Level 2 Data Flow Diagram for User 11

Figure 7 Class Diagram 12

Figure 8 Use Case Diagram for Admin 13

Figure 9 Use Case Diagram for User 14

Figure 10 ER (MySQL Auto Generated) 15

Figure 11 ER Diagram 15

**Introduction:**

# INTRODUCTION.

An e-banking is designed to provide online banking services to customers. It involves the development of a web-based platform that allows users to perform various banking activities, such as viewing account balances, withdraw, deposit and transferring funds between accounts,

The main objective of an e-banking project is to offer customers the convenience of accessing banking services anytime and anywhere they have an internet connection. This eliminates the need for physical visits to bank branches, allowing customers to manage their finances more efficiently.

Overall, an e-banking project can help banks improve customer satisfaction, reduce costs, and increase revenue by providing an efficient and convenient online banking service.

## Problem Statement:

Traditional banking services require customers to visit bank branches and interact with bank staff to perform various transactions. This can be time-consuming, inconvenient, and costly for both banks and customers. Additionally, with the rise of digital technologies, customers expect to access banking services anytime and anywhere through their devices.

However, many banks still rely on outdated systems and processes that make it difficult to offer online banking services. Some of the common problems faced by banks when implementing e-banking services include security concerns, outdated technology, inadequate infrastructure, and resistance to change from customers and staff.

As a result, banks need to find ways to overcome these challenges and provide secure, efficient, and convenient e-banking services to their customers. This requires a comprehensive strategy that addresses the technical, operational, and cultural aspects of e-banking implementation.

## Aims and Objective:

The aim of e-banking is to provide customers with a convenient and secure platform for accessing banking services anytime and anywhere they have an internet connection. The objective of e-banking are as follows:

1. Convenience: The primary objective of e-banking is to provide customers with a convenient and user-friendly platform for accessing banking services. Customers should be able to perform various transactions through the online platform.
2. Efficiency: E-banking aims to improve the efficiency of banking services by automating many processes, reducing the need for manual intervention, and enabling faster transactions. This can help banks reduce their operational costs and improve their overall productivity.
3. Accessibility: E-banking aims to make banking services more accessible to customers who may not be able to visit bank branches due to physical disabilities or geographical constraints.
4. Innovation: E-banking aims to foster innovation in banking services by enabling banks to introduce new products and services quickly and easily. The online platform should be flexible and scalable to accommodate new features and functions as required by changing customer needs and market trends.

**Proposed Methodology:**

# OVERALL DESCRIPTION.

The proposed methodology for e-banking can be broken down into several stages:

1. Requirement Analysis: The first step in the methodology for e-banking is to analyse the requirements of the customers and the bank. This involves gathering information about the types of transactions that customers want to perform, the features and functions that are needed, and the user interface that will be most convenient for customers.
2. Design and Development: Once the requirements have been analysed, the next step is to design and develop the e-banking platform. This involves developing the architecture, designing the user interface, integrating various systems and modules, and testing the platform to ensure that it meets the requirements and is free from defects.
3. Implementation: After the e-banking platform has been developed, the next step is to implement it. This involves deploying the platform on the bank's servers.
4. Testing and Quality Assurance: The e-banking platform must be tested rigorously to ensure that it is secure, reliable, and user-friendly. This involves conducting various tests such as functional testing, performance testing, and user acceptance testing.

## Operating Environment:

Server Side:

**Processor:** Intel® Xeon® processor 3500 series

**HDD:** Minimum 500GB Disk Space

**RAM:** Minimum 4GB **OS:** Windows 10, Linux 6 **Database:** MySQL

Client Side (minimum requirement):

**Processor:** Intel Dual Core

**HDD:** Minimum 80GB Disk Space

**RAM:** Minimum 2GB

**OS:** Windows 7, Linux

## Design and Implementation Constraints:

* The application will use Spring-Boot and React as main web technologies.
* HTTP is used for communication between clients and servers over the internet
* Several types of validations make this web application a secured one and SQL Injections can also be prevented.
* Since E-Banking is a web-based application, internet connection must be established.
* The E-Banking will be used on PCs and will function via internet or intranet in any web browser.

**External Interface Requirements:**

# Requirements Specification.

User Interfaces:

* + All the users will see the same page when they enter in this website. This page asks the users a username and a password.
  + After being authenticated by correct email and password, user will be redirect to their corresponding profile where they can do various activities.
  + The user interface will be simple and consistence, using terminology commonly understood by intended users of the system. The system will have simple interface, consistence with standard interface, to eliminate need for user training of infrequent users.

