CAREER CONNECTION PLATFORM USING STUDENT PROFILING

A Project Report

Submitted for the partial fulfillment for the award of degree of

BACHELOR OF COMPUTER APPLICATIONS

By

DEEPAK E

BU211010

Under the Guidance of

Mr. R .KARTHIK, M.Sc., M.Phil.,B.Ed.

DEPARTMENT OF COMPUTER APPLICATIONS (UG)

SACRED HEART COLLEGE (AUTONOMOUS)

Tirupattur, Tirupattur Dt- 635 601

APRIL – 2024

CERTIFICATE

CERTIFICATE

This is to certify that the report entitled

Being submitted to the Thiruvalluvar University,Vellore

Career Connection Platform Using Student Profiling

By

DEEPAK E

BU211010

For the partial fulfillment for the award of degree of

BACHELOR OF COMPUTER APPLICATIONS

Is a bona-fide record of work carried out by him under my guidance and Supervision.

Signature of the Guide Signature of the Head

Date:

Submitted for the viva-voce examination held on_____

Examiners:

1.

2.

ACKNOWLEDGMENT

ACKNOWLEDGEMENT

At the very outset I offer my sincere thanks to **Almighty God** for the grace and blessings that make me to complete the project in complete the project in successful manner. I sincerely thank my parents who have gifted me this life to attain many achievements.

As I submit the project report, it gives me great pleasure to acknowledge my gratitude to various people who were instrumental in the completion of this project.

I offer my humble gratitude to Rev. **Dr. P. Praveen Peter SDB,** the Rector,Rev. **Dr. D. Maria Antony Raj SDB**, The Principal and **Rev. Dr. K. A. Maria Arokiaraj SDB**, The Additional Principal, Sacred Heart College(Autonomous), Tirupattur for permitting me to do my project work.

I wish to convey my deep sense of gratitude to **Dr. A. JOHN MARTIN.** The Head, Department of Computer Applications Shift-II, Sacred Heart College, Tirupattur.

I express my deep sense of gratitude to my project guide **Mr. R. Karthik, M.Sc., M.Phil., B.Ed.** who is the source of inspiration that made me to pursue this work successfully. I thank him for his valuable guidance, encouragement and support in doing this project.

I would like to thank the entire teaching and non-teaching staff of the Department of Computer Applications for helping me in completing the project successfully. Finally, I thank each and every one of my friends, who assisted me in completing my project work.

DEEPAK E

TABLE OF CONTENT

CHAPTER 1
INTRODUCTION

INTRODUCTION

1.1 Abstract

This paper introduces a cutting-edge Career Connection Platform (CCP) designed to revolutionize the way students navigate their transition from education to employment through a machine learning-based recommendation system. Central to our approach is the utilization of student profiling, where detailed information on students' academic achievements, skills, extracurricular activities, and personal career interests are analyzed using advanced machine learning techniques. The core of the CCP is a sophisticated recommendation algorithm that processes this multifaceted data to match students with personalized career opportunities, internships, mentorship programs, and further educational pursuits that align with their profiles and aspirations. By leveraging the power of machine learning, our platform not only provides tailored recommendations but also dynamically adapts to the changing trends in the job market and the evolving interests and profiles of students. This ensures a high degree of relevance and timeliness in the opportunities presented to each user. Furthermore, the platform facilitates direct engagement between students and potential employers, creating a streamlined pathway for professional networking, job applications, and career development. This paper details the machine learning models and algorithms employed in the development of the CCP, including data preprocessing, feature selection, and the evaluation of various recommendation system architectures. We also discuss the challenges faced in balancing algorithmic accuracy with user privacy and data security, and outline the strategies implemented to address these concerns. The CCP represents a significant advancement in career guidance technology, offering a personalized, data-driven approach to career exploration and planning. By harnessing the predictive power of machine learning within the context of student career development, the platform aims to enhance the employability of students and closely align their educational experiences with the needs of the modern workforce.

1.2 Existing System

- Career Counseling and Advisory Services: Traditional career counseling involves face-to-face consultations with career advisors who provide guidance based on the student's interests, academic performance, and potential career paths. While personalized, this method is heavily reliant on the counselor's expertise and lacks the scalability and data-driven precision that ML algorithms can offer.
- Educational and Career Assessment Tools:Various tools exist to help students assess their skills, personality, and interests, usually through questionnaires and tests. Results are then matched with potential career paths. These tools provide a broad overview but may not account for real-time labor market trends or offer dynamic, personalized career progression pathways.

1.3 Problem Statement

- Generic Recommendations: Many current platforms offer broad, one-size-fits-all suggestions that fail to consider the individual's specific academic background, skill set, personal interests, and career goals. This lack of personalization can lead to mismatches between students and job opportunities, resulting in lower job satisfaction and higher turnover rates.
- Recommendations: There is no recommendation was built for career suggestion using the student profile.

1.4 Proposed System

- The proposed Career Connection Platform (CCP) integrates advanced machine learning algorithms to analyze student data and offer personalized career recommendations, while fostering interactive engagement with employers and industry professionals to enhance students' employability and align their educational journey with market demands.

1.5 Software and hardware Requirements

Software Requirements

- Eclipse IDE (Integrated Development Environment) is a widely used open-source IDE for Java development, although it supports various other programming languages as well.

Hardware Requirements

- System AMD Ryzen 3 Processor
- CPU ryzen 3
- Clock Speed 2.4 MHz
- Main Memory (RAM) 8GB DDR 4
- Secondary Memory 1 TB SSD
- Operating System MS-Windows 11
- Environment Windows 7,8,9,10

1.6 List of Modules

This the Career Connection Platform Using Student Profiling contains five main modules they are,

- Home.
- Profile.
- Feedback.
- Resume.
    - Upload Document.
    - Upload Text Of The Document.
- Administration
    - Overall View.
    - User Details.
    - User Feedback.
    - Plants Details.

Home

- Displays the content of the website based on video and gif.

Profile

- It is used to display the user details based on the user sign up.

Feedback

- User can send message of an issues or service rating to the admin.

Resume

Upload Document

- It gets the document from the user/admin for the further access.

Upload Text Of The Document

- It gets the text of a document by the user/admin for the further access.

Administration

- This module is used for admin to view the overall view, admin can add, update and delete the user.
- Admin can also view all the user sent feedback and delete it.
- Admin can also view all plants present in the dataset and add new plants to the dataset.

1.7 Summary

The chapter describes Project Proposal Document (Abstract, Existing System, Problem Statement, Proposed System, Software and hardware Requirements, List of Modules) and in next chapter describe Feasibility study.

CHAPTER 2
FEASIBILITY STUDY

FEASIBILITY STUDY

2.1 Introduction

In the current landscape of education and employment, there exists a significant gap between students' academic experiences and their career aspirations. A Career Connection Platform utilizing student profiling aims to bridge this gap by providing personalized career guidance and opportunities tailored to individual strengths, interests, and goals. This feasibility study examines the viability and potential challenges of implementing such a platform.

2.2 Feasibility Analysis

An important outcome of the preliminary investigation is the affirmation that system requested is feasible. The three types of feasibility studies which helped me to identity the solutions are stated below:

- Technical Feasibility
- Economical Feasibility
- Operational Feasibility

Technical Feasibility

- Data Collection and Analysis: Utilize advanced data collection techniques and scalable algorithms to gather and analyze student data effectively while ensuring security and compliance with data protection regulations.
- User Interface and Experience: Design a user-friendly interface with personalized features, ensuring cross-platform compatibility and seamless integration with existing educational and career platforms.

Economic Feasibility

- Initial Investment: Estimate development costs, allocate resources, and procure necessary equipment and software licenses for the platform's setup and deployment.
- Revenue Streams: Implement subscription models, advertisement revenue, and strategic partnerships to generate sustainable income streams and achieve profitability.

Operational Feasibility

- User Adoption: Provide comprehensive training and support, gather user feedback, and manage change effectively to ensure smooth user adoption and satisfaction.
- Maintenance and Compliance: Establish protocols for regular maintenance, updates, and scalability planning while ensuring full compliance with data protection regulations and legal requirements.

2.3 Vision Document

Problem Statement

The problem of	The student after the college completion they are in confuse state.
Affects	Students
The impact of which	In this process, they get a idea about their career.
A successful solution would be	Students get clear or appropriate idea about they a career.

Problem Positioning Statement

Name	Represent	Roles
User	Student/Employee	User helped to know about the application they are getting into.
Admin	To manage the Upload profile and text of the profile.	Making ensure the everything is fine or Issues.

2.4 Summary

This chapter concludes with the summary and objectives of the project. The next chapter describes the requirements analysis of the project.

CHAPTER 3

SYSTEM REQUIREMENTS SPECIFICATION (SRS)

SYSTEM REQUIREMENTS SPECIFICATION

3.1 Introduction

This chapter describes the modules that are implemented in the software with UML diagrams and use case designs.

3.2 Modules Description

It is an ML based website, which is used to show the Career through Resume/Profile.

Modules:

This the Enhancing Student Mark Prediction contains six main modules they are,

- Home.
- Profile
- Feedback.
- Resume.
    - Upload Document.
    - Upload Text Of The Document.
- Administration.
    - Overall View.
    - User Details.
    - User Feedback.
    - Uploaded Profiles
    - Recommended Profile.

3.3 Module Specification

Home

- **Purpose:**

  It is used to display and visit the home page.
- **Responsible person:**

  Users and Admin.
- **Entry Criteria:**

  By entering to the site through login.
- **Input:**

  Display the content of the website through video and gif.
- **Process:**

  Show the content based on the login.

- **Output:**

  Content of the site will display.

- **Exit Criteria:**

  You can exit by clicking the Logout button.

Profile

- **Purpose**

  It is used to display and visit the login details.

- **Responsible person**

  Users and Admin.

- **Entry Criteria**

  By clicking the Profile Module.

- **Input**

  Login/Sign up details are given by the user/admin.

- **Process**

  Get the values by Sign Up page.

- **Output**

  Sign Up Details will display.

- **Exit Criteria**

  You can exit by clicking the Back button.

Feedback

- **Purpose**

  User can send message of an issues or service rating to the admin.

- **Responsible person**

  Users and Admin.

- **Entry Criteria**

  By clicking the Feedback Module.

- **Input**

  Various question answered by the user.

- **Process**

  Get the details from the user and store in the user.

- **Output**

  Successfully sent message will display.

- **Exit Criteria**

  You can exit by clicking the back button.

Resume

Upload Document

- **Purpose:**

  User/Admin can upload document to get suggestion.

- **Responsible person:**

  User, Admin.

- **Entry Criteria:**

  By clicking the Upload Document button.

- **Input:**

  Upload the document for the next process.

- **Process**

  Give the suggestion based on the document uploaded.

- **Output:**

  Based on the document can get the suggestion.

- **Exit Criteria:**

  You can exit by clicking back button.

Upload Text Of The Document

- **Purpose:**

  User/Admin can upload text to get suggestion.

- **Responsible person:**

  User, Admin.

- **Entry Criteria:**

  By clicking the Upload Text Of The Document button.

- **Input:**

  Upload the document text for the next process.

- **Process**

  Give the suggestion based on the uploaded document text.

- **Output:**

  Based on the document text can get the suggestion.

- **Exit Criteria:**

You can exit by clicking back button.

Administration:

Dashboard

- **Purpose:**

  Admin can view the overall of total users, total Recommended Profiles, total Profile uploaded available in dataset and total feedback received.

- **Responsible person:**

  Admin.

- **Entry Criteria:**

  By clicking the Dashboard Module.

- **Input:**

  Total user, recommended profile, total profile uploaded and feedback from the database.

- **Process:**

  Retrieve data's from database.

- **Output:**

  Display the total users, recommended profile, total profile uploaded and total feedback.

- **Exit Criteria:**

  You can exit by clicking the back button.

User

- **Purpose:**

  Admin can view all user's details and Admin can add new user, delete user and update user details.

- **Responsible person:**

  Admin.

- **Entry Criteria:**

  By clicking the user Module.

- **Input:**

  User's details from the database.

- **Process:**

  Retrieve user's details from database, add new user, update the user details and delete the user.

- **Output:**

  Display the all the users details.

- **Exit Criteria:**

  You can exit by clicking the back button.

Feedback:

- **Purpose:**

  Admin can view and delete all received feedback.

- **Responsible person:**

  Admin.

- **Entry Criteria:**

  By clicking the feedback Module.

- **Input:**

  Feedback details from the database.

- **Process:**

  Retrieve user's details from database and delete the feedback.

- **Output:**

  Display the all the received feedback.

- **Exit Criteria:**

  You can exit by clicking the back button.

Uploaded Profiles:

- **Purpose:**

  User/Admin view the uploaded profile.

- **Responsible person:**

  User, Admin.

- **Entry Criteria:**

  By clicking the Upload Profile button.

- **Input:**

  The admin can add and delete profile.

- **Process:**

  The uploaded profiles will be shown uploaded by user/admin.

- **Output:**

  Based on the profile uploaded shows the total profiles.

- **Exit Criteria:**

  You can exit by clicking back button.

Recommended Profile:

- **Purpose:**

  User/Admin view the Suggested profile.

- **Responsible person:**

  User, Admin.

- **Entry Criteria:**

  By clicking the Recommended Profile button.

- **Input:**

  The admin can add and delete.

- **Process:**

  The uploaded profiles will be shown which is uploaded by user/admin.

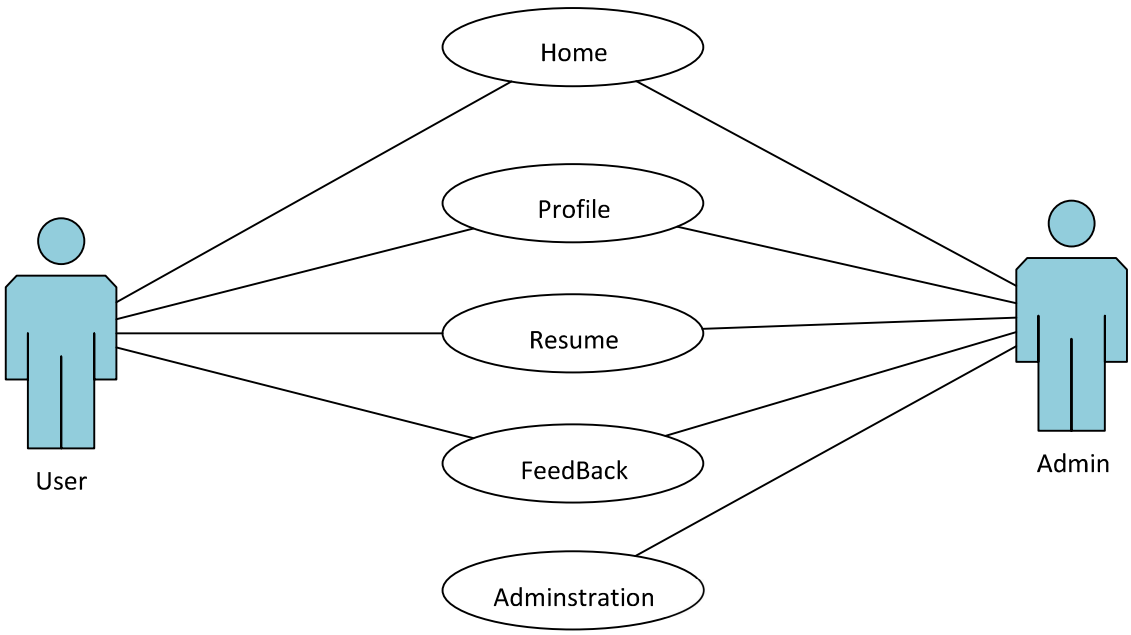
- **Output:**

  Based on the profile uploaded shows the total profiles.

- **Exit Criteria:**

  You can exit by clicking back button.

3.4 UML Use Case



USE CASE	DESCRIPTION
Home	The user and admin can view the content of the website.
Profile	The user and admin can view the sign up details.
Feedback	The user can send any issues or suggestions to the admin.
Resume	The user and the admin can upload the text and upload the document for the career suggestion from the two options.
Administration	The admin can view overall of users, predicted marks and feedbacks.

