


## Registration Form

Name	<input type="text"/>
Email	<input type="text"/>
Mobile	<input type="text"/>
DOB	<input type="text" value="dd-mm-yyyy"/> 
City	<input type="text" value="Pune"/> ▼
Gender	<input type="radio"/> Female <input type="radio"/> Male
<input type="button" value="Register"/>	
<input type="button" value="Show Users"/>	



# USER MANAGEMENT SYSTEM

GAYATHRI RAMADURGUM  
OCTOBER 1<sup>ST</sup>, 2023

---





# INTRODUCTION

---

This is a Java Dynamic Web Project - a User Management System which is built using the concepts of Servlets and JDBC with MySQL.

The home page serves as the point of entry for users to engage with the system. It consists of a user registration form which asks details such as name, email, mobile, dob, city, gender from the user. To initiate the registration process, users simply need to click the "Register" button after entering the required information. The registration process ensures that user data is collected and stored efficiently in a MySQL database.

By clicking on the "Show Users" button, users can see the table of previously registered users. All fields will be displayed to them, making it easy to manage and monitor the user base.

Additionally, this project allows for the deletion of registered users and the updation of user information whenever it is required.



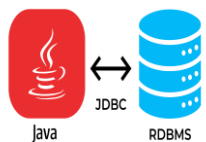
# TECHNOLOGIES USED

## 1. Servlets



- Servlets are a fundamental part of Java EE (Enterprise Edition) for building web applications. They are server-side Java components that handle HTTP requests and generate dynamic responses. Servlets are responsible for processing user input from the registration form. They capture user data, validate it, and perform necessary actions such as storing data in the database or retrieving user information.

## 2. JDBC



- JDBC (Java Database Connectivity) is a Java API that allows Java applications to interact with relational databases like MySQL. JDBC is used to establish a connection between your Java application (Servlets) and the MySQL database. It provides the necessary libraries and methods to connect, query, and manipulate the database.

## 3. Tomcat Server



- Apache Tomcat is a widely used open-source application server that implements the Java Servlet in this project. Apache Tomcat is responsible for hosting and running your Java web application. It handles the execution of Servlets when HTTP requests are made to your web application.



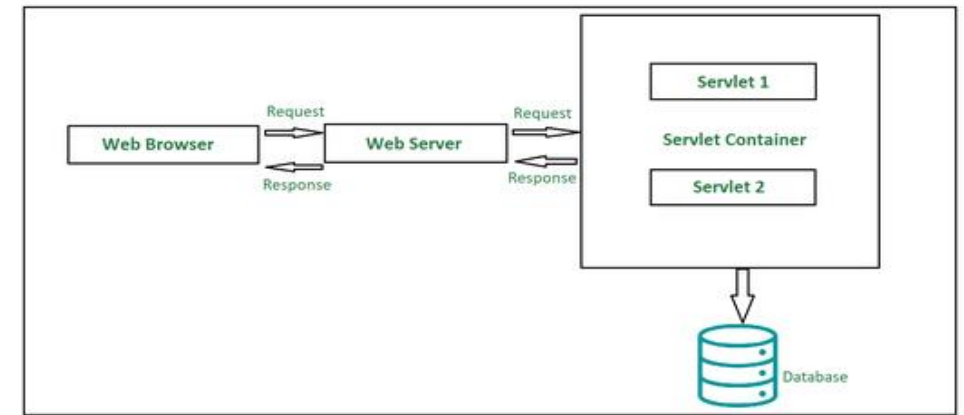
# WORKING OF THE PROJECT

## 1. Initial Page Load - index.html:

When a user accesses the system, the initial request is sent to retrieve the index.html page.

index.html contains a registration form where users can input their personal information, such as name, email, mobile number, date of birth, city, and gender.

After filling out the form, users can click the "Register" button to submit their registration details.



## 2. User Registration - RegisterServlet:

When the "Register" button is clicked, the RegisterServlet is invoked.

RegisterServlet processes the user's input:

It retrieves the data submitted in the registration form, including name, email, mobile number, date of birth, city, and gender.

This servlet's purpose is to store the user's registration data in a MySQL database using JDBC.

## 3. Displaying Registered Users - ShowDataServlet:

After successful registration or at any time, the user can click the "Show Users" button in index.html.

This action triggers an HTTP request to the ShowDataServlet.



# CONTINUED



Inside the ShowDataServlet:

The servlet executes a query using JDBC to retrieve a list of all previously registered users from the MySQL database.

The retrieved user data is formatted into an HTML table dynamically.

