Tech Saksham

Final Project Report

Web Track

“Banking Web Application”

“JNTUA College of Engineering Kalikiri”

Roll No.	Name
20KA5A0507	Masupalli Yuktha Priya

Trainer Name

R. Uma Maheshwari

ABSTRACT

This Project is aimed at developing an Online Banking for customer.

The system is an online application that can be accessed throughout the organization and outside as well with proper login provided. This application takes input from the user and allows us to carry out almost all banking operations like: Create account, Deposit money, Withdraw money, Check Balance and Transfer money to another account.

The entire project has been developed keeping in view of the distributed client server computing technology, in mind.

The user interfaces are browser specific to give distributed accessibility for the overall system.

At all proper levels high care was taken to check that the system manages the data consistency with proper business rules or validations. The database connectivity was planned using the latest “ Database connection” technology provided by MySQL. The authentication and authorization was cross checked at all the relevant stages.

INDEX

CHAPTER 1

INTRODUCTION

1.1 Overview

Internet Banking is all about knowing our customer need and provide them with the right service at the right time through right channel 24*7 day a week.

1.2 Feature

The Online Banking suite provides a global accounting foundation that provides the all private banks with electronic banking facilities. It allows client of private banks to carry out their day to day banking transactions.

- Withdrawal of amount by the client.
- Deposition of amount by the client.
- Faster balance enquiry.

1.3 Advantages

- Quick, authenticated access to accounts via the desktop.
- Easily scalable to grow with changing system requirement.
- Improved information security, restricting unauthorized access.
- Minimize Storage Space

1.4 Scope

The Online Banking project is widely applicable with private banks. It can even be used in industries for their personal transactions (working).

1.5 Future Work

- By adding new modules of different accounts like saving A/C, current A/C etc. to facilitate new customers/users.
- By the use of electronic media, “Digital Signature” on the card can be provided with the customer to make it secure and efficient.

CHAPTER 2

SERVICES AND TOOLS REQUIRED

2.1 Services Used

We will be creating a banking application using PostgreSQL, Express, React and Node.js.

1. How to work with PostgreSQL database from Node.js?
2. How to secure your APIs using JWT authentication?
3. How to maintain login session until user manually performs a logout?
4. How to perform CRUD operations in PostgreSQL from Node.js?
5. How to dynamically generate pdf to get list of transactions within selected date range?

2.1.1 Liberty Profile

Following are the functional needs of the software:-

1. Customer must have a valid user ID and password to login to the system.

2. After the valid user logs in, the system shows the present balance in that particular account number.

3. Customer can perform transactions like deposit and withdrawal from his account.

4. Proper help to be provided as and when requested by the customer.

2.2 Tools and Softwares used

Software Required:

The project is implemented using Html, CSS, React JS, Node JS,Express framework and Postgres SQL.Software's required in the creation and execution of the project is Visual Studio Code or Eclipse.Desired platform i.e. Linux ,windows or any operating system.

Hardware Required:

Any System with i7 processor, 32MB RAM, 1GB Hard Disk is sufficient. Its web based software so computers connected with any kind of mode (wireless, LAN connected etc) will suit its requirements.

2.2.1 CLIENT SIDE VALIDATION

Various client side validations are used to ensure on the client side that only valid data is entered. Client side validation saves server time and load to handle invalid data. Some checks imposed are:

- JavaScript in used to ensure those required fields are filled with suitable data only. Maximum lengths of the fields of the forms are appropriately defined.
- Forms cannot be submitted without filling up the mandatory data so that manual mistakes of submitting empty fields that are mandatory can be sorted out at the client side to save the server time and load.

2.2.2 SERVER SIDE VALIDATION

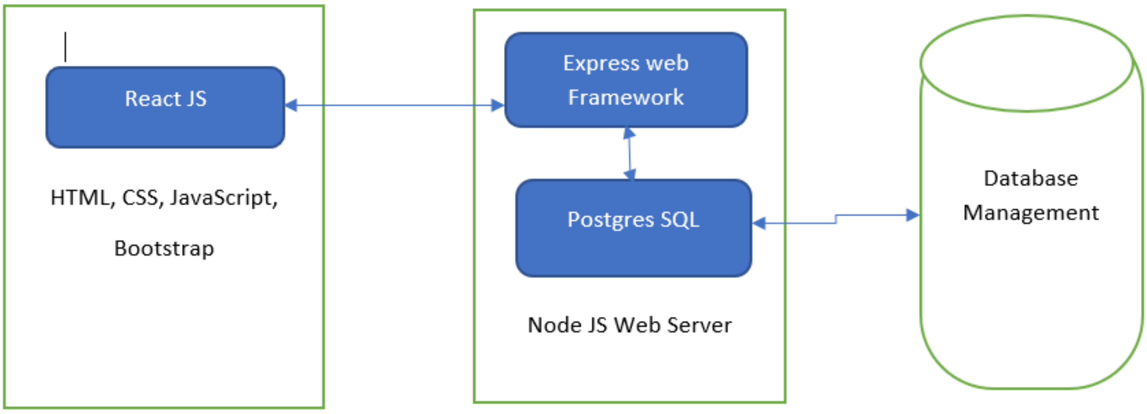
Some checks cannot be applied at client side. Server side checks are necessary to save the system from failing and intimating the user that some invalid operation has been performed or the performed operation is restricted. Some of the server side checks imposed is:

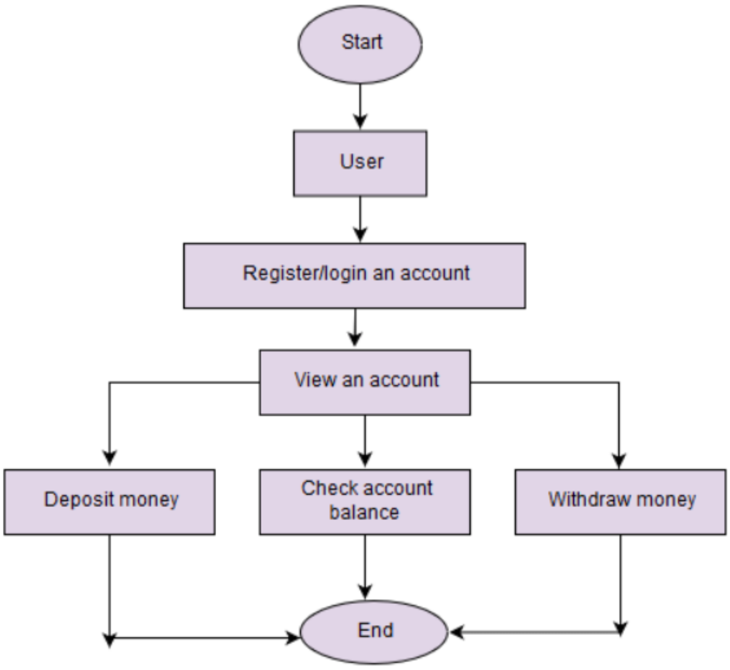
- Server side constraint has been imposed to check for the validity of primary key and foreign key. A primary key value cannot be duplicated. Any attempt to duplicate the primary value results into a message intimating the user about those values through the forms using foreign key can be updated only of the existing foreign key values.
- User is intimating through appropriate messages about the successful operations or exceptions occurring at server side.

CHAPTER 3

PROJECT ARCHITECTURE

3.1 Architecture





CHAPTER 4

ARCHITECTURE BLOCKS DETAIL WORKING

4.1 Blocks

We’ll be building a personal banking web app using the following technologies.

- MongoDB for our database
- Express & Node for our backend
- React for our frontend
- Redux for state management
- Plaid for allowing users to link their bank accounts

The app will allow users to

- Link multiple bank accounts
- Remove bank accounts
- View transactions from all linked accounts in a searchable and filterable data table
- User links a bank account within app, causing our app’s public key to be sent to api.
- Plaid responds with a public token, which is unique for each sign in and expires in 30 minutes
- We send our public token to our back-end server, exchanging it for an **access token** and **item id** (each bank account has a unique access token and item id)
- We’ll save this access token, item id and a few other fields in our database (while checking for duplicate account).

CHAPTER 5

PROJECT BUDGET

Sr. No	**Cloud Services and Coding Cost**	**Single Price (Rs)**	**Total**
Total			

CONCLUSION

This project developed, incorporated all the activities involved in the browsing centre.

It provides all necessary information to the management as well as the customer with the use of this system; the user can simply sit in front of the system and monitor all the activities without any physical movement of the file. Management can service the customers request best in time.

The system provides quickly and valuable information. These modules have been integrated for effective use of the management for future forecasting and for the current need.

REFERENCES

https://www.researchgate.net/publication/361465446_Application_using_MERN_Stack

https://blog.bitsrc.io/build-a-full-stack-banking-web-app-with-plaid-the-mern-stack-508914ce5694

https://javascript.plainenglish.io/create-a-fullstack-banking-application-using-react-e8c96d74cd39

CODE

Code through Git Hub Repo Link:

https://github.com/yukthapriya/Banking-Web-Application.git