3.5 Summary

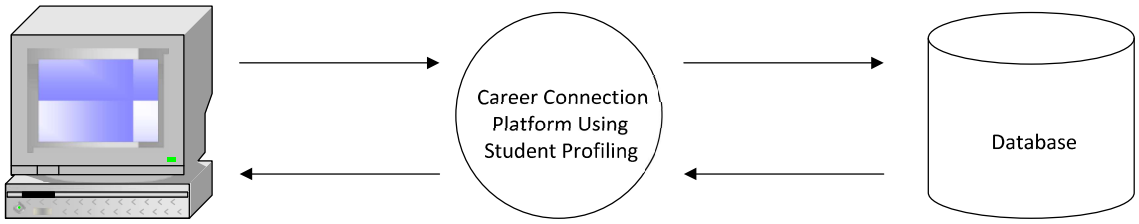
This session is described about System Analysis and design introduces the module description, and use case diagram

CHAPTER 4

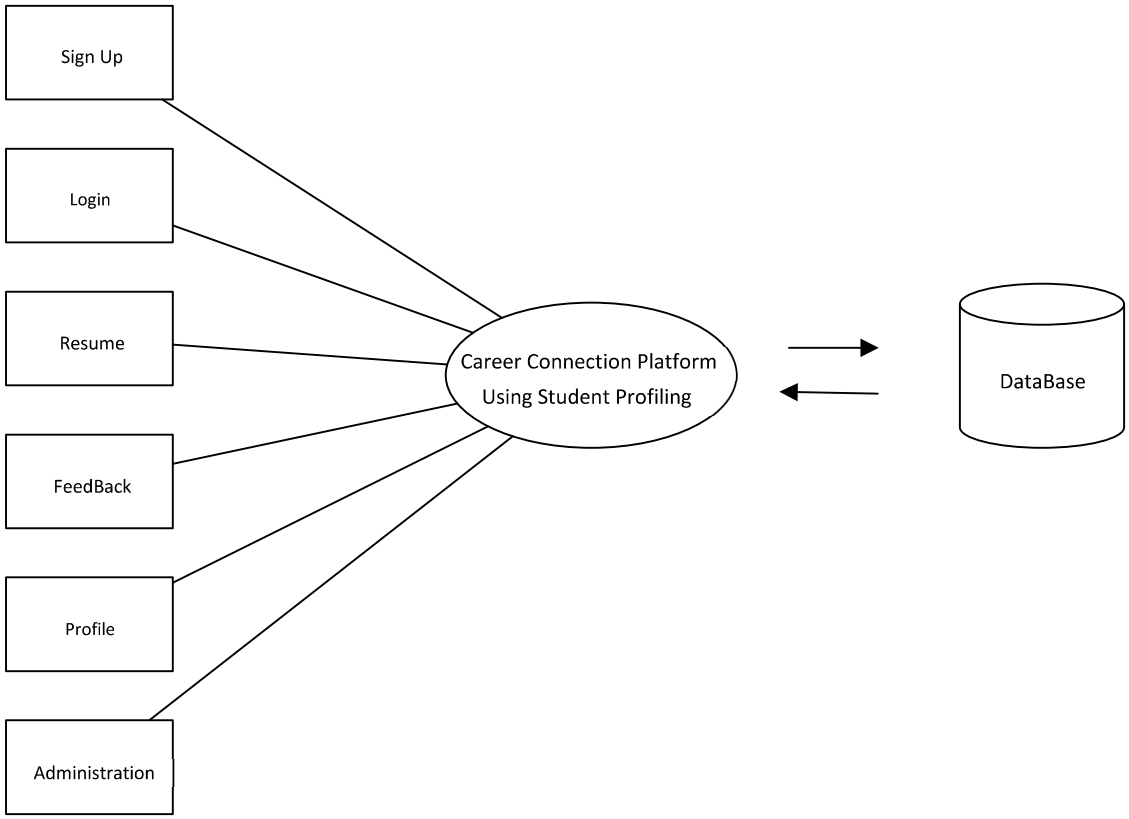
SOFTWARE ANALYSIS AND DESIGN

4.1 Data Flow Diagram

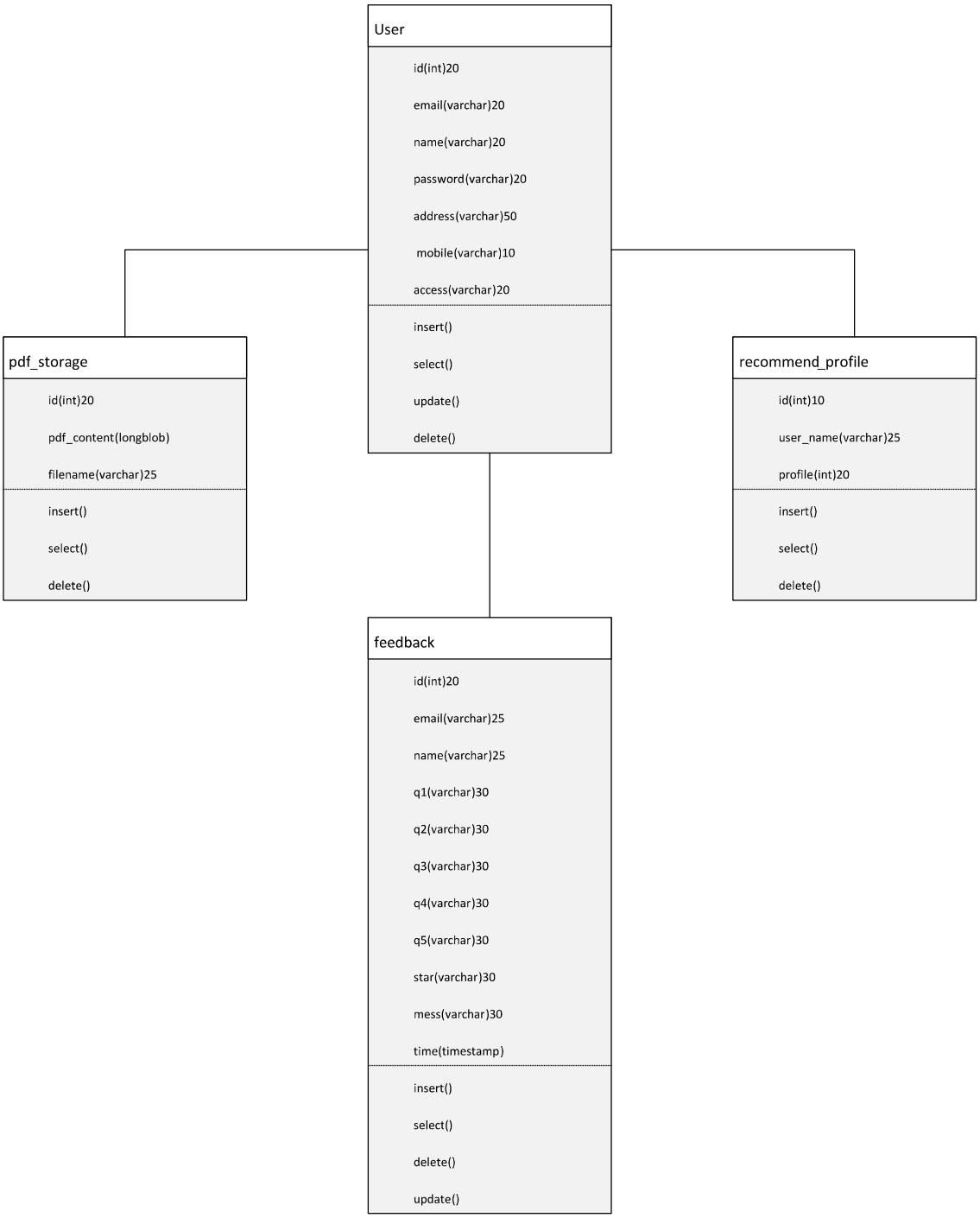
Level 0



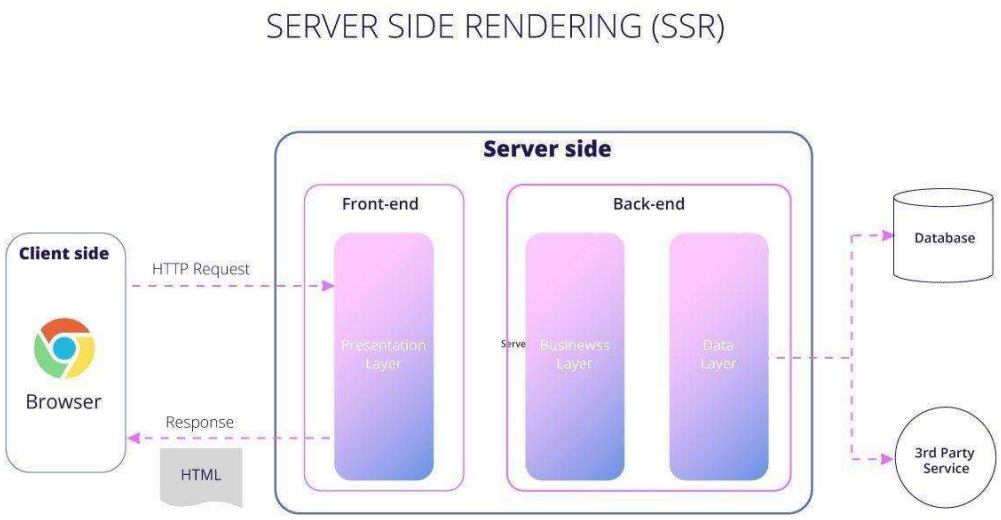
Level 1



4.2 Class Diagram



4.3 Architecture Diagram



4.4 Table Design

User

Column Name	Data type	Length	Key
id	int	11	Primary key /Auto Increment
email	Varchar	20	Primary
name	Varchar	20	Not Null
password	Varchar	12	Not Null
address	Varchar	50	Not Null
modile	Varchar	20	Not Null
access	varchar	20	Not Null

Recommend_profile:

Column Name	Data type	Length	Key
id	int	20	Primary key/Auto Increment
User_name	Varchar	25	Not Null
profile	int	20	Not Null

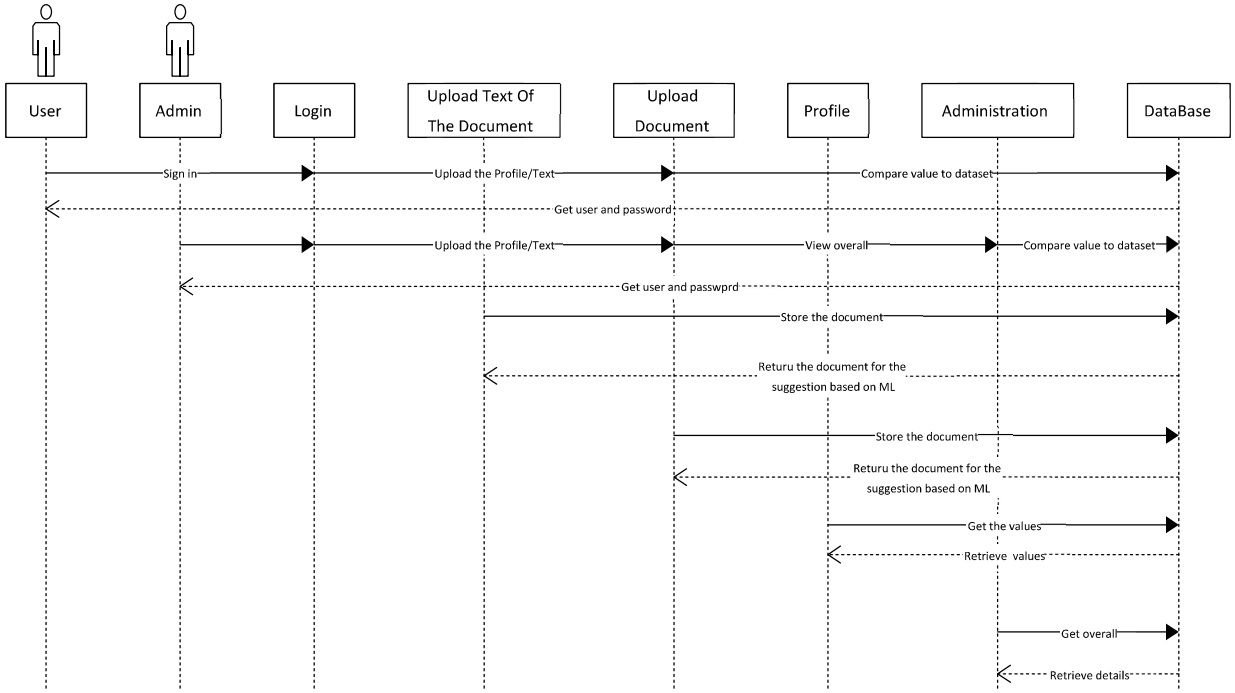
Pdf_storage

Column Name	Data type	Length	Key
id	int	20	Primary key/Auto Increment
Pdf_content	longblob	-	Not Null
filename	varchar	25	Not Null

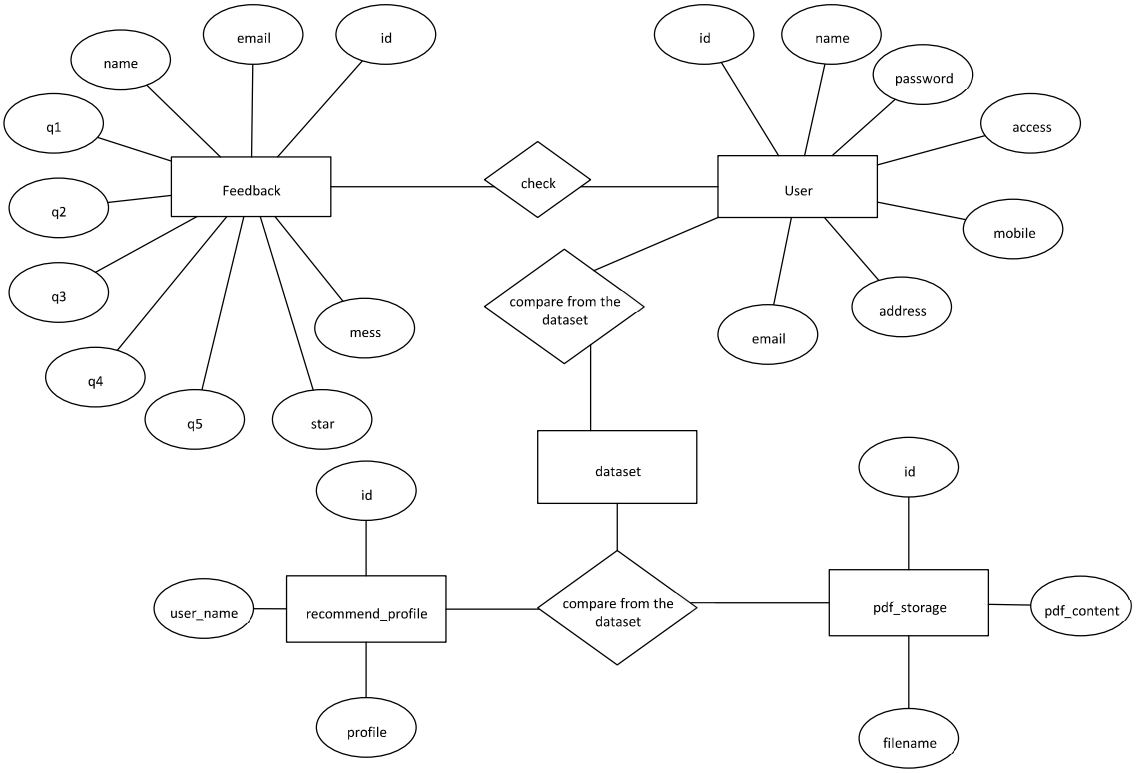
Feedback

Column Name	Data type	Length	Key
id	int	20	Primary key/Auto Increment
email	varchar	25	Primary key
name	varchar	20	Primary key
q1	varchar	30	Not Null
q2	varchar	30	Not Null
q3	varchar	30	Not Null
q4	varchar	30	Not Null
q5	varchar	30	Not Null
star	varchar	30	Not Null
mess	varchar	30	Not Null
time	timestamp	-	Not Null

