

JSW One Banking

i JSW ONE BANKING is a one stop solution project for all your banking transaction needs which gives users to deposit money and withdrawal it within few seconds. Some more additional features are there which makes it more unique than any other banking services.

Below are the description of the whole project.

Overview

Project summary

JSW One Banking Project is banking system which provides all functional of a bank where a user can register themselves and take the use of all the banking functions like deposit money and withdrawal etc.


Link of the flow diagram : [Flow Chart Diagram of JSW ONE Banking](#)


Project team

Project Lead	Self Based Project
Kavish Kr. Sharma	Kavish Kr. Sharma

Project status

Project has been successfully completed within the given time with all the functionality required.

 On Track

 Potentially At Risk

 At Risk / Blocked

♦ ♦ ♦

Tech Stack used

It's a Full Stack Project where we have implemented Frontend with HTML,CSS,JavaScript, and backend using JAVA and Servlet & Mysql for Database.

Tech Stack	For What Purpose	
HTML,CSS, JavaScript	To Implement the Frontend development	
Servlet	To Create the Web Application	
Java	For Implementing the whole project	
MYSQL	To Create the database and store the information	

◆ ◆ ◆

More functionality can be added to “enhance the project”

Many features have been planned to improve this project as this is just a basic project we would like to add some more features for sure mentioned below.

- 1.By adding new modules of different accounts like saving A/C, current A/C etc. to facilitate new customers/users.
2. By the use of electronic media, “Digital Signature” on the card can be provided with the customer to make it secure and efficient.
- 3.Let Users to have more security towards their password by encrypting and decryption them.

Software Requirement & Specification

Communication is key to successful project delivery, especially when many stakeholders are involved. This table gives an example of visibility over the communication channels for the project and expected cadence.

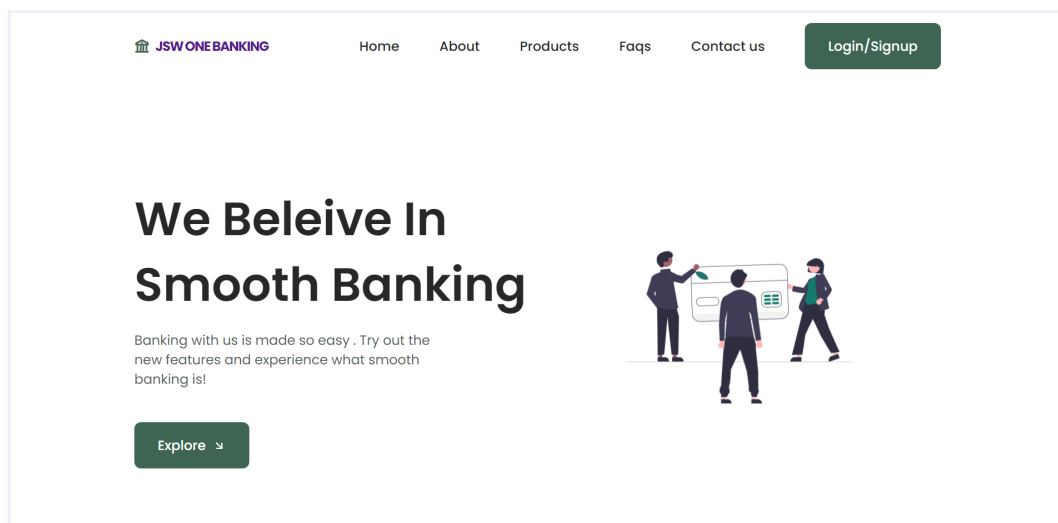
Software Required:

The project is implemented in Core Java as it provides the implementation of Socket and Server Socket classes that are used to connect distinct applications, hence the software's required in the creation and execution of the project are j2sdk1.7 or Eclipse .As we know JAVA is a platform independent language so this software runs with JRE environment on any desired platform i.e. Linux ,windows 9x, XP, or 2000 or any operating system.