Hardware Interfaces:

* + No extra hardware interfaces are needed.
  + The system will use the standard hardware and data communication resources.

This includes, but not limited to, general network connection at the server/hosting site, network server and network management tools.

Application Interfaces:

**Web Browser:**

The system is a web-based application; clients need a modern web browser such as Mozilla Firebox, Internet Explorer, Opera, and Chrome. The computer must have an Internet connection in order to be able to access the system.

Communications Interfaces:

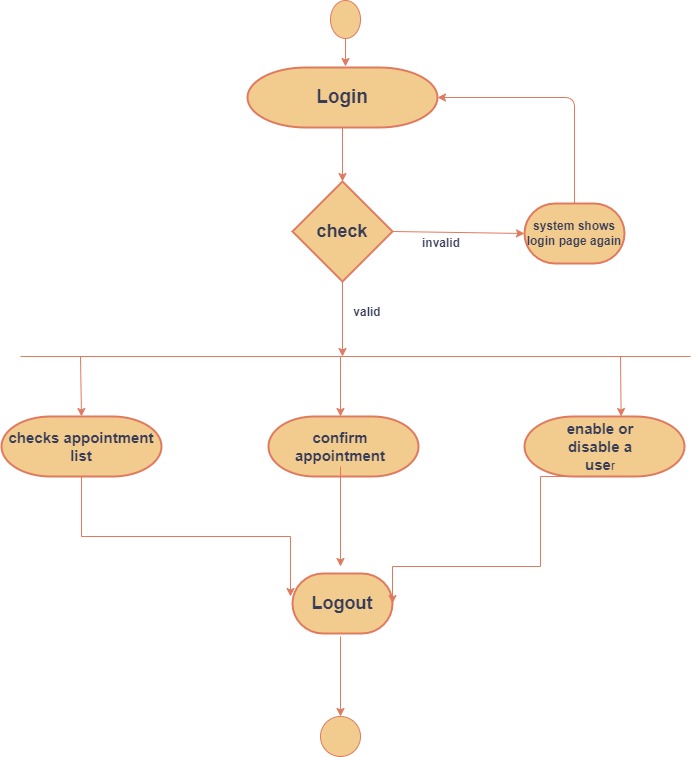
* + This system uses communication resources which includes but not limited to, HTTP protocol for communication with the web browser and web server and TCP/IP network protocol with HTTP protocol.
  + This application will communicate with the database that holds all the user information. Users can contact with server side through HTTP protocol by means of a function that is called HTTP Service. This function allows the application to use the data retrieved by server to fulfil the request fired by the user.

# System Diagrams.

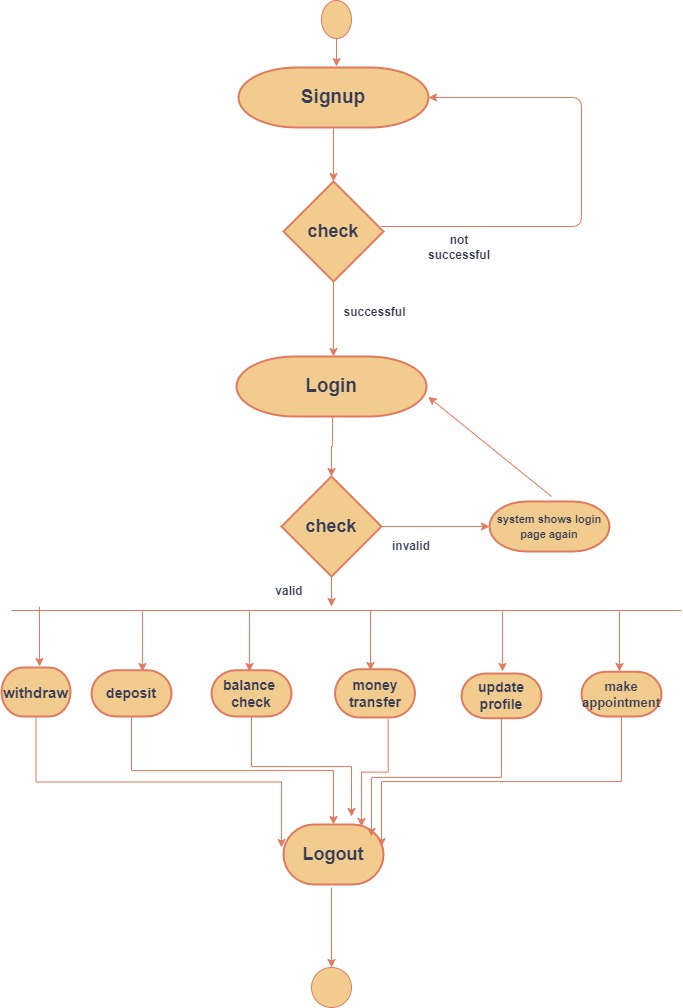
## Activity Diagram:

## ­­

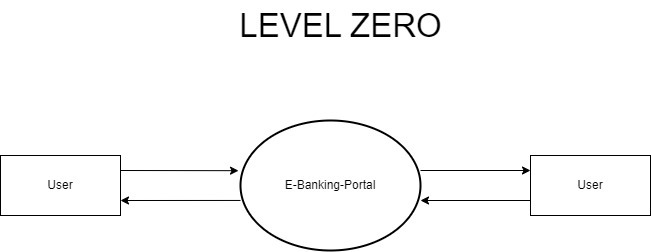
*Admin Activity:

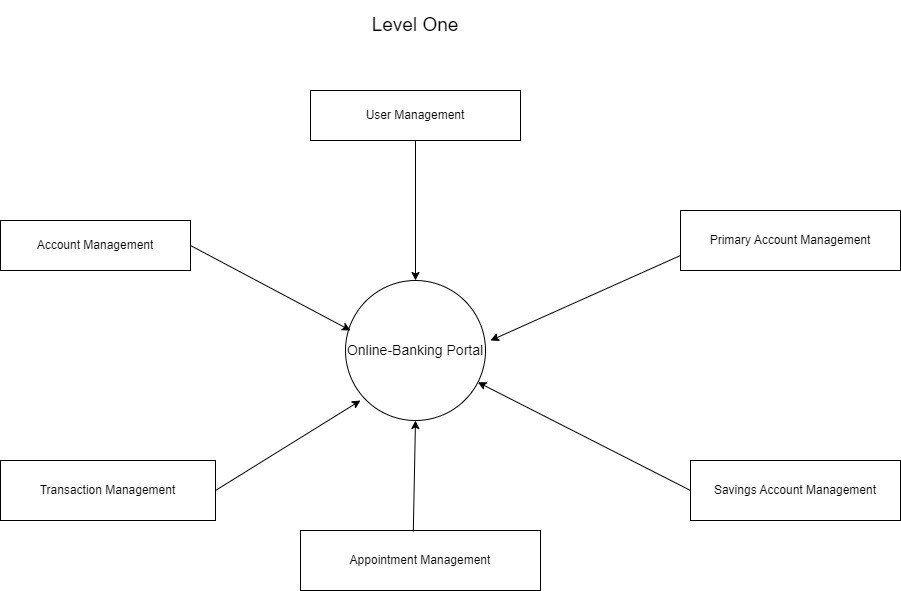


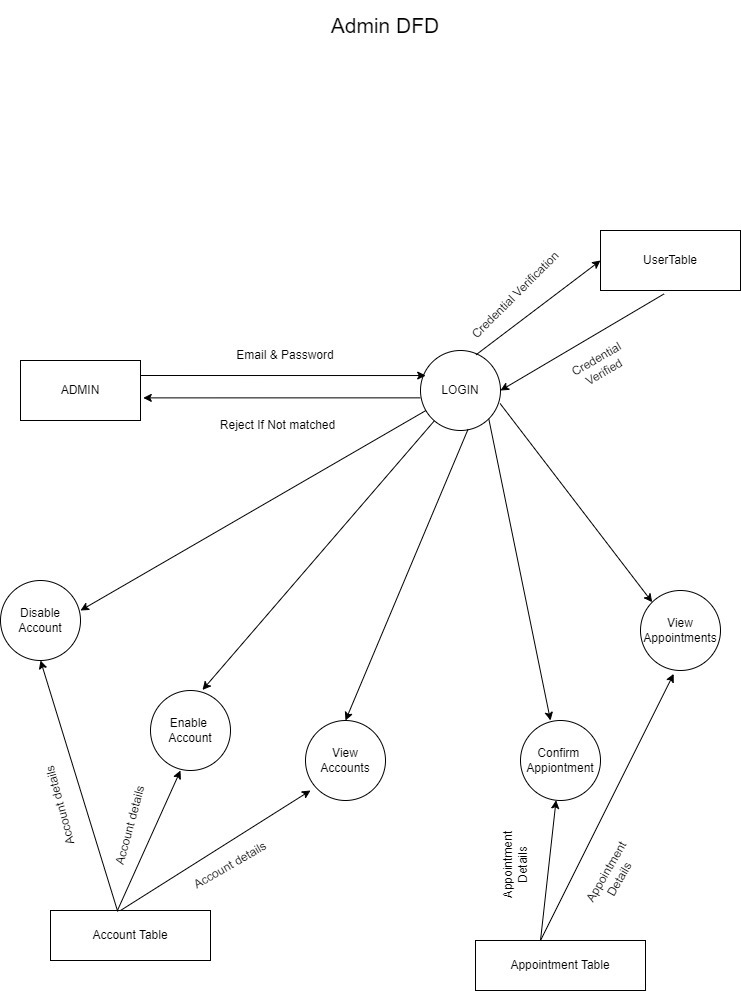
*User Activity:

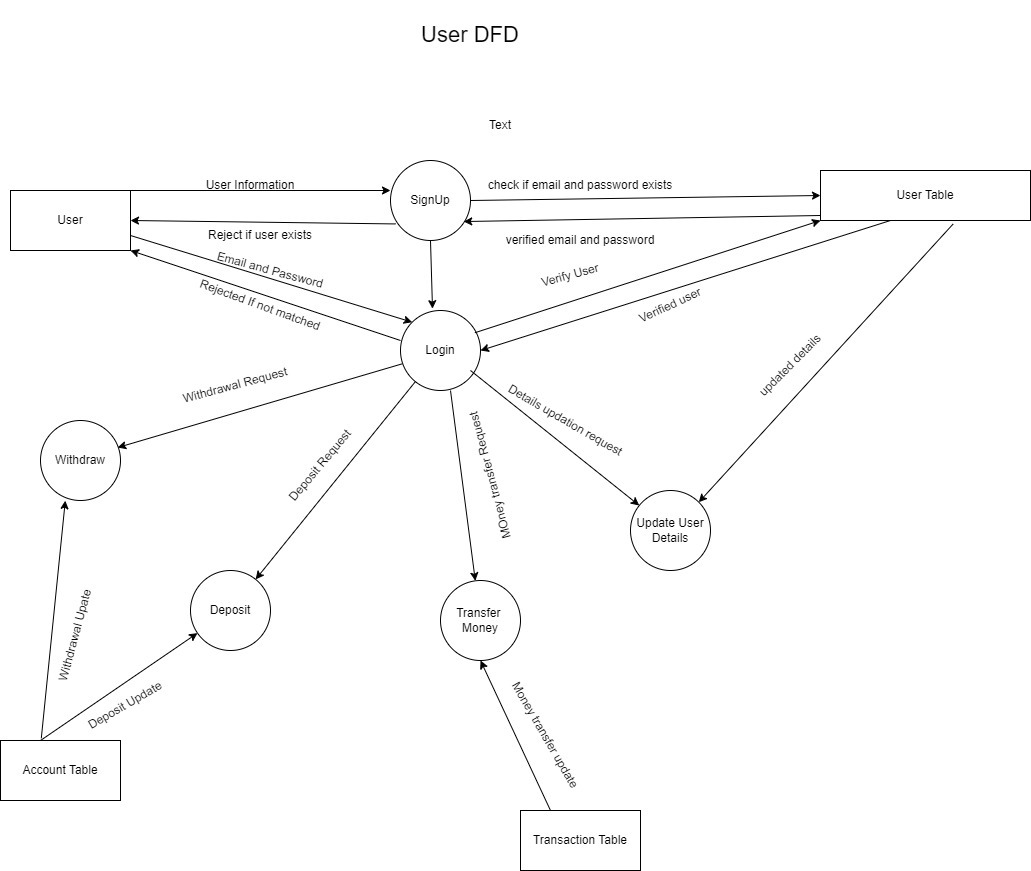


## Data Flow diagram:

****

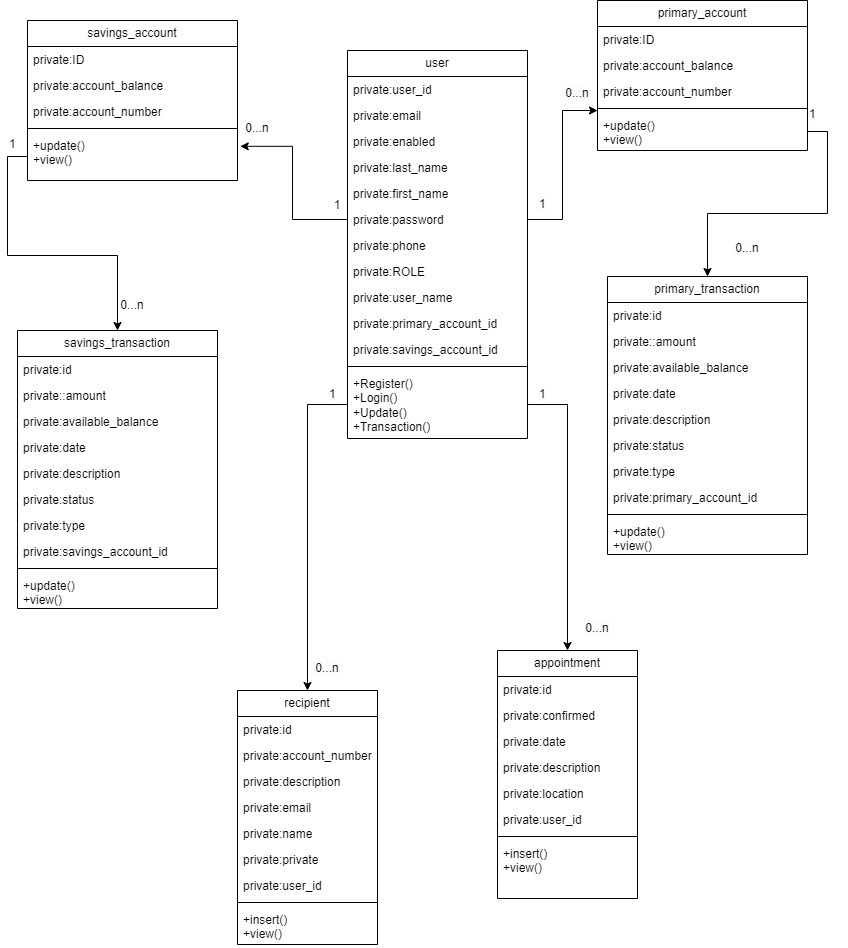