4.5 Sequence Diagram



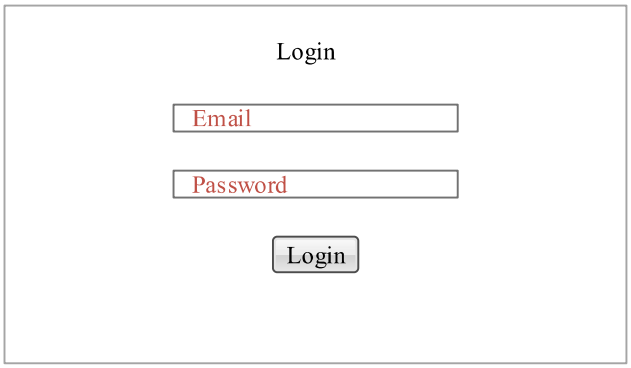
4.6 E-R Diagram



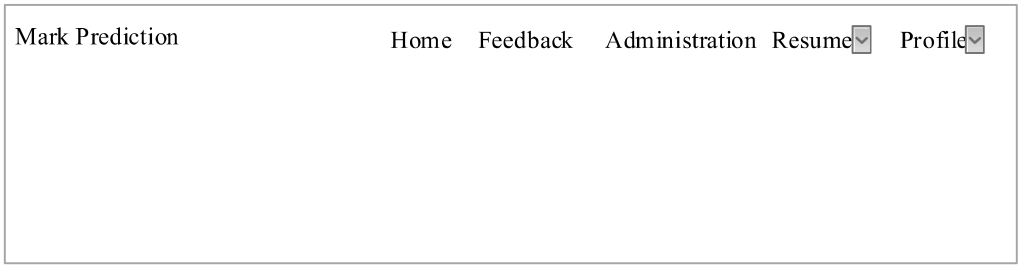
4.7 UI Design

Sign Up

Login

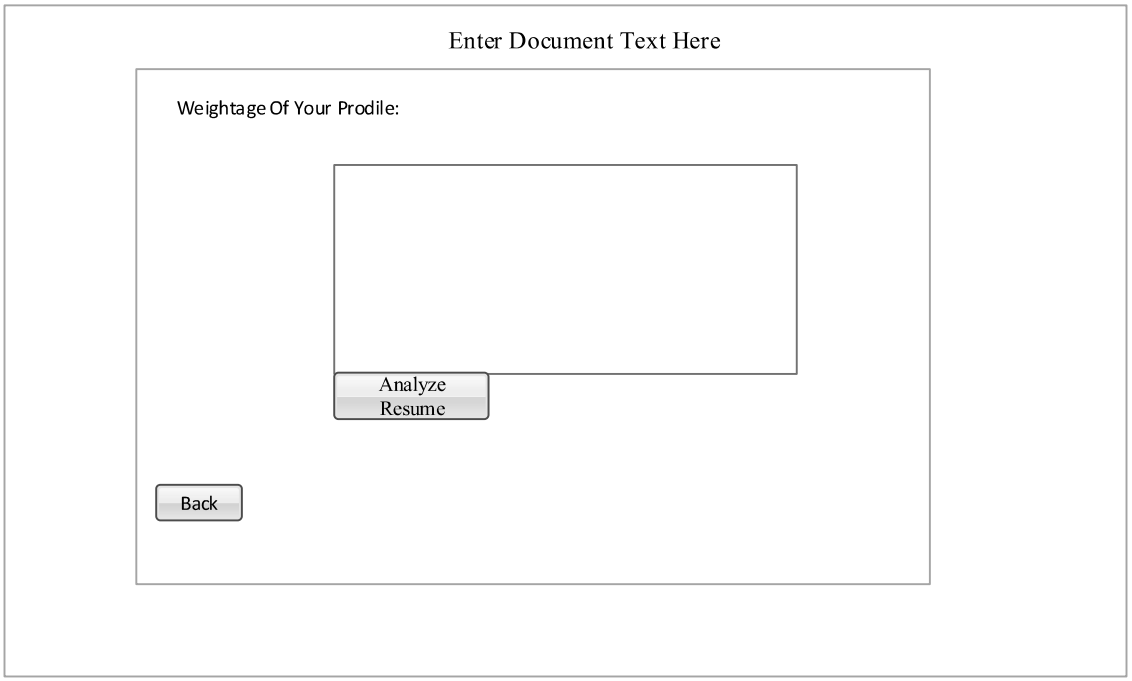


Home Page

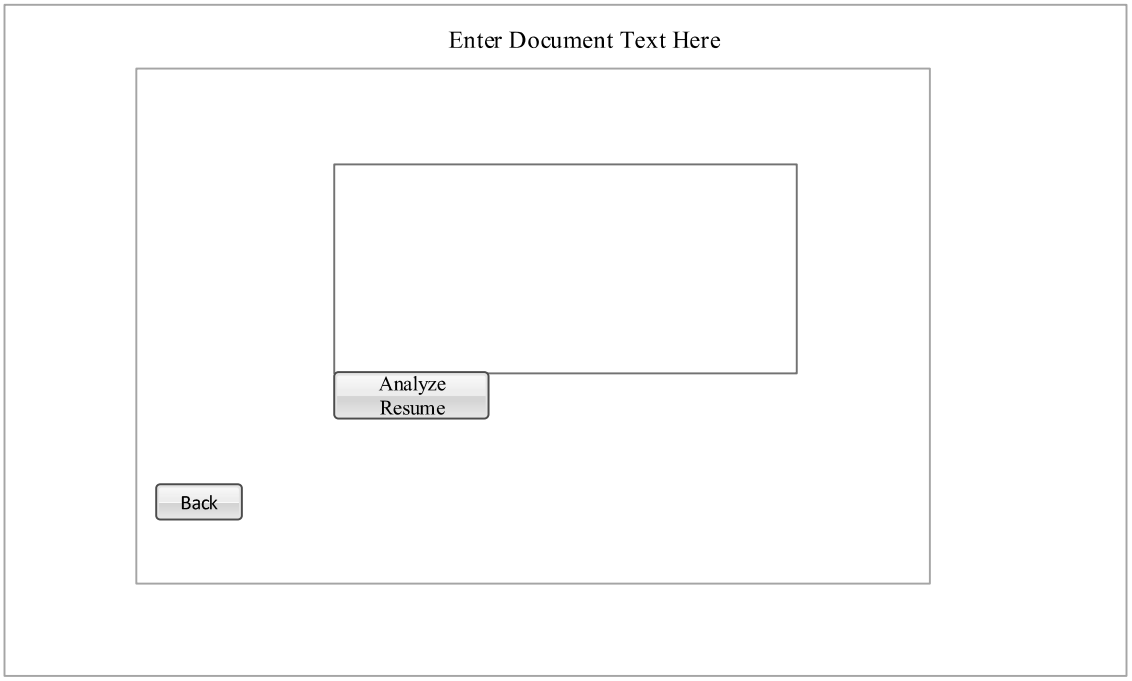


Profile

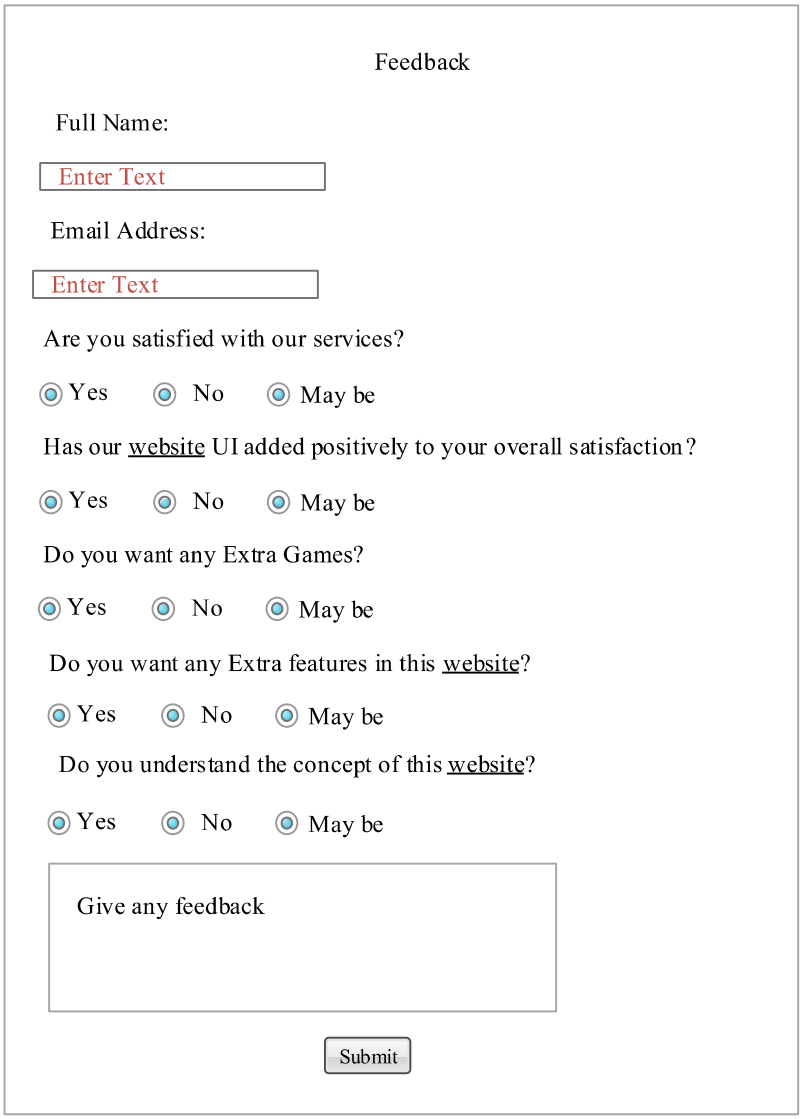
Weightage Of Profile



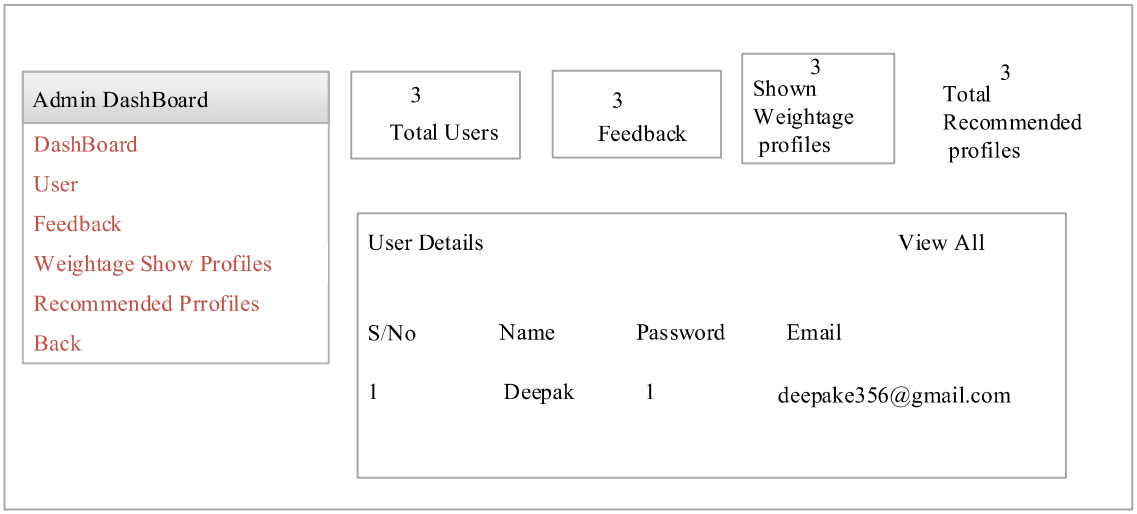
Upload Text Of The Profile



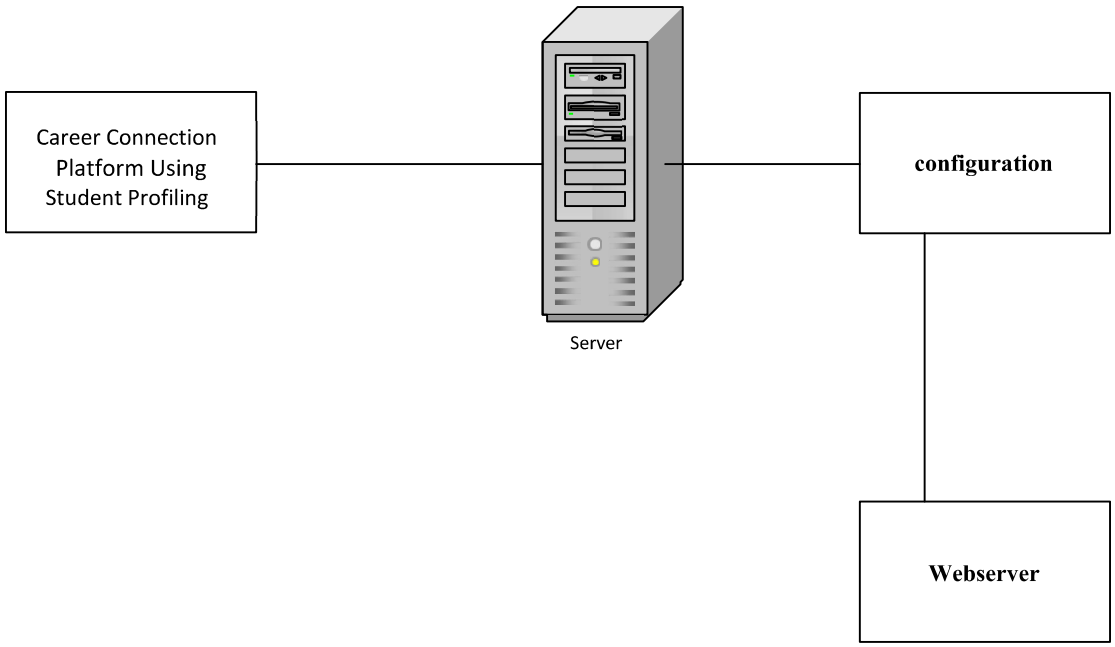
FeedBack


Feedback
Full Name:
Enter Text
Email Address:
Enter Text
Are you satisfied with our services?
Yes
No
May be
Has our website UI added positively to your overall satisfaction?
Yes
No
May be
Do you want any Extra Games?
Yes
No
May be
Do you want any Extra features in this website?
Yes
No
May be
Do you understand the concept of this website?
Yes
No
May be
Give any feedback
Submit

Admin DashBroad



4.8 Deployment Diagram



4.9 Summary

This session is described about software analysis and design, data flow diagram, class diagram, activity diagram, E-R diagram and table design.

CHAPTER 5
TEST CASE DESIGN

TEST CASE DESIGN

5.1 Introduction

Testing is the process of evaluating and verifying that a software product or application does what it is supposed to do. It includes preventing bugs, reducing deployment costs and improving performance.

5.2 Types of testing

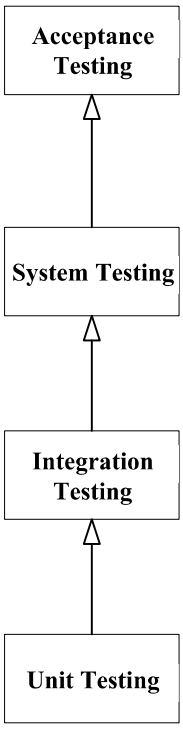
Manual Testing

Manual testing is a type of software testing in which test cases are executed manually by a tester without using any automated tools. The purpose of Manual Testing is to identify the bugs, issues, and defects in the software application.

Automatic Testing

Automated testing is a technique where software tests are executed automatically by a software tool or script, which compares the actual test results with the expected results.

5.3 Levels of Testing



Unit Testing

Unit testing is a type of software testing that focuses on individual units or components of a software system. The purpose of unit testing is to validate that each unit of the software works as intended and meets the requirements.