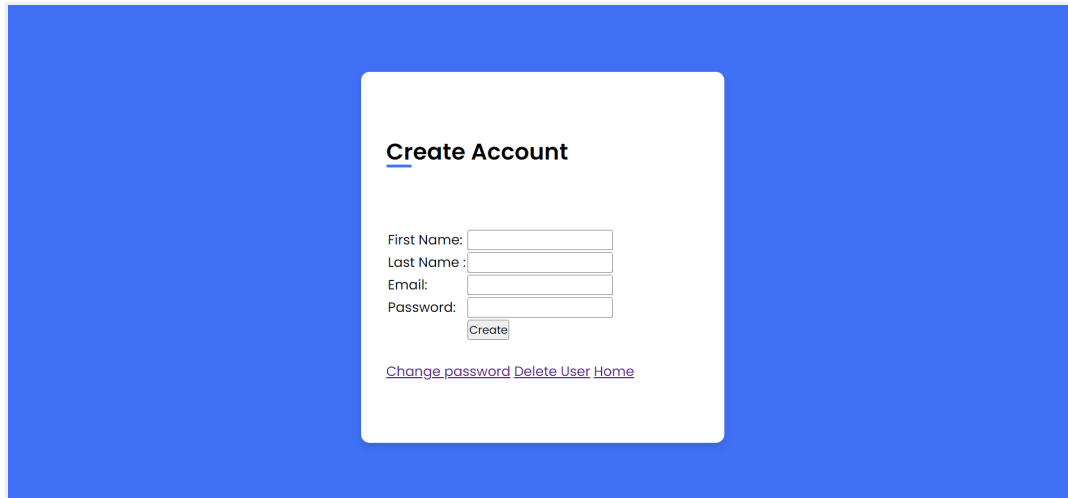
Hardware Required:

As the project does not involve any database, its hardware requirements are minimal. Any System with Pentium P2 or above processor, 32MB RAM, 1GB Hard Disk, a LAN Card, and a CDRom is sufficient. Its network based software so computers connected with any kind of mode (wireless, LAN connected etc) will suit its requirements. . . . It can also be run on a single machine for its demo use.

Snapshots of the Project



Home Page

A screenshot of a web page with a solid blue background. In the center is a white rounded rectangle containing a form titled "Create Account". The form has four input fields: "First Name:", "Last Name:", "Email:", and "Password:". Below the "Password:" field is a "Create" button. At the bottom of the form are three links: "Change password", "Delete User", and "Home".

Create Account

First Name:

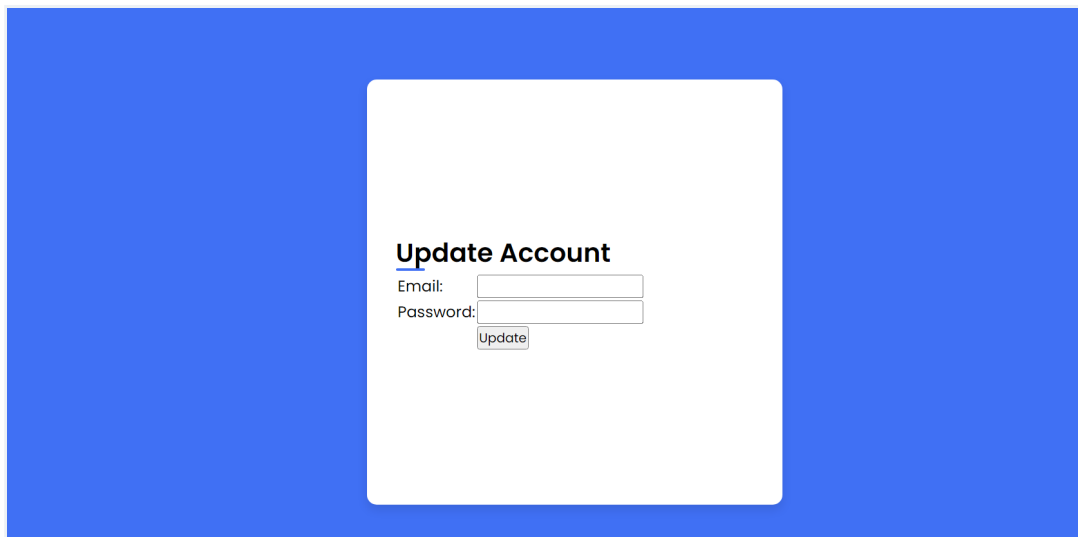
Last Name:

Email:

Password:

[Change password](#) [Delete User](#) [Home](#)

Sign up Page

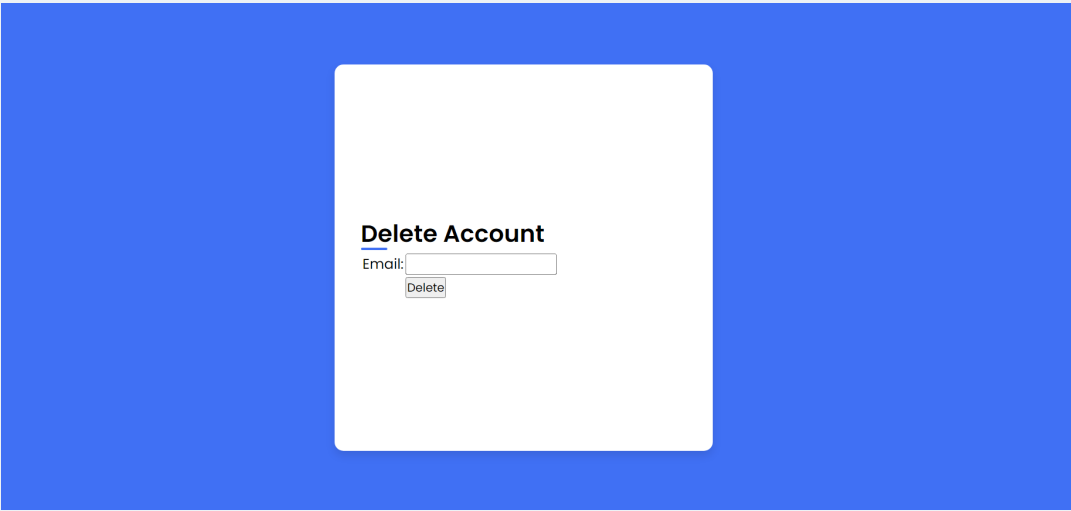
A screenshot of a web page with a solid blue background. In the center is a white rounded rectangle containing a form titled "Update Account". The form has two input fields: "Email:" and "Password:". Below the "Password:" field is an "Update" button.

Update Account

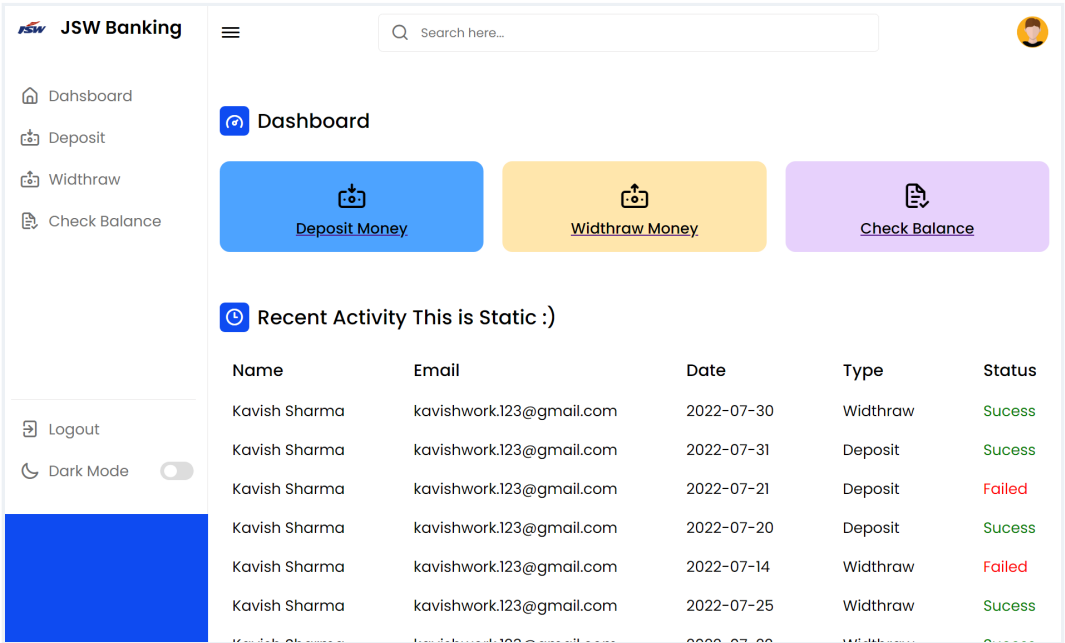
Email:

Password:

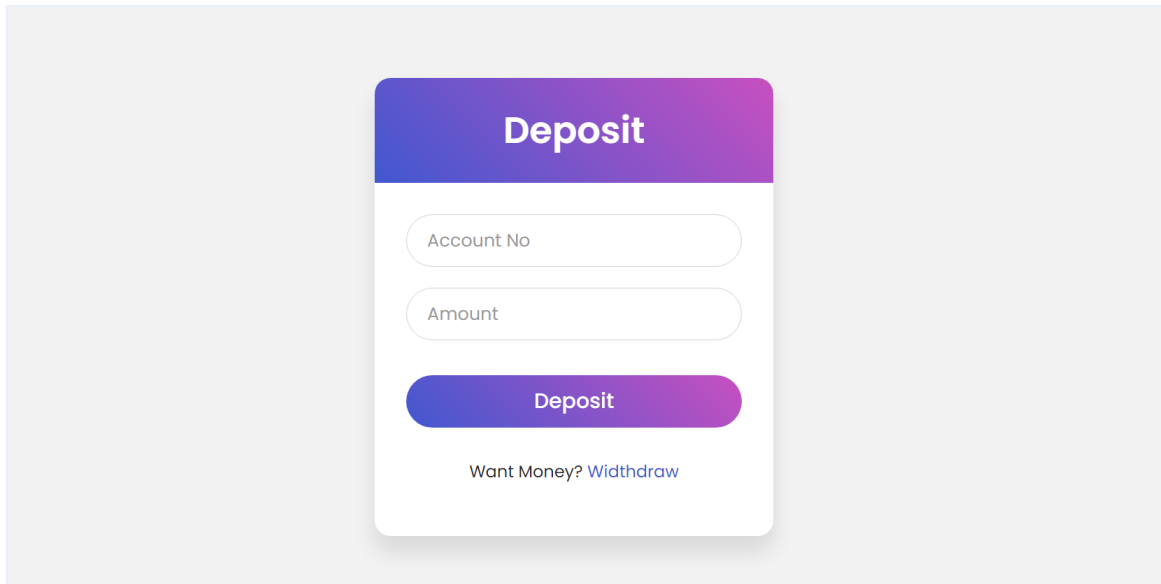
Change Password Page



Delete Account Page

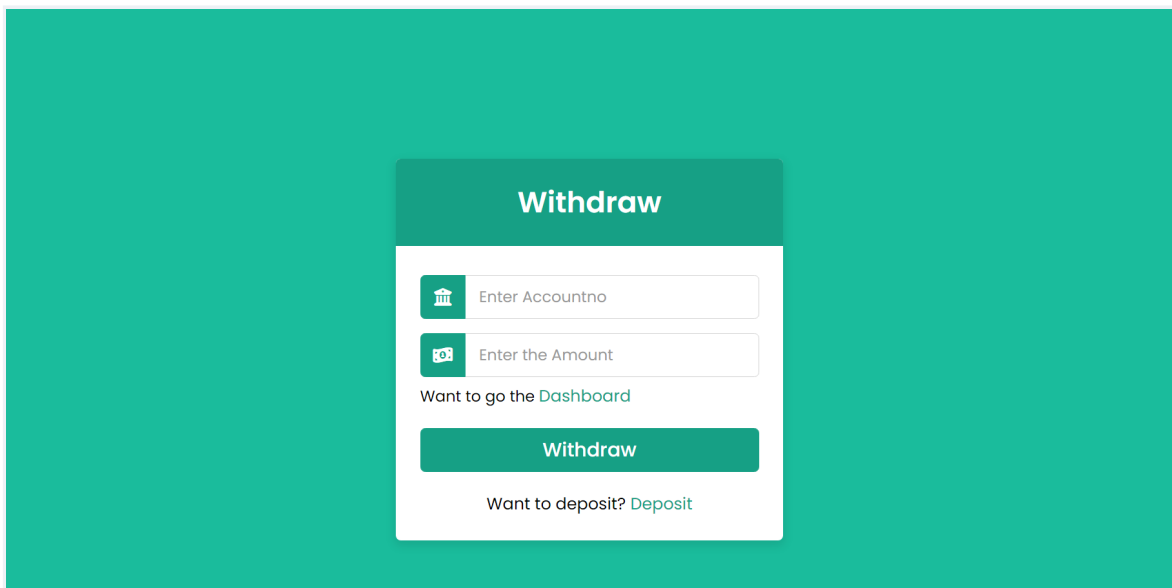


Dashboard Page



The image shows a mobile app interface for a deposit transaction. It features a purple header with the word "Deposit" in white. Below the header are two white input fields with rounded corners, labeled "Account No" and "Amount". Underneath these fields is a large, rounded purple button with the word "Deposit" in white. At the bottom of the form, there is a link that says "Want Money? [Withdraw](#)". The entire form is centered on a light gray background.

Deposit Page



The image shows a mobile app interface for a withdrawal transaction. It features a teal header with the word "Withdraw" in white. Below the header are two white input fields with rounded corners, each preceded by a teal icon: a bank building for "Enter Accountno" and a cash register for "Enter the Amount". Below these fields is a link that says "Want to go the [Dashboard](#)". Underneath is a