The HTML table is then sent as a response to the user's browser for display.

## 4. User Editing - EditServlet:

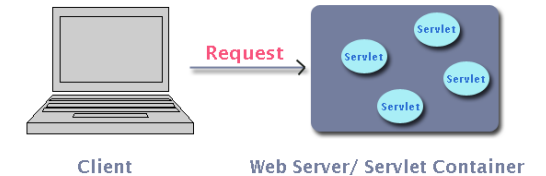
On the "Show Users" page, users can click on the "Edit" link associated with a specific user.

Clicking "Edit" invokes the EditServlet.

Inside the EditServlet:

The servlet retrieves the user's current information from the database based on the user's ID. Users can make changes to their data, such as updating their email, password, contact information, or other profile attributes.

The servlet securely updates the user's information in the MySQL database using JDBC.



Client making a request to Web Server

## 5. User Deletion - DeleteServlet:

On the "Show Users" page, users can click on the "Delete" link associated with a specific user.

Clicking "Delete" invokes the DeleteServlet.

Inside the DeleteServlet:

The servlet deletes the user's account.



# PROJECT FEATURES



C R U D

## User Registration:

**User Onboarding:** New users can easily register their information by providing their personal information, including name, email, mobile number, date of birth, city, and gender.

**Efficiency:** The registration process is streamlined, reducing the time and effort required for users to join the system.

## User Listing:

**User Monitoring:** People can view the list of users that were previously registered

**Data Visibility:** User details are presented in a user-friendly and organized table, making it easy to monitor and manage the user base.

## User Editing:

**Profile Management:** user information can be updated, including email, city, contact information, etc.

**Flexibility:** Users have the ability to maintain their profiles and keep their information up-to-date.



# CONTINUED



## User Deletion:

**Account Management:** It is possible to delete user accounts as needed, ensuring efficient user account management.

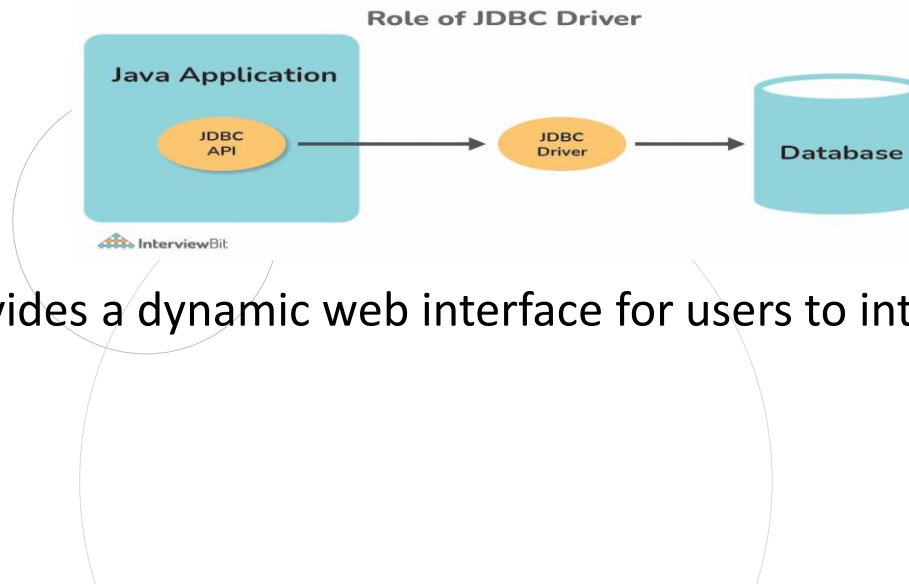
## Data Storage and Retrieval:

**Efficient Database Interaction:** JDBC with MySQL allows for efficient storage and retrieval of user data.

**Data Accessibility:** A centralized database stores user records, making it easy to manage and query user information.

## Dynamic Web Application:

**User Interaction:** The project provides a dynamic web interface for users to interact with, enhancing user engagement and experience.





# FUTURE SCOPE

---



The User Management System project can be expanded and enhanced in several ways to meet evolving user needs and technological advancements. Here are some potential features to consider:

- **Email Verification:** Enhance user validation by adding email verification. Users receive a verification link or code via email to confirm their registration.
- **Profile Pictures:** Allow users to upload profile pictures or avatars to personalize their profiles.
- **User Messaging and Notifications:** Integrate messaging and notification features for users to communicate within the platform and receive updates.
- **Multi-Language Support:** Offer support for multiple languages to cater to a global user base.