Test No	**Module Name**	**Test Case**	**Expected Input**	**Expected Output**	**Test status**
1	Sign in	To allow the user to enter the user details	Username	The username must contain only with alphabets and digits. Or If the username doesn't valid.	Successful Unsuccessful
			Password	The password should contain at least one special character, digits, uppercase letter, and a lowercase letter. Or If the password doesn't valid.	Successful Unsuccessful
			Mobile	The only numbers expected in Mobile Or If the Mobile number doesn't valid.	Successful Unsuccessful
			Email	Email is valid to the email format Or If the email doesn't valid.	Successful Unsuccessful
			Address	Address Should Contain be address Or If Address is empty doesn't Valid	Successful Unsuccessful

2	Login	Checks whether the username and password correct or not.	Email Password	Email is valid to the email format The password should contain at least one special character, digits, uppercase letter, and a lowercase letter.	Successful Successful
3	Upload Text Of The Document	Allow the user to upload text of the document.	Text	If the text uploaded. Or If the text not is connected.	Successful Unsuccessful
4	Upload Document	Allow the user to upload the document.	Word document/pdf.	If the text Uploaded Or If the text not uploaded.	Successful Unsccessful

Integration Testing

Integration testing is a software testing technique that tests the interaction between different software modules or components. It is conducted after unit testing and before system testing.

Test No	Module Name	Test Case	Expected Input	Expected Output	Test status
1	Login & Sign in	Checks whether the email and password retrieved from sign up form.	Email Password	If the Email and password is in signup form, it receives. Or If the Email and password is not in the signup form, it doesn't receive.	Successful Unsuccessful

System Testing

System testing is a type of software testing that evaluates the overall functionality and performance of a complete and fully integrated software solution. It is conducted after integration testing and before acceptance testing. The purpose of system testing is to ensure that the software meets the customer's requirements and specifications, and to identify any defects or issues that may arise when the software is deployed to the end users.

Acceptance Testing

Acceptance testing is a type of software testing that is performed to determine whether the software meets the customer's requirements and specifications. It is conducted as a formal testing process based on user requirements and function processing, and it determines whether the software is conforming to specified requirements and user requirements or not .

5.4 Summary

The chapter describes Test design case and in next chapter describe Conclusion.

CHAPTER 6
CONCLUSION

CONCLUSION

The feasibility study strongly supports the development of a Career Connection Platform centered on student profiling, emphasizing its potential to narrow the gap between educational experiences and career aspirations. This entails leveraging advanced data collection techniques and scalable algorithms for effective student data analysis, while ensuring stringent security measures and compliance with data protection regulations. The platform's success also hinges on intuitive user interface design for a seamless experience across devices, integration with existing platforms for enhanced functionality, and viable revenue streams such as subscriptions, advertisements, and partnerships to ensure sustainable income generation. Operationally, user adoption and satisfaction play pivotal roles in the platform's success. Comprehensive training, ongoing user support, and active solicitation of user feedback are crucial for fostering a positive user experience and driving platform usage. Additionally, maintaining optimal platform performance, implementing timely updates, and planning for scalability are essential for its continued relevance and effectiveness in addressing the evolving needs of students, educational institutions, and employers. Despite potential challenges related to data privacy, regulatory compliance, and market competition, upholding ethical standards and transparent communication can build trust and contribute to the platform's long-term success in revolutionizing career guidance and opportunities for individuals.

CHAPTER 7
REFERENCE

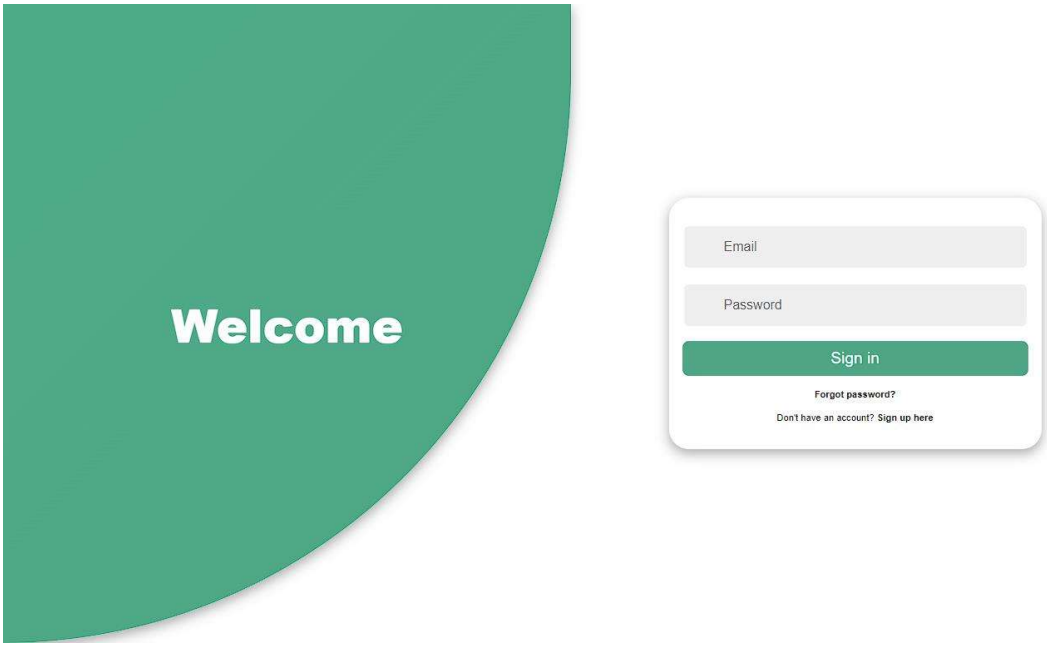
REFERENCE

- https//www.w3schools.com/html/html_responsive.asp
- https//griddb.net/en/blog/k-nearest-neighbor-algorithm-in-java/
- https//themewagon.com/blog/loginsign-up-form-to-compliment-your-website/
- https//getbootstrap.com/docs/4.0/getting-started/introduction/
- https//getbootstrap.com/docs/4.0/utilities/embed/