large, rounded teal button with the word "Withdraw" in white. At the bottom of the form, there is a link that says "Want to deposit? [Deposit](#)". The entire form is centered on a teal background.

Withdraw Page

SCHEMAS

Filter objects

- mydb
 - Tables
 - account
 - Columns
 - Indexes
 - Foreign Keys
 - Triggers
 - user
 - Columns
 - Indexes
 - Foreign Keys
 - Triggers
 - Views
 - Stored Procedures
 - Functions
- mydb1
 - registration

Administration Schemas

Information

Table: user

```

5 • INSERT INTO user (FIRSTNAME, LASTNAME, EMAIL, password)
6   VALUES ("a", "b", "a@a.com", "test");
7
8 • use mydb;
9 • create table dashboard(accountno varchar(33), amount varchar(33))
10 select * from dashboard;
11 • INSERT INTO dashboard (accountno, amount)
12   VALUES ("323", "776");

```

Result Grid

firstName	lastName	email	password
ekdn	dem ded	dedm d	tst
kavish	verma	kavsi@gmail.com	12222
kavish	kabvisj	kabbh	1222
kavish11	kabvisj	kabbh11	111
dededf	dffdfdf	dfdf	dfdf
Abhisek	Sharma	abhisek@gmail.com	122
kavish	dwdw	dww	wddw