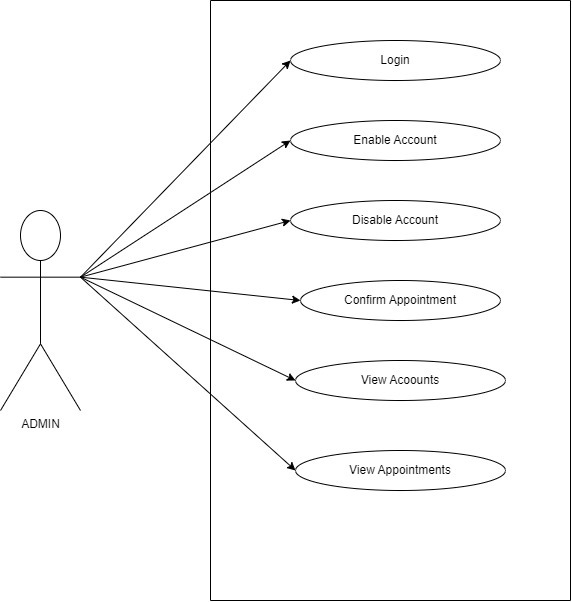
## 

## Class Diagram:

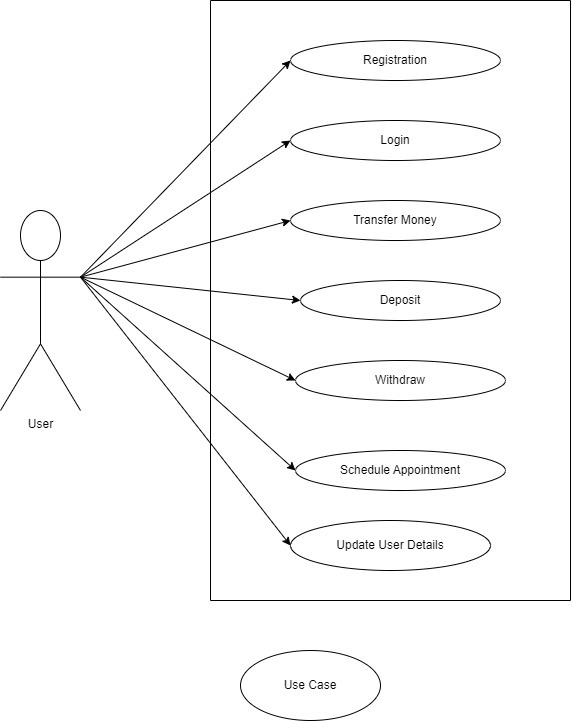