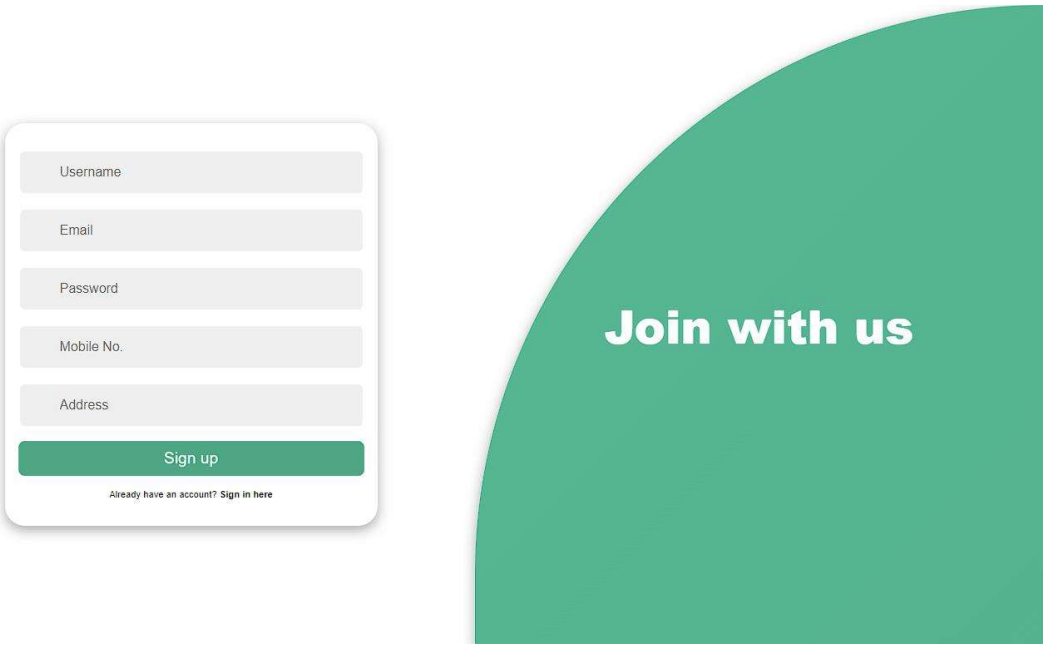
CHAPTER 8
BIBLIOGRAPHY

Screenshot

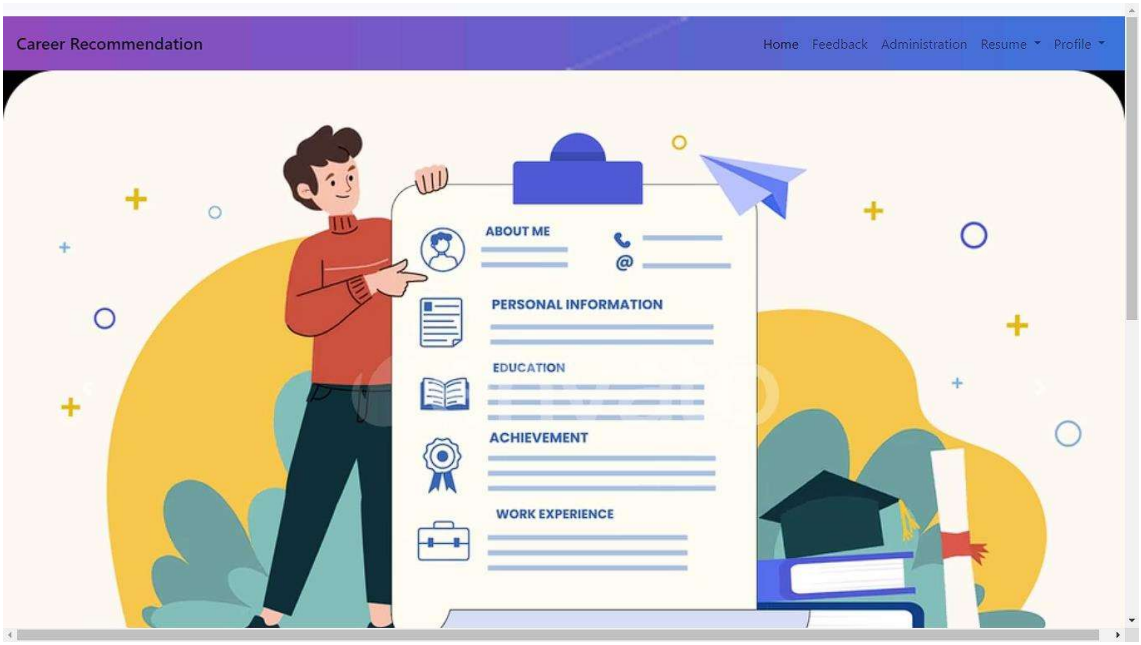
Login Page

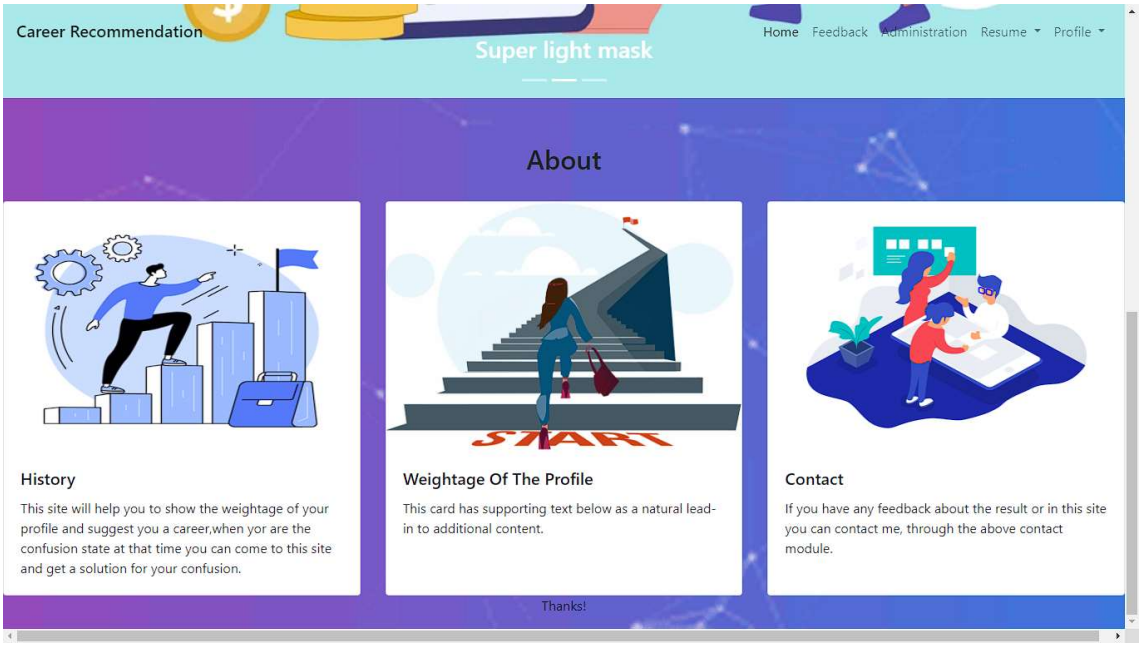


Sign Up Page

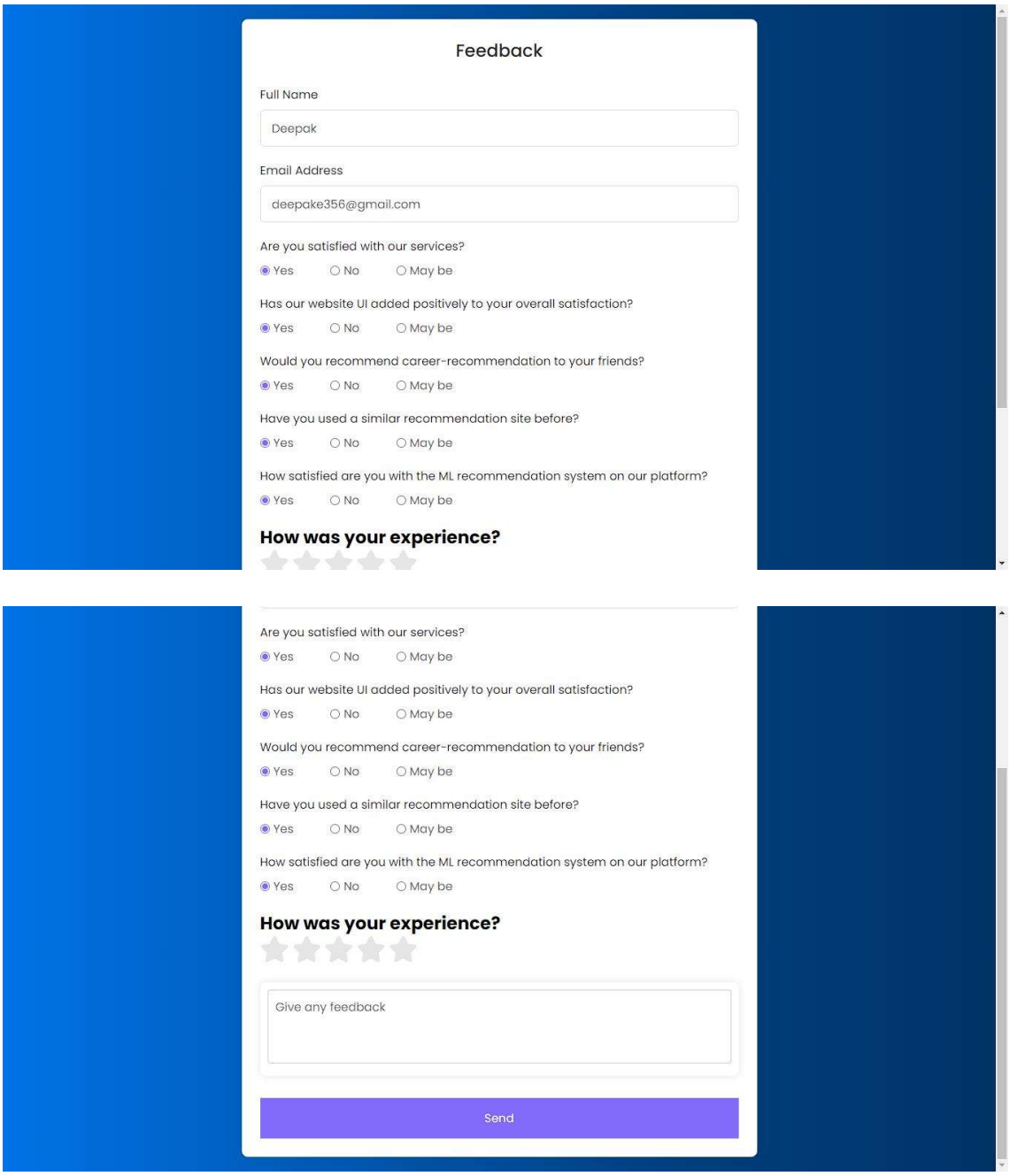


Home Page

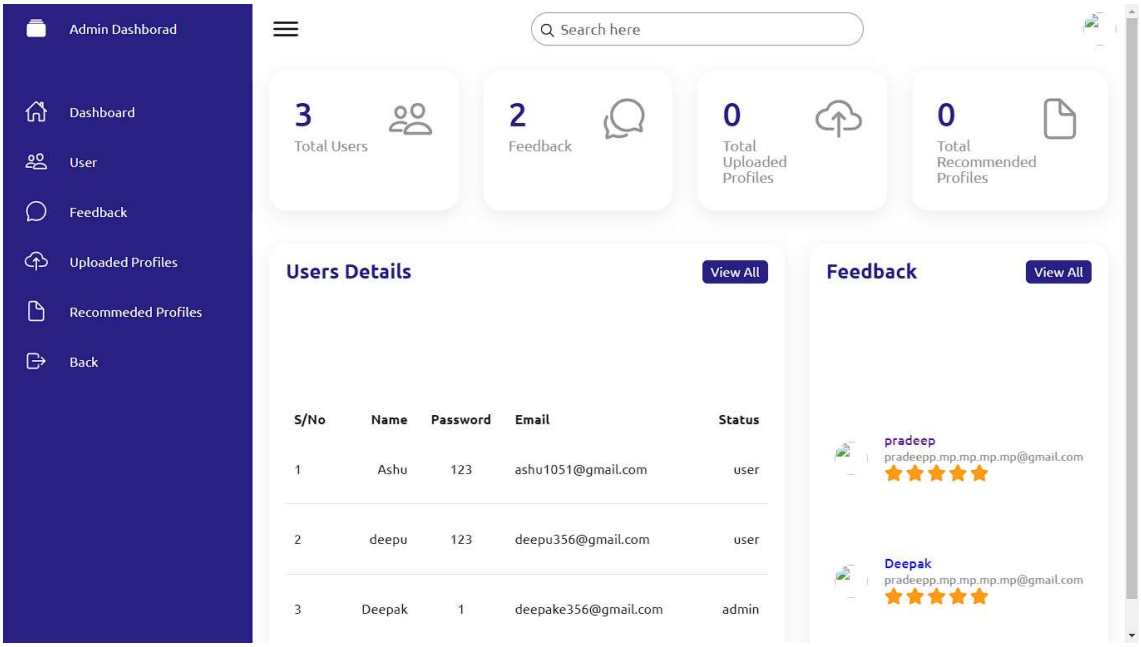

Career Recommendation
Home
Feedback
Administration
Resume
Profile
ABOUT ME
PERSONAL INFORMATION
EDUCATION
ACHIEVEMENT
WORK EXPERIENCE

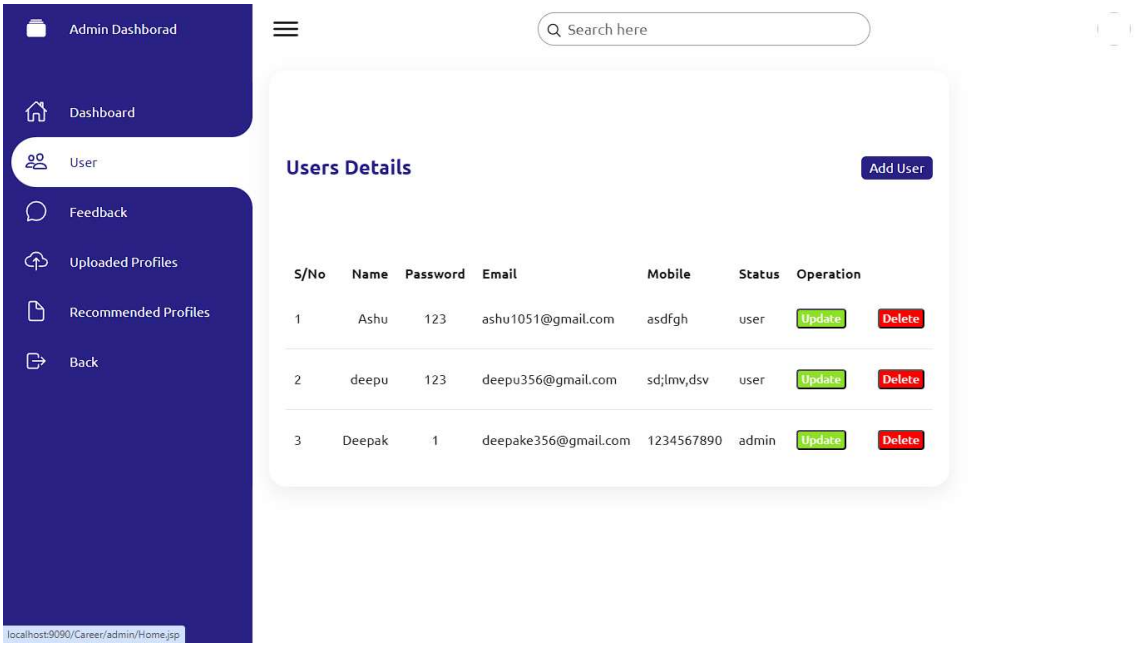

Career Recommendation
Super light mask
Home
Feedback
Administration
Resume
Profile
About
History
This site will help you to show the weightage of your profile and suggest you a career,when yor are the confusion state at that time you can come to this site and get a solution for your confusion.
Weightage Of The Profile
This card has supporting text below as a natural lead-in to additional content.
Contact
If you have any feedback about the result or in this site you can contact me, through the above contact module.
Thanks!

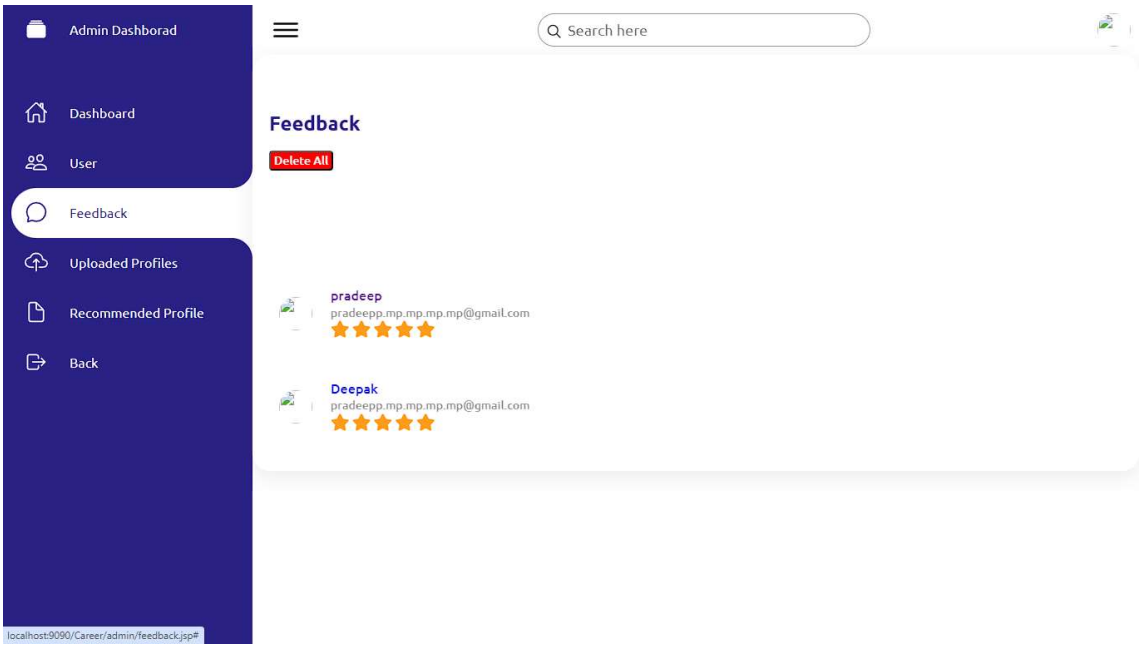
Feedback Page



Admin Dashboard







Weightage Of The Profile

Upload Text Of The Profile

SOURCE CODE

Login Page

index.jsp:

```
<%@ page language="java" contentType="text/html; charset=UTF-8"

pageEncoding="UTF-8"%>

<!DOCTYPE html>

<!--Code by Divinector - divinectorweb.com-->

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Career Recommendation</title>

<link href="https://fonts.googleapis.com/style?family=Baloo+2:wght@700&display=swap"
rel="stylesheet">

<link href="style.css" rel="stylesheet">

</head>

<body>

<%

HttpSession ce=request.getSession();

ce.invalidate();

%>

<div id="container" class="container">

<!-- FORM SECTION -->
```

```
<div class="row">

<!-- SIGN UP -->

<div class="col align-items-center flex-col sign-up">

<div class="form-wrapper align-items-center">

<div class="form sign-up">

<form action="register" method="post" class="login">

<div class="input-group">

<i class='bx bxs-user'></i>

<input type="text" placeholder="Username" name="na" >

</div>

<div class="input-group">

<i class='bx bx-mail-send'></i>

<input type="email" placeholder="Email" name="email">

</div>

<div class="input-group">

<i class='bx bxs-lock-alt'></i>

<input type="password" placeholder="Password" name="pass">

</div>

<div class="input-group">

<i class='bx bxs-lock-alt'></i>

<input type="text" placeholder="Mobile No." name="mob">

</div>

<div class="input-group">
```

```
<i class='bx bxs-lock-alt'></i>

<input type="text" placeholder="Address" name="add">

</div>

<button>

Sign up

</button>

<p>

<span>

Already have an account?

</span>

<b onclick="toggle()" class="pointer">

Sign in here

</b>

</p>

</form>

</div>

</div>

</div>

<!-- END SIGN UP -->

<!-- SIGN IN -->

<div class="col align-items-center flex-col sign-in">

<div class="form-wrapper align-items-center">

<div class="form sign-in">
```

```
<form action="login" method="post" class="login">
<div class="input-group">
<i class='bx bxs-user'></i>
<input type="text" placeholder="Email" name="em">
</div>
<div class="input-group">
<i class='bx bxs-lock-alt'></i>
<input type="password" placeholder="Password" name="pa">
</div>
<button>
Sign in
</button>
<p>
<b>
Forgot password?
</b>
</p>
<p>
<span>
Don't have an account?
</span>
<b onclick="toggle()" class="pointer">
Sign up here
```

```
</b>

</p>

</form>

</div>

</div>

<div class="form-wrapper">

</div>

</div>

<!-- END SIGN IN -->

</div>

<!-- END FORM SECTION -->

<!-- CONTENT SECTION -->

<div class="row content-row">

<!-- SIGN IN CONTENT -->

<div class="col align-items-center flex-col">

<div class="text sign-in">

<h2>

Welcome

</h2>

</div>

<div class="img sign-in">

</div>

</div>
```

```
<!-- END SIGN IN CONTENT -->

<!-- SIGN UP CONTENT -->

<div class="col align-items-center flex-col">

<div class="img sign-up">

</div>

<div class="text sign-up">

<h2>

Join with us

</h2>

</div>

</div>

<!-- END SIGN UP CONTENT -->

</div>

<!-- END CONTENT SECTION -->

</div>

<script src="script.js"></script>

</body>

</html>
```

Java code

Login.java

```
package sam1;

import java.io.IOException;

import java.io.PrintWriter;
```

```
import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.Statement;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import javax.servlet.http.HttpSession;

 Servlet implementation class Login

@WebServlet("/login")

public class Login extends HttpServlet {

private static final long serialVersionUID = 1L;

 @throws IOException

 @see HttpServlet#HttpServlet()

public Login(){

super();

}

 @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)

protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {

try {
```

```
String em1=request.getParameter("em");

String na1=request.getParameter("pa");

Class.forName("com.mysql.jdbc.Driver");

Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/recommend",
"root", "");

String query="Select * from user where email='"+em1+"'";

Statement st=con.createStatement();

ResultSet rs=st.executeQuery(query);

PrintWriter res=response.getWriter();

String
dbemail=null,dbname=null,dbaccess=null,dbmobile=null,dbaddress=null,dbpassword=null;

while(rs.next()) {

dbemail=rs.getString("email");

dbname=rs.getString("name");

dbaccess=rs.getString("access");

dbmobile=rs.getString("mobile");

dbaddress=rs.getString("address");

dbpassword=rs.getString("password");

}

if(dbemail!=null) {

if(em1.equalsIgnoreCase(dbemail) && na1.equalsIgnoreCase(dbpassword)) {

HttpSession ses=request.getSession();

ses.setAttribute("email", em1);
```

```
ses.setAttribute("name", dbname);

ses.setAttribute("access", dbaccess);

ses.setAttribute("mobile", dbmobile);

ses.setAttribute("address", dbaddress);

ses.setAttribute("password", dbpassword);

res.println("<script>");

res.println("window.location.replace('Home.jsp?access="+dbaccess+"&email="+em1+"');");

res.println("</script>");

}

else {

res.println("<script>");

res.println("window.location.replace('index.jsp?id=Password was Wrong&code=0');");

res.println("</script>");

}

}

else {

res.println("<script>");

res.println("window.location.replace('index.jsp?id=The user was not Found&code=0');");

res.println("</script>");

}

}catch(Exception e) {

PrintWriter res=response.getWriter();

res.println(e);
```

```
}

}

}
```

Register.java

```
package sam1;

import java.io.FileInputStream;

import java.io.IOException;

import java.io.PrintWriter;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.Statement;

import java.sql.ResultSet;

import java.util.HashMap;

import javax.servlet.RequestDispatcher;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

Servlet implementation class Register

@WebServlet("/register")

public class Register extends HttpServlet {

private static final long serialVersionUID = 1L;
```

```
 @see HttpServlet#HttpServlet()

public Register() {

super();

}

@see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)

protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {

try {

String em=request.getParameter("email");

String ps=request.getParameter("pass");

String ad=request.getParameter("add");

String mob=request.getParameter("mob");

String na1=request.getParameter("na");

String acc=request.getParameter("ac");

Class.forName("com.mysql.jdbc.Driver");

Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/recommend",
"root", "");

String q;

if(acc==null || acc=="" || acc=="") {

q="insert into user(email,name,password,address,mobile,access)
values('"+em+"','"+na1+"','"+ps+"','"+ad+"','"+mob+"','user')";

}else {

q="insert into user(email,name,password,address,mobile,access)
values('"+em+"','"+na1+"','"+ps+"','"+ad+"','"+mob+"','"+acc+"')";
```

```
}
String query="Select * from user where email='"+em+"'";
String e1="";
String n1="";
PrintWriter re=response.getWriter();
Statement st=con.createStatement();
ResultSet rs=st.executeQuery(query);
if(rs.next()) {
e1=rs.getString("email");
}
if(em.equalsIgnoreCase(e1)) {
if(acc==null || acc=="" || acc==" ") {
re.println("<script>");
re.println("window.location.replace('index.jsp?id=The User was already Found&code=0');");
re.println("</script>");
}else {
re.println("<script>");
re.println("window.location.replace('./admin/Home.jsp?id=The User was already
Found&code=0');");
re.println("</script>");
}
}
else{
```

```
Statement fs=con.createStatement();

fs.executeUpdate(q);

if(acc==null || acc=="" || acc==" ") {

re.println("<script>");

re.println("window.location.replace('index.jsp?id=Registered Successfully&code=1');");

re.println("</script>");

}else {

re.println("<script>");

re.println("window.location.replace('./admin/Home.jsp?id=New User Added
Successfully&code=1');");

re.println("</script>");

}

}

response.getWriter().close();

}

catch(Exception e) {

response.getWriter().println(e);

}

}

}
```