Form Editor

Database

Project Explorer

- method
- Methods
- methods1
- Online_Banking
- Registration form
- S03L02 -Hello World
- ServeletBasics
- Servelets1
- Servers
- udemyclass
- udemylearnings
- UserApp
 - Deployment Descriptor: UserApp
 - JAX-WS Web Services
 - Service Endpoint Interfaces
 - Web Services
 - Java Resources
 - src/main/java
 - com.user
 - CreateUserServlets.java
 - DeleteUserServlets.java
 - depoServlet.java
 - ReadUserServlet.java
 - UpdateUserServlets.java
 - withServlet.java
 - Libraries
 - Referenced Libraries
 - build
 - src
 - main
 - java

withServlet.java

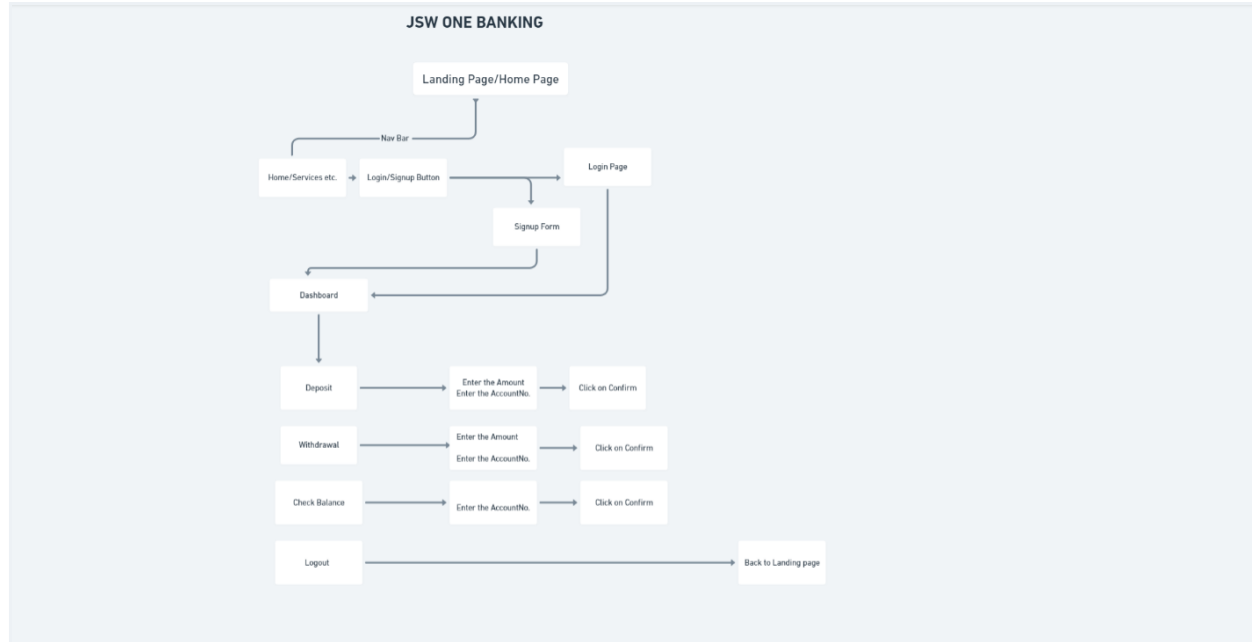
```

19 @WebServlet("/depoServlet")
20 public class depoServlet extends HttpServlet {
21     private static final long serialVersionUID = 1L;
22
23     private Connection connection;
24
25     public void init(ServletConfig config) {
26         try {
27             Class.forName("com.mysql.cj.jdbc.Driver");
28             connection=DriverManager.getConnection("jdbc:mysql://localhost:3306
29         } catch (SQLException e) {
30             e.printStackTrace();
31         } catch (ClassNotFoundException e) {
32             e.printStackTrace();
33         }
34     }
35     /**
36      * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse res
37      */
38     protected void doPost(HttpServletRequest request, HttpServletResponse response
39         String Accountno = request.getParameter("Accountno");
40         String Amount = request.getParameter("Amount");

```

need either to explicitly disable SSL by setting useSSL=false, or set useSSL=true and provide truststore for server certificate verification.

Code



Flow Diagram

Conclusion

The Jsw One Banking Project was basically a CRUD operation Project which has some of the basic functionalities and would love to add more of the facilities like opening a new account transfer of Bitcoins , Encryption & Decryption of the Passwords etc. to make it more function-able .Last but not the least this project has taught me about the concepts of java, servlets and the frontend part which will be surely helpful in my future.