****

## Use Case Diagram:

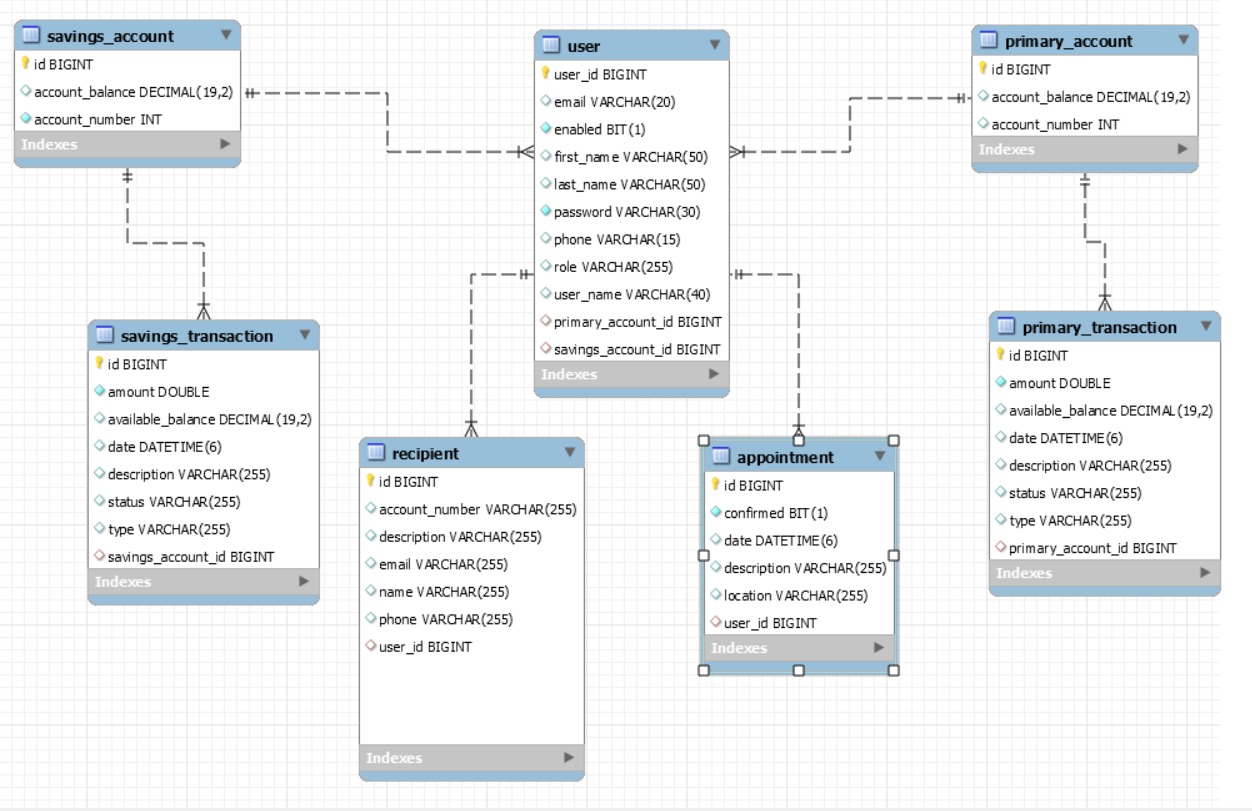
*admin Use-case:

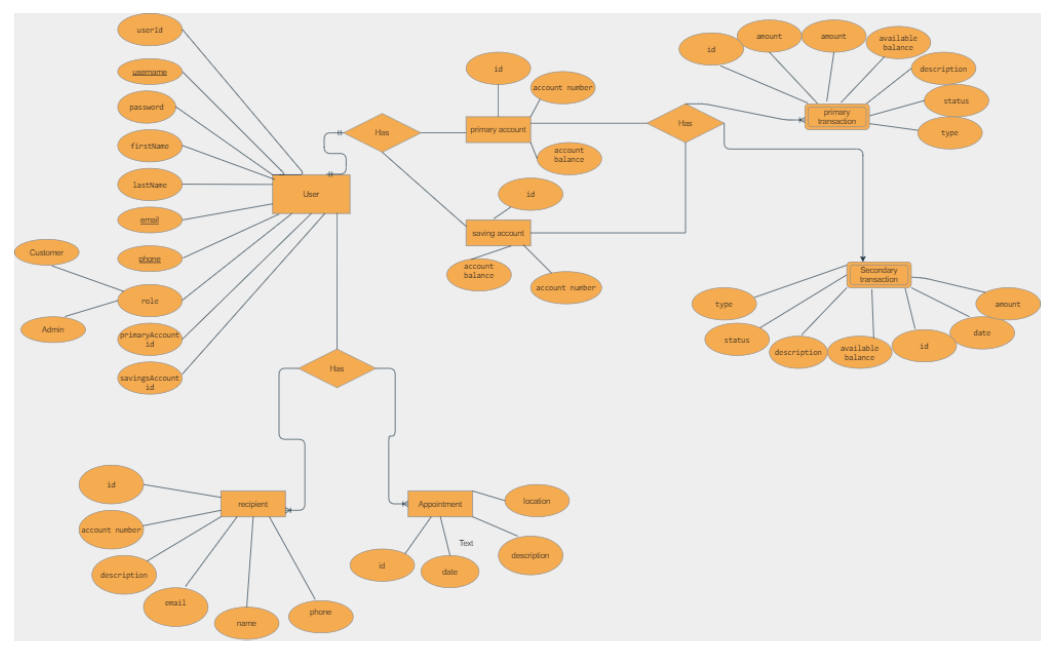


*user Use-case:



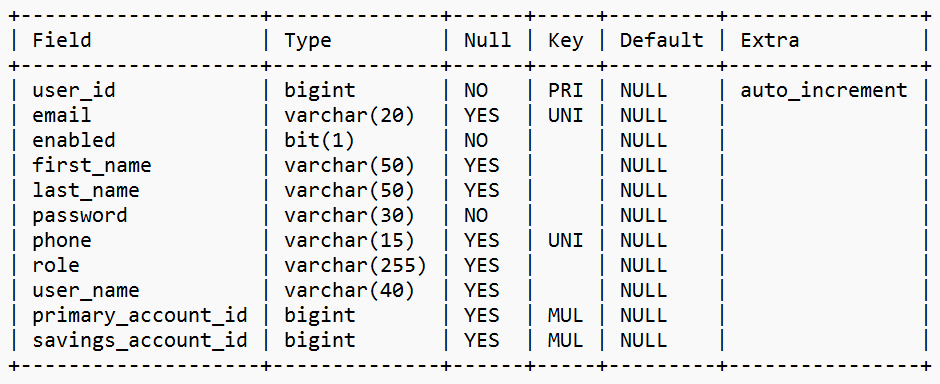
## ER Diagram:

****

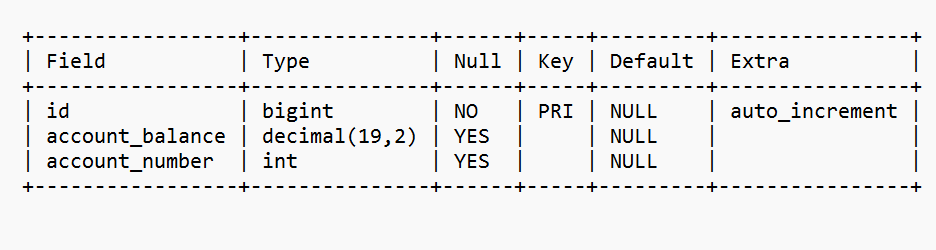


# Table Structure.

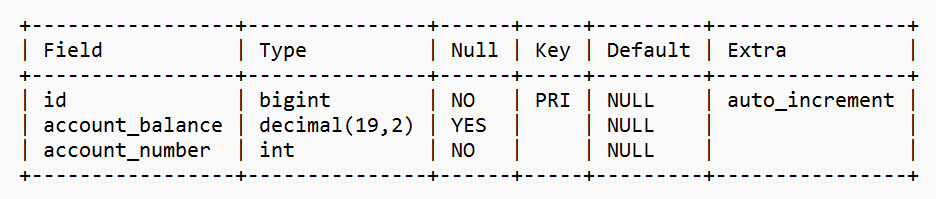
* + User:



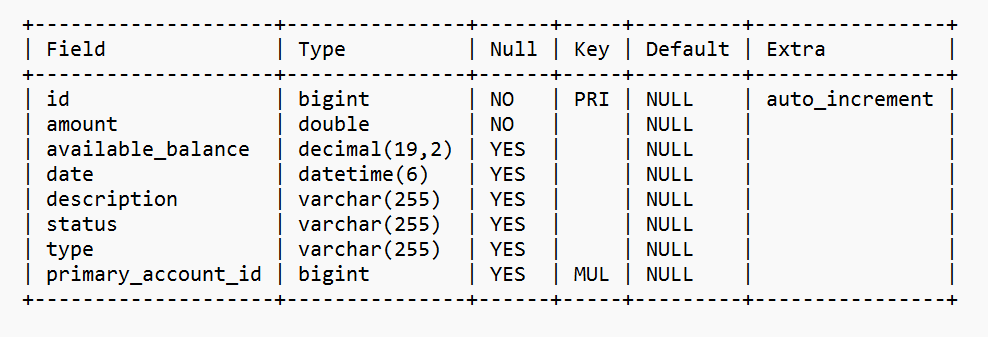
* + Primary\_account:



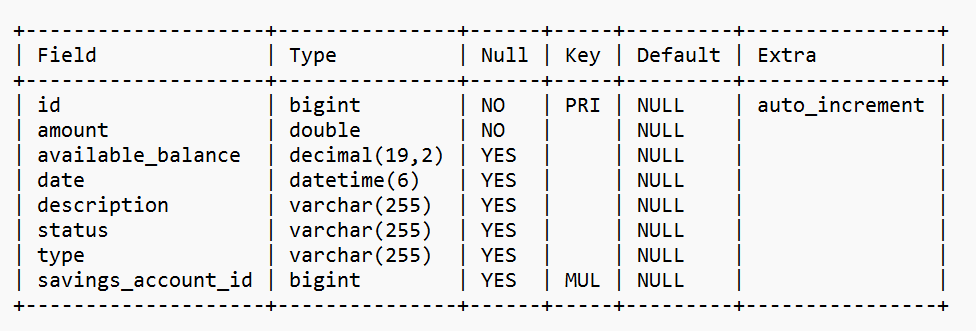
* + Savings\_account:



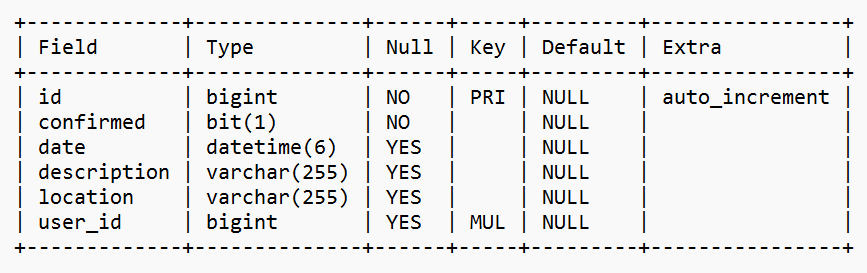
* + Primary\_transaction:



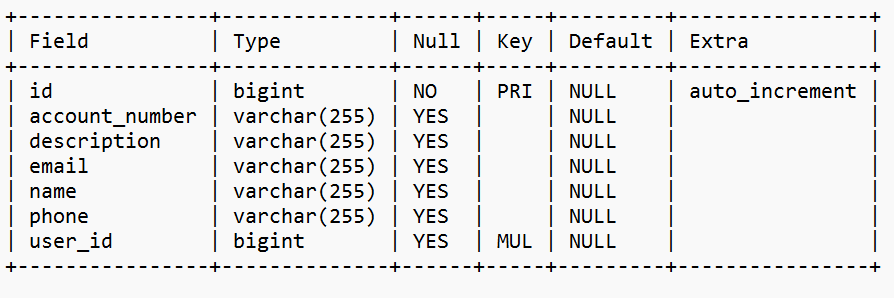
* + Savings\_transaction:



* + appointment:



* + recipient:



* + **Conclusion:**

# CONCLUSION

This project helps banks manage their operations efficiently and effectively. The system provides several benefits such as enhanced security, increased accuracy in transactions, real-time data access, and improved customer service. With the use of an e-bank management system, banks can reduce operational costs and streamline their processes, which in turn, can result in higher profitability. Additionally, this system allows customers to access banking services anytime and anywhere, making it more convenient for them. Overall, an e-bank management system is a valuable investment for any bank seeking to remain competitive in today's digital age.

## Future Scope:

This project can be enhanced further by adding payment gateway integration: Integration with Multiple Payment Gateways, Seamless Payment Processing, Enhanced Security, International Transactions.

The software is flexible enough to be modified and implemented as per future requirements.

# REFERENCES.

## References:

* <https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/>
* <https://www.w3schools.com/>
* <https://javaee.github.io/javaee-spec/javadocs/>
* <https://www.javatpoint.com/java-mail-api-tutorial>
* https://reactjs.org/docs/getting-started.html