Landing Page

Home.jsp:

```
<!doctype html>
```

```
<html lang="en">

<head>

<title>Title</title>

<!-- Required meta tags -->

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<!-- Bootstrap CSS -->

<link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
integrity="sha384-
ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
crossorigin="anonymous">

<style>

body {

background-image: url(./image/14.gif);

background-size: cover;

background-position: center;

background-repeat: no-repeat;

margin: 0;

padding: 0;

}

</style>

</head>

<body>
```

<nav class="navbar navbar-light bg-light">

</nav>

<nav class="navbar navbar-expand-lg navbar-light bg-transparent justify-content-between sticky-top">

<h2>Career Recommendation</h2>

<button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNavDropdown" aria-controls="navbarNavDropdown" aria-expanded="false" aria-label="Toggle navigation">

</button>

<ul class="navbar-nav">

<li class="nav-item active">

Home (current)

<li class="nav-item">

Feedback

<li class="nav-item"><%

String id=request.getParameter("access");

String email=request.getParameter("email");

String a=request.getParameter("id");

if(id.equalsIgnoreCase("admin")){

```
%>

<a class="nav-link" href="Admin.jsp">Administration</a>

</li>

<%

}

%>

<li class="nav-item dropdown">

<a class="nav-link dropdown-toggle" href="#" id="navbarDropdownMenuLink" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">

Resume

</a>

<div class="dropdown-menu" aria-labelledby="navbarDropdownMenuLink" >

<a class="nav-link" href="./upload/insert2.jsp">Upload Document</a>

<a class="dropdown-item" href="./upload/index2.jsp">Upload Text Resume</a>

</div>

<li class="nav-item dropdown">

<a class="nav-link dropdown-toggle" href="#" id="navbarDropdownMenuLink" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">

Profile

</a>

<div class="dropdown-menu" aria-labelledby="navbarDropdownMenuLink">

<a class="dropdown-item" href="./profile/profile.jsp">My Profile</a>

<a class="dropdown-item" href="index.jsp">Logout</a>
```

```
</div>

</li>

</ul>

</div>

</div>

</nav>

<div id="video-carousel-example2" class="carousel slide carousel-fade" data-ride="carousel">

<!--Indicators-->

<ol class="carousel-indicators">

<li data-target="#video-carousel-example2" data-slide-to="0" class="active"></li>

<li data-target="#video-carousel-example2" data-slide-to="1"></li>

<li data-target="#video-carousel-example2" data-slide-to="2"></li>

</ol>

<!--/.Indicators-->

<!--Slides-->

<div class="carousel-inner" role="listbox">

<!-- First slide -->

<div class="carousel-item active">

<!--Mask color-->

<!--Video source-->

<video class="video-fluid" style="width:100%;height:100%;" autoplay loop muted>

<source src="./image/11.mp4" style="width:640px;height:400px;" type="video/mp4"/>
```

```
</video>
<div class="mask rgba-black-strong"></div
<!--Caption-->
<div class="carousel-caption">
<div class="animated fadeInDown">
<h3 class="h3-responsive">Light mask</h3>
</div>
</div>
<!--Caption-->
</div>
<div class="carousel-item">
<video class="video-fluid" style="width:100%;height:100%;" autoplay loop muted>
<source src="./image/12.mp4" type="video/mp4" style="width:640px;height:400px;" />
</video>
<div class="mask rgba-purple-slight"></div>
<div class="carousel-caption">
<div class="animated fadeInDown">
<h3 class="h3-responsive">Super light mask</h3>
</div>
</div>
</div>
<div class="carousel-item">
<div class="view">
```

```
<video class="video-fluid" style="width:100%;height:100%;" autoplay loop muted>
<source src="./image/13.mp4"  style="width:640px;height:400px;" type="video/mp4"  />
</video>
<div class="mask rgba-black-strong"></div>
</div>
<div class="carousel-caption">
<div class="animated fadeInDown">
<h3 class="h3-responsive">Strong mask</h3>
</div>
</div>
</div>
</div>
<a class="carousel-control-prev" href="#video-carousel-example2" role="button" data-slide="prev">
<span class="carousel-control-prev-icon" aria-hidden="true"></span>
<span class="sr-only">Previous</span>
</a>
<a class="carousel-control-next" href="#video-carousel-example2" role="button" data-slide="next">
<span class="carousel-control-next-icon" aria-hidden="true"></span>
<span class="sr-only">Next</span>
</a>
</div>
```

```
<br/>

<br/>

<center>

<p class="h2">About</p>

</center>

<br>

<div class="card-deck">

<div class="card">

<img class="card-img-top" src="./image/8.gif" alt="Card image cap" height="300px" >

<div class="card-body">

<h5 class="card-title">History</h5>

<p class="card-text">This site will help you to show the weightage of your profile and suggest you a career,when yor are the confusion state at that time you can come to this site and get a solution for your confusion.  </p>

</div>

</div>

<div class="card">

<img class="card-img-top" src="./image/9.gif" alt="Card image cap" height="300px" >

<div class="card-body">

<h5 class="card-title">Weightage Of The Profile</h5>

<p class="card-text">This card has supporting text below as a natural lead-in to additional content.</p>

</div>

</div>
```

<img class="card-img-top" src="./image/6.gif" alt="Card image cap" height="300px" >

<h5 class="card-title">Contact</h5>

If you have any feedback about the result or in this site you can contact me, through the above contact module.

Thanks!

<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo" crossorigin="anonymous"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1" crossorigin="anonymous"></script>

<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM" crossorigin="anonymous"></script>

</body>

</html>

FeedBack

feedback.jsp

<%@page import="java.sql.Statement"%>

```
<%@page import="java.sql.DriverManager"%>
<%@page import="java.sql.Connection"%>
<%@page import="java.sql.ResultSet"%>
<%
HttpSession ses=request.getSession();
if(ses.getAttribute("name")==null || ses.getAttribute("name")==""){
%>
<script type="text/javascript">
window.location.replace("../error.jsp");
</script>
<%} %>
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<meta http-equiv="X-UA-Compatible" content="ie=edge" />
<title>User Feedback</title>
<link rel="stylesheet" href="style.css" />
<link rel="stylesheet" href="star.css" />
<style>
.containers {
max-width: 700px;
```

```
width: 100%;

padding: 10px;

box-sizing: border-box;

background-color: #fff;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

border-radius: 8px;

}

textarea {

width: 100%;

height: 100px;

padding: 10px;

box-sizing: border-box;

border: 1px solid #ccc;

border-radius: 4px;

resize: none;

font-size: 16px;

}

</style>

<script src="script.js" defer></script>

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.2.1/css/all.min.css" />

</head>

<%
```

```
String id=request.getParameter("feedback");

if(id==null || id==" "){

%>

<body>

<section class="container">

<header>Feedback</header>

<form action="sendfeedback.jsp" class="form" name="form1" method="post"
onSubmit="return go()">

<div class="input-box">

<label>Full Name</label>

<input type="text" name="name" value="<%=ses.getAttribute("name")%>"
placeholder="Enter full name" required readonly="readonly" />

</div>

<div class="input-box">

<label>Email Address</label>

<input type="text" name="email" value="<%=ses.getAttribute("email")%>"
placeholder="Enter email address" required readonly="readonly" />

</div>

<div class="gender-box">

<h3>Are you satisfied with our services?</h3>

<div class="gender-option">

<div class="gender">

<input type="radio" id="check-male" name="q1" value="yes" checked />

<label for="check-male">Yes</label>
```

```
</div>
<div class="gender">
<input type="radio" id="check-female" name="q1" value="no" />
<label for="check-female">No</label>
</div>
<div class="gender">
<input type="radio" id="check-other" name="q1" value="may be" />
<label for="check-other">May be</label>
</div>
</div>
</div>
<div class="gender-box">
<h3>Has our website UI added positively to your overall satisfaction?</h3>
<div class="gender-option">
<div class="gender">
<input type="radio" id="check-male" name="q2" value="yes" checked />
<label for="check-male">Yes</label>
</div>
<div class="gender">
<input type="radio" id="check-female" name="q2" value="no" />
<label for="check-female">No</label>
</div>
<div class="gender">
```

```
<input type="radio" id="check-other" name="q2" value="may be" />
<label for="check-other">May be</label>
</div>
</div>
</div>
<div class="gender-box">
<h3>Would you recommend career-recommendation to your friends?</h3>
<div class="gender-option">
<div class="gender">
<input type="radio" id="check-male" name="q3" value="yes" checked />
<label for="check-male">Yes</label>
</div>
<div class="gender">
<input type="radio" id="check-female" name="q3" value="no" />
<label for="check-female">No</label>
</div>
<div class="gender">
<input type="radio" id="check-other" name="q3" value="may be" />
<label for="check-other">May be</label>
</div>
</div>
</div>
<div class="gender-box">
```

```
<h3>Have you used a similar recommendation site before?</h3>

<div class="gender-option">

<div class="gender">

<input type="radio" id="check-male" name="q4" value="yes" checked />

<label for="check-male">Yes</label>

</div>

<div class="gender">

<input type="radio" id="check-female" name="q4" value="no" />

<label for="check-female">No</label>

</div>

<div class="gender">

<input type="radio" id="check-other" name="q4" value="may be" />

<label for="check-other">May be</label>

</div>

</div>

</div>

<div class="gender-box">

<h3> How satisfied are you with the ML recommendation system on our platform?</h3>

<div class="gender-option">

<div class="gender">

<input type="radio" id="check-male" name="q5" value="yes" checked />

<label for="check-male">Yes</label>

</div>
```

```
<div class="gender">
<input type="radio" id="check-female" name="q5" value="no" />
<label for="check-female">No</label>
</div>
<div class="gender">
<input type="radio" id="check-other" name="q5" value="may be" />
<label for="check-other">May be</label>
</div>
</div>
</div>
<div class="input-box">
<h2>How was your experience?</h2>
<div class="stars">
<i class="fa-solid fa-star"></i>
<i class="fa-solid fa-star"></i>
<i class="fa-solid fa-star"></i>
<i class="fa-solid fa-star"></i>
<i class="fa-solid fa-star"></i>
</div>
</div>
<br>
<div class="containers">
<textarea placeholder="Give any feedback" name="messages"></textarea>
```

```
</div>

<input type="text" id="number" name="number" hidden >

<input type="text" id="number" name="access" value="<%=ses.getAttribute("access")%>"
hidden >

<input type="submit" value="Send" class="g1" />

</form>

</section>

</body>

<%

}else{

Class.forName("com.mysql.jdbc.Driver");

Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/recommend",
"root", "");

String query="Select * from feedback where id="+id;

Statement st=con.createStatement();

ResultSet de=st.executeQuery(query);

while(de.next()){

String star=de.getString("star");

%>

<body>

<section class="container">

<header>Feedback</header>

<form action="sendfeedback.jsp" class="form" name="form1" method="post">
```

<label>Full Name</label>

<input type="text" name="name" value="<%=de.getString("name")%>" placeholder="Enter full name" required readonly="readonly" />

<label>Email Address</label>

<input type="text" name="email" value="<%=de.getString("email")%>" placeholder="Enter email address" required readonly="readonly" />

<h3>Are you satisfied with our services?</h3>

<input type="radio" id="check-male" name="q1" value="yes" checked />

<label for="check-male"><%=de.getString("q1")%></label>

<h3>Has our website UI added positively to your overall satisfaction?</h3>

```
<input type="radio" id="check-male" name="q2" value="yes" checked />
<label for="check-male"><%=de.getString("q2")%></label>
</div>
</div>
</div>
<div class="gender-box">
<h3>Would you recommend Plant recommend for work to your friends?</h3>
<div class="gender-option">
<div class="gender">
<input type="radio" id="check-male" name="q3" value="yes" checked />
<label for="check-male"><%=de.getString("q3")%></label>
</div>
</div>
</div>
<div class="gender-box">
<h3>Have you used a similar Plant recommend before?</h3>
<div class="gender-option">
<div class="gender">
<input type="radio" id="check-male" name="q4" value="yes" checked />
<label for="check-male"><%=de.getString("q4")%></label>
</div>
</div>
</div>
```

```
<div class="gender-box">
<h3>Do you feel our Plant recommend is worth the cost?</h3>
<div class="gender-option">
<div class="gender">
<input type="radio" id="check-male" name="q5" value="yes" checked />
<label for="check-male"><%=de.getString("q5")%></label>
</div>
</div>
</div>
<div class="input-box">
<h2>How was your experience?</h2>
<%
if(star==null ||star==" "){
%>
<i id="i1" class="fa-solid fa-star"></i>
<i id="i1" class="fa-solid fa-star"></i>
<i id="i1" class="fa-solid fa-star"></i>
<i id="i1" class="fa-solid fa-star"></i>
<i id="i1" class="fa-solid fa-star"></i>
<%}
else if(star.equalsIgnoreCase("0")){
%>
<i id="i2" class="fa-solid fa-star"></i>
```

```
<i id="i1" class="fa-solid fa-star"></i>

<i id="i1" class="fa-solid fa-star"></i>

<i id="i1" class="fa-solid fa-star"></i>

<i id="i1" class="fa-solid fa-star"></i>

<%}else if(star.equalsIgnoreCase("1")){

%>

<i id="i2" class="fa-solid fa-star"></i>

<i id="i2" class="fa-solid fa-star"></i>

<i id="i1" class="fa-solid fa-star"></i>

<i id="i1" class="fa-solid fa-star"></i>

<i id="i1" class="fa-solid fa-star"></i>

<%}

else if(star.equalsIgnoreCase("2")){

%>

<i id="i2" class="fa-solid fa-star"></i>

<i id="i2" class="fa-solid fa-star"></i>

<i id="i2" class="fa-solid fa-star"></i>

<i id="i1" class="fa-solid fa-star"></i>

<i id="i1" class="fa-solid fa-star"></i>

<%}

else if(star.equalsIgnoreCase("3")){

%>

<i id="i2" class="fa-solid fa-star"></i>
```

```
<i id="i2" class="fa-solid fa-star"></i>

<i id="i2" class="fa-solid fa-star"></i>

<i id="i2" class="fa-solid fa-star"></i>

<i id="i1" class="fa-solid fa-star"></i>

<%}

else if(star.equalsIgnoreCase("4")){

%>

<i id="i2" class="fa-solid fa-star"></i>

<i id="i2" class="fa-solid fa-star"></i>

<i id="i2" class="fa-solid fa-star"></i>

<i id="i2" class="fa-solid fa-star"></i>

<i id="i2" class="fa-solid fa-star"></i>

<%

}

%>

</div>

<br>

<div class="containers">

<textarea placeholder="Give any feedback"
readonly="readonly"><%=de.getString("mess")%></textarea>

</div><%

}%>

</form>
```

<a href="../admin/deletefeedback.jsp?option=one&id=<%=id%>"><button class="g2">Delete</button>

