***Write ups***

**Project Introduction**

Online Bank Management System is very simple and entry level project developed in java using jdbc technology. I have used database to store all the details of customer account in a table. This system is a simple project and useful for those who want to learn jsp/servlet with jdbc.

I have implemented most of the feature of online banking in it. User can login, do withdrawal, Request a Deposit amount, Transfer amount from one account to another account, check his/her current balance, and view reports of all the transaction.

The project is for beginner level students; developer, those who want to learn jsp/java. I have not use any servlet in this system and database connectivity is written in jsp/java. It should be noted that writing database connection or sql statement in jsp is not recommended.

**Project Synopsis on Internet Banking System**

The main aim of designing and developing this Internet banking System J2EE primarily based Engineering project is to provide secure and efficient net banking facilities to the banking customers over the internet. Java Server Pages, J2EE, Java, Enterprise Beans, Oracle database used to develop this bank application where all banking customers can login through the secured web page by their account login id and password. Users will have all options and features in that application like get money from western union, money transfer to others, make payments for electricity bills, mobile prepaid recharges, mobile bills, water bills etc, and send cash or money to inter banking as well as other banking customers by simply adding them as payees.

**Background of the problem statement:**

ICIN is one of the top banking firms that accepts deposits from the public for the purpose of lending loans to the public. It also invests an amount in securities.  
Recently, the business analysts noticed a drop in the number of customers of the bank. They found out that online banking systems of banks like AXIS and American Express are gaining more profits by eliminating middlemen from the equation. As a result, the team decided to hire a Full Stack developer who can develop an online banking web application with a rich and user-friendly interface.  
You are hired as one of the Full Stack Java developers and have been asked to develop the web application. The management team has provided you the requirements and their business model so that you can easily arrange different components of the application.

**Purpose**

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

**Features of the application:**

1. Registration
2. Login
3. Account transactions
4. Transfers
5. Savings details
6. Profile settings
7. Requesting cheque books

**Benefits of the System**

* Quick, simple, authenticated access to accounts via the desktop.
* Simply scalable to grow with changing system requirement.
* Global enterprise wide access to information.
* Improved data security, restricting unauthorized access.
* Minimize Storage Space

**Specifications**

THIS INTERNET BANKING supports the functionalities of banking services with secure login procedure to interact within the CUSTOMERS and DATABASE.

* + - Application Server (WebLogic 7.0)

J2EE(Enterprise Edition) Technology

**Recommended technologies:**

1. Database management: MySQL
2. Back-end logic: Java programming, SpringBoot framework
3. Front-end development: Angular 2, HTML/CSS,
4. Automation and testing technologies: Selenium and JUnit
5. DevOps and production technologies: Git, GitHub, Jenkins, Docker, and AWS

**Project development guidelines:**

* The project will be delivered within four sprints with every sprint delivering a minimal viable product.
* It is mandatory to perform proper sprint planning with user stories to develop all the components of the project.
* The learner can use any technology from the above-mentioned technologies for different layers of the project.
* The web application should be responsive and should fetch or send data dynamically without hardcoded values.
* The learner must maintain the version of the application over GitHub and every new change should be sent to the repository.
* The learner must implement a CI/CD pipeline using Jenkins.
* The learner should also deploy and host the application on an AWS EC2 instance.
* The learner should also implement automation testing before the application enters the CI/CD pipeline.
* The learner should use Git branching to separately perform the basic automation testing of application.
* The learner should make a rich front-end of the application, which is user- friendly and easy for the user to navigate through the application.
* There will be two portals in the application, namely the admin and user portal.

**Admin Portal:**

It deals with all the back-end data generation and product information. The admin user should be able to:

* Authorize the roles and guidelines for the user
* Grant access to the user regarding money transfer, deposits, and withdrawal
* Block the user account in case of any threat
* Authorize the cheque book requests

**User Portal:**

It deals with the user activities. The user should be able to:

* Register or log in to the application to maintain a record of activities
* Deposit and withdraw money from the account
* View transactions and balance in the primary and savings account
* Transfer funds between different accounts and add recipients
* Request cheque books for different accounts

**The below are the Keywords which used to get the following statements**

* getBalance(): It simply returns available bank account balance.
* add(amt): It is used to deposit the amount or update the balance(amount).
* withdrw(amt): It removes the amount from balance amount from bank account.
* acceptInput(): It is used to ask n take input from user.
* verify(): This method is used to check if the login was successful or not successful. If the log is successful then use the banking class to fetch balance and then show a menu-driven option to the user to select the menu.
* If the login isn’t successful then show a proper message to a user by using the invalid\_transaction Customized Exception class.
* invalid\_transaction: This exception class is used to handle all the user errors.

**Conclusion**

The application illustrated or demonstrates the way to develop an online banking system by using interactive web client by using JSP, Servlet with safer way to access &encapsulate database by EJB component. This suggests the application server simply deployable and accessible.