<button class="g1">Back</button>

</section>

</body>

<%

}

%>

</html>

Admin Dashbroad

Home.jsp

<%@page import="java.io.PrintWriter"%>

<%@page import="java.sql.Statement"%>

<%@page import="java.sql.DriverManager"%>

<%@page import="java.sql.Connection"%>

<%@page import="java.sql.ResultSet"%>

<%@page import="java.util.List"%>

<%@page import="java.util.List"%>

<!DOCTYPE html>

<html lang="en" dir="ltr">

<head>

<meta charset="UTF-8">

```
<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.2.1/css/all.min.css">

<title>Admin Dashboard</title>

<!-- ======= Styles ====== -->

<link rel="stylesheet" href="style2.css">

<link rel="stylesheet" href="../Toast.css">

</head>

<body>

<div class="container">

<!-- ================ Navigation ================ -->

<div class="navigation">

<ul>

<li>

<a href="">

<span class="icon">

<ion-icon name="albums"></ion-icon>

</span>

<span class="title">Admin Dashborad</span>

</a>

</li>

<li>
```

```
<a href="../Admin.jsp">
<span class="icon">
<ion-icon name="home-outline"></ion-icon>
</span>
<span class="title">Dashboard</span>
</a>
</li>
<li>
<a href="Home.jsp">
<span class="icon">
<ion-icon name="people-outline"></ion-icon>
</span>
<span class="title">User</span>
</a>
</li>
<li>
<a href="feedback.jsp">
<span class="icon">
<ion-icon name="chatbubble-outline"></ion-icon>
</span>
<span class="title">Feedback</span>
</a>
</li>
```

```
<li>
<a href="#">
<span class="icon">
<ion-icon name="cloud-upload-outline"></ion-icon>
</span>
<span class="title">Uploaded Profiles</span>
</a>
</li>
<li>
<a href="#">
<span class="icon">
<ion-icon name="document-outline"></ion-icon>
</span>
<span class="title">Recommended Profiles</span>
</a>
</li>
<li>
<%
HttpSession se=request.getSession();
%>
<a href="../Home.jsp?access=<%=se.getAttribute("access")%>">
<span class="icon">
<ion-icon name="log-out-outline"></ion-icon>
```

```
</span>

<span class="title">Back</span>

</a>

</li>

</ul>

</div>

<!-- ========================== Main ==================== -->

<div class="main">

<div class="topbar">

<div class="toggle">

<ion-icon name="menu-outline"></ion-icon>

</div>

<div class="search">

<label>

<input type="text" placeholder="Search here">

<ion-icon name="search-outline"></ion-icon>

</label>

</div>

<div class="user">

<img alt="">

</div>

</div>

<div class="details">
```

<ul class="notifications">

<h2>Users Details</h2>

Add User

<table>

<thead>

<tr>

<td>S/No</td>

<td>Name</td>

<td>Password</td>

<td>Email</td>

<td>Mobile</td>

<td>Status</td>

<td>Operation</td>

</tr>

</thead>

<tbody>

<%

Class.forName("com.mysql.jdbc.Driver");

Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/recommend", "root", "");

```
Statement st=con.createStatement();

String q="select * from user order by id DESC";

ResultSet re=st.executeQuery(q);

int i=1;

String name,password,email,mobile,status;

while(re.next()){

name=re.getString("name");

password=re.getString("password");

email=re.getString("email");

mobile=re.getString("mobile");

status=re.getString("access");

%>

<tr>

<td><%=i%></td>

<td><%=name%></td>

<td><%=password%></td>

<td><%=email%></td>

<td><%=mobile%></td>

<td><%=status%></td>

<td><a href="update.jsp?id=<%=email%>" ><button class="status
delivered">Update</button></a></td>

<td><a href="delete.jsp?id=<%=email%>" ><button class="status
return">Delete</button></a></td>
```

```
</tr>

<% i++;} %>

</tbody>

</table>

</div>

</div>

</div>

</div>

<script src="main.js"></script>

<script type="module"
src="https://unpkg.com/ionicons@5.5.2/dist/ionicons/ionicons.esm.js"></script>

<script nomodule
src="https://unpkg.com/ionicons@5.5.2/dist/ionicons/ionicons.js"></script>

</body>

</html>
```

update.jsp

```
<%@page import="java.io.PrintWriter"%>

<%@page import="java.sql.Statement"%>

<%@page import="java.sql.DriverManager"%>

<%@page import="java.sql.Connection"%>

<%@page import="java.sql.ResultSet"%>

<%@page import="java.util.List"%>
```

```
<%@page import="java.util.List"%>

<%@ page language="java" contentType="text/html; charset=UTF-8"

pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<title>Add New User</title>

<link rel="stylesheet" href="..\style.css">

</head>

<body>

<%

String a=request.getParameter("id");

Class.forName("com.mysql.jdbc.Driver");

Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/recommend",
"root", "");

String query="select * from user where email='"+a+"'";

Statement st=con.createStatement();

ResultSet de=st.executeQuery(query);

while(de.next()){

%>

<div class="wrapper">

<div class="title-text">

<div class="title login">Update User</div>
```

```
</div>

<div class="form-container">

</div>

<div class="form-inner">

<form action="dataupdate.jsp?id=<%=de.getInt("id")%>" method="post" class="login">

<div class="field">

<input type="text" placeholder="Username" name="na"
value="<%=de.getString("name")%>" required>

</div>

<div class="field">

<input type="text" placeholder="Email Address" name="em"
value="<%=de.getString("email")%>" required readonly="readonly">

</div>

<div class="field">

<input type="password" placeholder="Password" name="pa"
value="<%=de.getString("password")%>" required>

</div>

<div class="field">

<input type="text" placeholder="Mobile" name="mo"
value="<%=de.getString("mobile")%>" required>

</div>

<div class="field">

<input type="text" placeholder="Address" name="add"
value="<%=de.getString("address")%>" required>
```

```
</div>
<%
if(de.getString("access").equalsIgnoreCase("user")){
%>
<div class="field btn">
Access:
<select name="ac">
<option>user</option>
<option>admin</option>
</select>
</div>
<%
}
else{
%>
<div class="field btn">
Access:
<select name="ac">
<option>admin</option>
<option>user</option>
</select>
</div>
<%
```

```
}}
%>
<div class="field btn">
<div class="btn-layer"></div>
<input type="submit" value="Update">
</div>
</form>
</div>
</div>
</body>
<script src="..\script.js"></script>
</html>
```

feedback.jsp

```
<%@page import="java.io.PrintWriter"%>
<%@page import="java.sql.Statement"%>
<%@page import="java.sql.DriverManager"%>
<%@page import="java.sql.Connection"%>
<%@page import="java.sql.ResultSet"%>
<%@page import="java.util.List"%>
<%@page import="java.util.List"%>
<!DOCTYPE html>
<html lang="en">
```

```
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Admin Dashboard</title>
<!-- ======= Styles ====== -->
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.2.1/css/all.min.css" />
<link rel="stylesheet" href="style2.css">
<link rel="stylesheet" href="star.css">
<link rel="stylesheet" href="../Toast.css">
</head>
<body>
<div class="container">
<div class="navigation">
<ul>
<li>
<a href="">
<span class="icon">
<ion-icon name="albums"></ion-icon>
</span>
<span class="title">Admin Dashborad</span>
</a>
```

```
</li>

<li>

<a href="../Admin.jsp">

<span class="icon">

<ion-icon name="home-outline"></ion-icon>

</span>

<span class="title">Dashboard</span>

</a>

</li>

<li>

<a href="Home.jsp">

<span class="icon">

<ion-icon name="people-outline"></ion-icon>

</span>

<span class="title">User</span>

</a>

</li>

<li>

<a href="#">

<span class="icon">

<ion-icon name="chatbubble-outline"></ion-icon>

</span>

<span class="title">Feedback</span>
```

```
</a>

</li>

<li>

<a href="#">

<span class="icon">

<ion-icon name="cloud-upload-outline"></ion-icon>

</span>

<span class="title">Uploaded Profiles</span>

</a>

</li>

<li>

<a href="#">

<span class="icon">

<ion-icon name="document-outline"></ion-icon>

</span>

<span class="title">Recommended Profile</span>

</a>

</li>

<li>

<%

HttpSession se=request.getSession();

%>

<a href="../Home.jsp?access=<%=se.getAttribute("access")%>">
```

```
<span class="icon">
<ion-icon name="log-out-outline"></ion-icon>
</span>
<span class="title">Back</span>
</a>
</li>
</ul>
</div>
<div class="main">
<div class="topbar">
<div class="toggle">
<ion-icon name="menu-outline"></ion-icon>
</div>
<div class="search">
<label>
<input type="text" placeholder="Search here" value="">
<ion-icon name="search-outline"></ion-icon>
</label>
</div>
<div class="user">
<img src="" alt="">
</div>
</div>
```

```
<div class="recentCustomers">

<ul class="notifications"></ul>

<div class="cardHeader">

<h2>Feedback</h2>

<br>

<a href="deletefeedback.jsp"><button class="status return">Delete All</button></a>

</div>

<br>

<table >

<%

Class.forName("com.mysql.jdbc.Driver");

Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/recommend",
"root", "");

Statement st=con.createStatement();

String q1="select * from feedback order by id DESC";

ResultSet rs=st.executeQuery(q1);

String name,email,star;

ResultSet re1=st.executeQuery(q1);

while(re1.next()){

name=re1.getString("name");

email=re1.getString("email");

star=re1.getString("star");

%>
```

```
<tr>
<td width="60px">
<div class="imgBx"><img src="../img/profile.png" alt=""></div>
</td>
<td>
<a id="a1" href="../home/feedback.jsp?feedback=<%=re1.getInt("id")%>">
<h4><%=name%><br> <span><%=email%></span><br> <span><div class="stars">
<%
if(star==null ||star==" "){
%>
<i id="i1" class="fa-solid fa-star"></i>
<i id="i1" class="fa-solid fa-star"></i>
<i id="i1" class="fa-solid fa-star"></i>
<i id="i1" class="fa-solid fa-star"></i>
<i id="i1" class="fa-solid fa-star"></i>
<%}
else if(star.equalsIgnoreCase("0")){
%>
<i id="i2" class="fa-solid fa-star"></i>
<i id="i1" class="fa-solid fa-star"></i>
<i id="i1" class="fa-solid fa-star"></i>
<i id="i1" class="fa-solid fa-star"></i>
<i id="i1" class="fa-solid fa-star"></i>
```

```
<%}else if(star.equalsIgnoreCase("1")){
%>
<i id="i2" class="fa-solid fa-star"></i>
<i id="i2" class="fa-solid fa-star"></i>
<i id="i1" class="fa-solid fa-star"></i>
<i id="i1" class="fa-solid fa-star"></i>
<i id="i1" class="fa-solid fa-star"></i>
<%}
else if(star.equalsIgnoreCase("2")){
%>
<i id="i2" class="fa-solid fa-star"></i>
<i id="i2" class="fa-solid fa-star"></i>
<i id="i2" class="fa-solid fa-star"></i>
<i id="i1" class="fa-solid fa-star"></i>
<i id="i1" class="fa-solid fa-star"></i>
<%}
else if(star.equalsIgnoreCase("3")){
%>
<i id="i2" class="fa-solid fa-star"></i>
<i id="i2" class="fa-solid fa-star"></i>
<i id="i2" class="fa-solid fa-star"></i>
<i id="i2" class="fa-solid fa-star"></i>
<i id="i1" class="fa-solid fa-star"></i>
```

```
<%}
else if(star.equalsIgnoreCase("4")){
%>
<i id="i2" class="fa-solid fa-star"></i>
<i id="i2" class="fa-solid fa-star"></i>
<i id="i2" class="fa-solid fa-star"></i>
<i id="i2" class="fa-solid fa-star"></i>
<i id="i2" class="fa-solid fa-star"></i>
<%
}
%>
</div></span></h4></a>
</td>
</tr>
<%
}
%>
</table>
</div>
</div>
</div>
</div>
<script src="main.js"></script>
```

<script type="module"
src="https://unpkg.com/ionicons@5.5.2/dist/ionicons/ionicons.esm.js"></script>

<script nomodule
src="https://unpkg.com/ionicons@5.5.2/dist/ionicons/ionicons.js"></script>

</body>

</html>

deletefeedback.jsp

<%@page import="java.io.PrintWriter"%>

<%@page import="java.sql.Statement"%>

<%@page import="java.sql.DriverManager"%>

<%@page import="java.sql.Connection"%>

<%@page import="java.sql.ResultSet"%>

<%@page import="java.util.List"%>

<%

PrintWriter res=response.getWriter();

String option=request.getParameter("option");

if(option==null ||option==" "){

Class.forName("com.mysql.jdbc.Driver");

Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/recommend",
"root", "");

String query="delete from feedback";

Statement st=con.createStatement();

st.executeUpdate(query);

```
res.println("<script>");

res.println("window.location.replace('feedback.jsp?id=Deleted Successfully&code=1');");

res.println("</script>");


}else{

String id=request.getParameter("id");

Class.forName("com.mysql.jdbc.Driver");

Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/crop", "root",
"");

String query="delete from feedback where id="+id;

Statement st=con.createStatement();

st.executeUpdate(query);

res.println("<script>");

res.println("window.location.replace('feedback.jsp?id=Deleted Successfully&code=1');");

res.println("</script>");

}

%>
```

delete.jsp

```
<%@page import="java.io.PrintWriter"%>

<%@page import="java.sql.Statement"%>

<%@page import="java.sql.DriverManager"%>

<%@page import="java.sql.Connection"%>
```

```
<%@page import="java.sql.ResultSet"%>

<%@page import="java.util.List"%>

<%@page import="java.util.List"%>

<%

String a=request.getParameter("id");

Class.forName("com.mysql.jdbc.Driver");

Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/recommend",
"root", "");

String query="delete from user where email='"+a+"'";

Statement st=con.createStatement();

st.executeUpdate(query);

%>

<script>

window.location.replace("Home.jsp?id=Deleted Successfully&code=1");

</script>
```

dataupdate.jsp

```
<%@page import="java.io.PrintWriter"%>

<%@page import="java.sql.Statement"%>

<%@page import="java.sql.DriverManager"%>

<%@page import="java.sql.Connection"%>

<%@page import="java.sql.ResultSet"%>

<%@page import="java.util.List"%>
```

```
<%@page import="java.util.List"%>

<%

String email=request.getParameter("em");

String ps=request.getParameter("pa");

String ad=request.getParameter("add");

String mob=request.getParameter("mo");

String na1=request.getParameter("na");

String acc=request.getParameter("ac");

String id=request.getParameter("id");

Class.forName("com.mysql.jdbc.Driver");

Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/recommend",
"root", "");

Statement st=con.createStatement();

String q1="update user set
email='"+email+"',name='"+na1+"',password='"+ps+"',address='"+ad+"',mobile='"+mob+"',a
ccess='"+acc+"' where id="+id;

st.executeUpdate(q1);

%>

<script>

window.location.replace("Home.jsp?id=Updated Successfully&code=1");

</script>
```

Weightage Of The Profile

```
<%@ page import="java.io.IOException" %>

<%@ page import="java.io.PrintWriter" %>
```

```
<%@ page import="java.util.HashMap" %>
<%@ page import="java.util.Map" %>
<%@ page import="org.apache.poi.openxml4j.exceptions.InvalidFormatException" %>
<%@ page import="org.apache.poi.xwpf.usermodel.XWPFDocument" %>
<%@ page import="org.apache.poi.xwpf.extractor.XWPFWordExtractor" %>
<%@ page contentType="text/html;charset=UTF-8" language="java" %>
<%@ page import="java.io.*" %>
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Career Suggestion</title>
    <style>
      body {
          font-family: Arial, sans-serif;
          background-image: url(./image/2.gif); /* Add your GIF URL */
          background-size: cover; /* Cover the whole page */
          background-position: center; /* Center the background image */
          margin: 0;
          padding: 0;
      }
      h2 {
          color: #333;
```

```
}
form {
    background-color: #fff;
    padding: 20px;
    margin: 20px auto;
    width: 50%;
    border-radius: 5px;
    box-shadow: 0 2px 5px rgba(0,0,0,0.1);
}
textarea {
    width: 100%;
    padding: 10px;
    margin-bottom: 10px;
    border: 1px solid #ccc;
    border-radius: 5px;
}
input[type="submit"] {
    padding: 10px 20px;
    background-color: #007bff;
    color: #fff;
    border: none;
    border-radius: 5px;
    cursor: pointer;
```

```
        }
        input[type="submit"]:hover {
            background-color: #0056b3;
        }
        .result-container {
            background-color: #fff;
            padding: 20px;
            margin: 20px auto;
            width: 50%;
            border-radius: 5px;
            box-shadow: 0 2px 5px rgba(0,0,0,0.1);
        }
        p {
            color: #333;
            margin-bottom: 10px;
        }


    </style>
</head>
<body>


<h2>Enter Your Resume</h2>
<form method="post">
```

```
    <textarea name="resumeText" rows="10" cols="50"></textarea><br>
    <input type="submit" value="Analyze Resume">

</form>

<%
    response.setContentType("text/html");
    PrintWriter printWriter = response.getWriter();

    // Retrieve resume text from the form
    String resumeText = request.getParameter("resumeText");

    // Check if resumeText is empty
    if (resumeText == null || resumeText.trim().isEmpty()) {
        printWriter.println("<html><body><h2>Please enter your
resume.</h2></body></html>");
    } else {
        try {
            // Define required skills and their scores
            Map<String, Integer> requiredSkills = new HashMap<>();
            requiredSkills.put("java", 5);
            requiredSkills.put("php", 3);
            requiredSkills.put("sql", 2);
```

```
        requiredSkills.put("python", 4);
        requiredSkills.put("workshop", 2);
        requiredSkills.put("seminar", 2);

        // Add more required skills as needed

        // Analyze resume text and find the skill with the highest score
        Map<String, Integer> skillScores =
analyzeResumeForSkills(resumeText.toLowerCase(), requiredSkills);

        // Calculate total score and display weighted percentages
        int totalScore = 0;
        for (int score : skillScores.values()) {
            totalScore += score;
        }

        printWriter.println("<html><body><h2>Skills Weightage:</h2>");
        for (Map.Entry<String, Integer> entry : skillScores.entrySet()) {
            double percentage = (double) entry.getValue() / totalScore * 100;
            printWriter.println("<p>" + entry.getKey() + ": " + Math.round(percentage) +
"%</p>");
        }
```

```
            double overallPercentage = (double) totalScore / (requiredSkills.size() * 5) * 100;

            printWriter.println("<p>Overall Profile Weightage: " +
Math.round(overallPercentage) + "%</p>");


            // Determine recommended career based on highest weighted percentage

            String recommendedCareer = determineRecommendedCareer(skillScores);


            // Display recommended career with weighted percentage



        } catch (Exception e) {

            printWriter.println("<html><body><h2>Error processing the
resume.</h2></body></html>");

            e.printStackTrace();

        }


    }

%>

<%-- Method to analyze the resume text and calculate scores for each skill --%>

<%!

    private Map<String, Integer> analyzeResumeForSkills(String resumeText, Map<String,
Integer> requiredSkills) {

        Map<String, Integer> skillScores = new HashMap<>();

        for (String skill : requiredSkills.keySet()) {
```

```
            int occurrences = countOccurrences(resumeText, skill);
            int score = occurrences * requiredSkills.get(skill);
            skillScores.put(skill, score);
        }
        return skillScores;
    }
%>
<%-- Method to count occurrences of a substring in a string --%>
<%!
    private int countOccurrences(String text, String searchString) {
        int count = 0;
        int lastIndex = 0;
        while ((lastIndex = text.indexOf(searchString, lastIndex)) != -1) {
            count++;
            lastIndex += searchString.length();
        }
        return count;
    }
%>

<%-- Method to determine the recommended career based on highest weighted percentage --
%>
<%!
```

```
    private String determineRecommendedCareer(Map<String, Integer> skillScores) {
        int maxScore = 0;
        String recommendedCareer = "";
        for (Map.Entry<String, Integer> entry : skillScores.entrySet()) {
            if (entry.getValue() > maxScore) {
                maxScore = entry.getValue();
                recommendedCareer = entry.getKey();
            }
        }
        return recommendedCareer;
    }
%>

<a href="Home.jsp">Back</a>
</body>
</html>
```

Upload Text Of The Profile

```
<%@ page import="java.io.IOException" %>
<%@ page import="java.io.PrintWriter" %>
<%@ page import="java.util.HashMap" %>
<%@ page import="java.util.Map" %>
<%@ page import="org.apache.poi.openxml4j.exceptions.InvalidFormatException" %>
<%@ page import="org.apache.poi.xwpf.usermodel.XWPFDocument" %>
```

```
<%@ page import="org.apache.poi.xwpf.extractor.XWPFWordExtractor" %>
<%@ page contentType="text/html;charset=UTF-8" language="java" %>
<%@ page import="java.io.*" %>
<%@ page import="java.sql.*" %>
<%@ page import="java.util.UUID" %>
<%@ page import="java.io.*" %>
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Career Suggestion</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-image: url(./image/2.gif); /* Add your GIF URL */
            background-size: cover; /* Cover the whole page */
            background-position: center; /* Center the background image */
            margin: 0;
            padding: 0;
        }
        h2 {
            color: #333;
        }
```

```
form {
    background-color: #fff;
    padding: 20px;
    margin: 20px auto;
    width: 50%;
    border-radius: 5px;
    box-shadow: 0 2px 5px rgba(0,0,0,0.1);
}
textarea {
    width: 100%;
    padding: 10px;
    margin-bottom: 10px;
    border: 1px solid #ccc;
    border-radius: 5px;
}
input[type="submit"] {
    padding: 10px 20px;
    background-color: #007bff;
    color: #fff;
    border: none;
    border-radius: 5px;
    cursor: pointer;
}
```

```
        input[type="submit"]:hover {
            background-color: #0056b3;
        }
        .result-container {
            background-color: #fff;
            padding: 20px;
            margin: 20px auto;
            width: 50%;
            border-radius: 5px;
            box-shadow: 0 2px 5px rgba(0,0,0,0.1);
        }
        p {
            color: #333;
            margin-bottom: 10px;
        }

    </style>
</head>
<body>

<h2>Enter Your Resume</h2>
<form method="post">
    <textarea name="resumeText" rows="10" cols="50"></textarea><br>
```

```
    <input type="submit" value="Analyze Resume">


</form>


<%
    response.setContentType("text/html");
    PrintWriter printWriter = response.getWriter();

    // Retrieve resume text from the form
    String resumeText = request.getParameter("resumeText");

    // Check if resumeText is empty
    if (resumeText == null || resumeText.trim().isEmpty()) {
        printWriter.println("<html><body><h2>Please enter your
resume.</h2></body></html>");
    } else {
            Connection conn = null;
        PreparedStatement pstmt = null;
        ResultSet rs = null;

        try {
            // Generate unique ID using UUID
            String id = UUID.randomUUID().toString();
```

```
            // Establish database connection
            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/recommend",
"root", "");

            // Prepare SQL statement to insert data into the database
            String sql = "INSERT INTO recommend_profile (id, resume_text) VALUES (?, ?)";
            pstmt = conn.prepareStatement(sql);
            pstmt.setString(1, id);
            pstmt.setString(2, resumeText);
            pstmt.executeUpdate();

            // Continue with resume analysis and display
            // Your existing code for resume analysis and display goes here...
            // I'll add a placeholder here for your existing code:

            // For demonstration purposes, let's print the ID and resume text
            printWriter.println("<h2>Stored Resume:</h2>");
            printWriter.println("<p>ID: " + id + "</p>");
            printWriter.println("<p>Resume Text: " + resumeText + "</p>");

        } catch (Exception e) {
```

```
            printWriter.println("<html><body><h2></h2></body></html>");
            e.printStackTrace();
        } finally {
            // Close resources
            if (rs != null) rs.close();
            if (pstmt != null) pstmt.close();
            if (conn != null) conn.close();
        }
        try {
            // Define required skills and their scores
            Map<String, Integer> requiredSkills = new HashMap<>();
            requiredSkills.put("java", 5);
            requiredSkills.put("php", 3);
            requiredSkills.put("sql", 2);
            requiredSkills.put("python", 4);
            requiredSkills.put("workshop", 2);
            requiredSkills.put("seminar", 2):
            Map<String, Integer> skillScores =
analyzeResumeForSkills(resumeText.toLowerCase(), requiredSkills);
            int totalScore = 0;
            for (int score : skillScores.values()) {
                totalScore += score;
            }
```

```
        double overallPercentage = (double) totalScore / (requiredSkills.size() * 5) * 100;

        printWriter.println("<html><body><h2>Overall Weightage of the
Profile</h2></body></html>");

        printWriter.println("<p>Overall Profile Weightage: " +
Math.round(overallPercentage) + "%</p>");

        String recommendedCareer = determineRecommendedCareer(skillScores);

        printWriter.println("<h2>Recommended Career:</h2>");

        if (recommendedCareer.equals("java")) {

            printWriter.println("<p>" + recommendedCareer + "</p>");

            printWriter.println("<p>You Are Wellverse In Java So You Approach The
Companies Using Java As A Language. Or Else You Can Go As A Java Teacher For
Colleges.</p>"+ Math.round(overallPercentage) + "% Weightage</p>");

        } else if (recommendedCareer.equals("php")) {

            printWriter.println("<p>" + recommendedCareer + "</p>");

            printWriter.println("<p>You Are Wellverse In PHP So You Approach The
Companies Using PHP As A Language. Or Else You Can Go As A PHP Teacher For
Colleges.</p>"+ Math.round(overallPercentage) + "% Weightage</p>");

        } else if (recommendedCareer.equals("sql")) {

            printWriter.println("<p>" + recommendedCareer + "</p>");

            printWriter.println("<p>You Are Wellverse In Backend So You Approach The
Companies As A Backend Developer.</p>"+ Math.round(overallPercentage) + "%
Weightage</p>");

        } else if (recommendedCareer.equals("python")) {

            printWriter.println("<p>" + recommendedCareer + "</p>");
```

```
            printWriter.println("<p>You Are Wellverse In Python So You Approach The Companies Using Python As A Fundamental Language. Or Else You Can Go As A Python Teacher For Colleges.</p>"+ Math.round(overallPercentage) + "% Weightage</p>");

        } else if (recommendedCareer.equals("workshop")) {

            printWriter.println("<p>" + recommendedCareer + "</p>");

            printWriter.println("<p>You Have Attended Many Workshops So You Can Go To Colleges To Conduct And Organize Workshops.</p>"+ Math.round(overallPercentage) + "% Weightage</p>");

        } else if (recommendedCareer.equals("seminar")) {

            printWriter.println("<p>" + recommendedCareer + "</p>");

            printWriter.println("<p>You Have Gained Much Knowledge Through Attending Seminars So You Can Go As A Seminar Conductor For Colleges And Schools.</p>"+ Math.round(overallPercentage) + "% Weightage</p>");

        } else {

            printWriter.println("<p>No suitable career found based on the resume.</p>");

        }

        printWriter.println("</body></html>");

    } catch (Exception e) {

        printWriter.println("<html><body><h2>Error processing the resume.</h2></body></html>");

        e.printStackTrace();

    }

  }

%>

<%!
```

```
    private Map<String, Integer> analyzeResumeForSkills(String resumeText, Map<String,
Integer> requiredSkills) {
        Map<String, Integer> skillScores = new HashMap<>();
        for (String skill : requiredSkills.keySet()) {
            int occurrences = countOccurrences(resumeText, skill);
            int score = occurrences * requiredSkills.get(skill);
            skillScores.put(skill, score);
        }

        return skillScores;
    }
%>
<%!
    private int countOccurrences(String text, String searchString) {
        int count = 0;
        int lastIndex = 0;
        while ((lastIndex = text.indexOf(searchString, lastIndex)) != -1) {
            count++;
            lastIndex += searchString.length();
        }
        return count;
    }
%>
```

```
<%!
    private String determineRecommendedCareer(Map<String, Integer> skillScores) {
        int maxScore = 0;
        String recommendedCareer = "";
        for (Map.Entry<String, Integer> entry : skillScores.entrySet()) {
            if (entry.getValue() > maxScore) {
                maxScore = entry.getValue();
                recommendedCareer = entry.getKey();
            }
        }
        return recommendedCareer;
    }
%>
<a href="Home.jsp">Back</a>
</body>
</html>
```

Dependencies

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">
<modelVersion>4.0.0</modelVersion>
<groupId>Career</groupId>
<artifactId>Career</artifactId>
```

<version>0.0.1-SNAPSHOT</version>

<packaging>war</packaging>

<build>

<plugins>

<plugin>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.8.1</version>

<configuration>

<release>20</release>

</configuration>

</plugin>

<plugin>

<artifactId>maven-war-plugin</artifactId>

<version>3.2.3</version>

</plugin>

</plugins>

</build>

<dependencies>

<dependency>

<groupId>javax.servlet</groupId>

<artifactId>javax.servlet-api</artifactId>

<version>3.1.0</version>

<scope>provided</scope>

```
</dependency>
<dependency>
<groupId>javax.servlet.jsp</groupId>
<artifactId>javax.servlet.jsp-api</artifactId>
<version>2.3.1</version>
<scope>provided</scope>
</dependency>
<dependency>
<groupId>com.indoorvivants.cloudflare</groupId>
<artifactId>worker-types_sjs1_3</artifactId>
<version>3.3.0</version>
</dependency>
<dependency>
<groupId>mysql</groupId>
<artifactId>mysql-connector-java</artifactId>
<version>8.0.33</version>
</dependency>
<dependency>
<groupId>nz.ac.waikato.cms.weka</groupId>
<artifactId>weka-dev</artifactId>
<version>3.9.6</version>
</dependency>
<dependency>
```

<groupId>org.apache.commons</groupId>

<artifactId>commons-math3</artifactId>

<version>3.6.1</version>

</dependency>

<dependency>

<groupId>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

<version>8.0.28</version> <!-- Or the latest version -->

</dependency>

<dependency>

<groupId>org.apache.poi</groupId>

<artifactId>poi</artifactId>

<version>5.2.4</version> <!-- Or the latest version -->

</dependency>

<dependency>

<groupId>org.apache.poi</groupId>

<artifactId>poi-ooxml</artifactId>

<version>5.2.4</version> <!-- Or the latest version -->

</dependency>

<dependency>

<groupId>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

<version>8.0.27</version> <!-- Or the latest version available -->

```
</dependency>
<!-- Servlet API (provided by the servlet container) -->
<dependency>
<groupId>javax.servlet</groupId>
<artifactId>javax.servlet-api</artifactId>
<version>4.0.1</version>
<scope>provided</scope>
</dependency>
<!-- Apache POI for handling Word documents -->
<dependency>
<groupId>org.apache.poi</groupId>
<artifactId>poi</artifactId>
<version>5.2.3</version>
</dependency>
<dependency>
<groupId>org.apache.poi</groupId>
<artifactId>poi-ooxml</artifactId>
<version>5.2.3</version>
</dependency>
<dependency>
<groupId>org.apache.poi</groupId>
<artifactId>poi-ooxml-schemas</artifactId>
<version>4.1.2</version>
```

```
</dependency>
<!-- Apache Commons FileUpload for handling file uploads -->
<dependency>
<groupId>commons-fileupload</groupId>
<artifactId>commons-fileupload</artifactId>
<version>1.4</version>
</dependency>
<!-- Apache POI -->
<dependency>
<groupId>org.apache.poi</groupId>
<artifactId>poi</artifactId>
<version>5.2.2</version> <!-- Make sure to use the latest version -->
</dependency>
<dependency>
<groupId>org.apache.poi</groupId>
<artifactId>poi-ooxml</artifactId>
<version>5.2.2</version> <!-- Make sure to use the latest version -->
</dependency>
<!-- Apache PDFBox -->
<dependency>
<groupId>org.apache.pdfbox</groupId>
<artifactId>pdfbox</artifactId>
<version>2.0.24</version> <!-- Make sure to use the latest version -->
```

</dependency>

<dependency>

<groupId>commons-fileupload</groupId>

<artifactId>commons-fileupload</artifactId>

<version>1.4</version>

</dependency>

<dependency>

<groupId>org.apache.opennlp</groupId>

<artifactId>opennlp-tools</artifactId>

<version>1.9.3</version>

</dependency>

<dependency>

<groupId>commons-io</groupId>

<artifactId>commons-io</artifactId>

<version>2.15.1</version>

</dependency>

<dependency>

<groupId>org.apache.poi</groupId>

<artifactId>poi</artifactId>

<version>4.1.2</version> <!-- Or the latest version available -->

</dependency>

<dependency>

<groupId>org.apache.poi</groupId>

<artifactId>poi-ooxml</artifactId>

<version>4.1.2</version> <!-- Or the latest version available -->

</dependency>

<dependency>

<groupId>org.apache.opennlp</groupId>

<artifactId>opennlp-tools</artifactId>

<version>1.9.3</version>

</dependency>

<!-- Jsoup for web scraping -->

<dependency>

<groupId>org.jsoup</groupId>

<artifactId>jsoup</artifactId>

<version>1.14.3</version>

</dependency>

<!-- JavaMail API for email processing -->

<dependency>

<groupId>com.sun.mail</groupId>

<artifactId>javax.mail</artifactId>

<version>1.6.2</version>

</dependency>

</dependencies>

</project>