##### *“RecruitUS”*

****

**MINOR PROJECT REPORT**

***Submitted to***

**RAJIV GANDHI TECHNICAL UNIVERSITY Bhopal (M.P)**

***In partial fulfillment for the award of the degree***

***Of Bachelor of Engineering***

***In***

***Computer Science & Engineering***

***Session 2018-19***

**Supervised By: Submitted By:**

Ms. Namrata Sharma Anshumaan S. Thakur

Anjani S. Thakur

**Department of Computer Science & Engineering**

**Sushila Devi Bansal College Of Engineering Indore**

<www.sdbce.ac.in>

**CERTIFICATE**

This is to certify that the Minor Project Report entitled “**RecruitUs**” submitted by **Anshumaan Singh Thakur** and **Anjani Singh Thakur** is approved for the award of Degree of Bachelor of Engineering in Computer Science & Engineering.

**INTERNAL EXAMINER EXTERNAL EXAMINER**

**DATE: DATE:**

**HEAD OF DEPARTMENT**

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**RECOMMENDATION**

The Minor Project entitled “**RecruitUs**” Submitted by **Anshumaan Singh Thakur** and **Anjani Singh Thakur** is a satisfactory account of the bona fide work done under our supervision is recommended towards partial fulfillment for the award of the Bachelor of Engineering (Computer Science & Engineering) degree by Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal.

**Project Coordinator:**

Ms. Namrata Sharma

**Endorsed by:**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

HOD (CS) Director

Ms. Varsha Zokarkar Dr. Ruchi vijaywargiya

SHUSHILA DEVI BANSAL COLLEGE OF ENGINEERING, INDORE

<(www.sdbce.ac.in)>

**Table of Contents**

1. **Introduction**
   1. Purpose
   2. Scope
   3. Definition
   4. Platform Specification

1.4.1 Hardware

1.4.2 Software

1.4.3 Implementation Language

**2. Requirement Gathering and Analysis**

2.1System Analysis

2.1.1 Functional and Non-functional Requirements

2.1.2 Existing vs. Proposed System

2.2 Software Engineering Paradigm Applied

2.2.1 Description of process Model

2.2.2 Advantages & Disadvantages

2.2.3 Why Used For Project

2.3 Requirement Analysis

2.3.1 Data flow diagram

2.3.2 Identify Actor

2.3.3 Identify Use Cases

2.3.4 Use Case Diagram

2.3.5 Sequence Diagram

2.3.6 Class Diagram

2.3.7 Activity diagram

**3.**  **Design**

3.1 Design Classes

3.2 ER Diagram

3.3 Database tables

3.4 Snap shots

**4. Conclusion**

**5. Future Scope**

**6. References**

1. **Introduction**
   1. **Purpose:**

This purpose of the project is to assist the recruiters to select the appropriate candidates who have applied for a particular post. It is a recruitment website (RecruitUs) where startups or companies can register and start grading the candidates. The candidates with higher grades will appear at the top of list and HRs can send joining mails to them. They can also discard the inept candidates.

It is often observed that during the recruitment process, the recruiters or HRs have to do manual pen-paper evaluation for the grading of the candidates who have applied for the job post. This process is very time consuming and mentally tiring to the recruiters. Therefore, to solve this problem this website is introduced, i.e., to replace the manual evaluation procedure.

**1.2 Scope:**

As this website can be used by any firm, institute or startup, thus it can be extensively used for mass recruitment and also for evaluating and ranking the candidates. It is proficiently designed using various tools of Java such as Java Mail API, servers, JSP, MySQL jar, designing languages such as HTML and CSS and scripting language JavaScript. By means of the provision of mail HR/CEO/Recruiter of the registered company can send the joining mails to the candidates.

**1.3 Definition:**

The project is a website named RecruitUs which is designed in order to stamp out the problem of manual pen paper evaluation procedure for recruiting candidates and it is designed using JAVA technology, MySQL, HTML, CSS and JavaScript. This website will be particularly helpful to the recruiters and the HRs of the various firms who want to recruit the candidates in multitude without the hectic manual paper-pen procedure for the recruitment.

* 1. **Platform Specification:**

**1.4.1 Hardware:** This project is made on the system which has the following hardware specifications:

Processor: Intel® Core™ i36006U CPU @ 2.00GHz Installed Memory (RAM): 8.00 GB (7.89 GB usable)

System Type: 64-bit Operating System, x64-based processor

**1.4.2 Software:** The software being used to make this website is NetBeans and has the following specifications:

Type: Application

File Version: 8.0.0.0

Product Name: NetBeans IDE 8.0

Product Version: 8.0.0.0

Copyright: © 2007, 2013 Oracle and/or its affiliates

Size: 1.58 MB

Language: Language Neutral

**1.4.3 Implemented Language(s):** The following languages have been used for various methodologies of this web project:

Java

MySQL

HTML and CSS

JavaScript

1. **Requirement Gathering and Analysis**
   1. **System Analysis:**

**2.1.1 Functional and Non-Functional Requirements:** The Functional and Non-Functional requirements of this software project are as follows:

**Functional Requirements:**

* Regulatory Requirements:

1. Company Details
2. Candidate Details

* External Interface Requirements:

1. Software Interfaces

* mysqlconnector.jar
* mail.jar
* JDK 1.8

1. Servers:

* Apache Tomcat or TomEE

1. User Interfaces:

* Web Browser – Mozilla Firefox
* Webpages made using HTML and CSS
* Security/Authentication Requirements:

1. OTP (One Time Password)

* Other Functional Requirements:

1. Logout
2. Create New Account
3. Deleting a Candidate
4. Sending Mail
5. Calculation of Grades

**Non-Functional Requirements:**

* Performance:

1. Maximum time taken by the server to start = 2501 ms
2. Maximum time taken by the project to run = 1000 ms
3. Response time depends upon internet speed, in some pages maximum response time is 1000 ms.

* Reliability
* Availability
* Usability
* Precision

**2.1.2 Existing Vs. Proposed System:**

**Existing System:**

The existing system of recruitment is quite tedious as is a manual paper-pen procedure. Most of the time HRs or recruiters, when they go to any college for recruitment, they have to do lot of paperwork to hire the candidates.

All the work done by the recruiters is manual including giving the marks to the candidate during the recruitment process. This creates a lot of problems for the recruiters and they also face difficulties doing all the manual work and giving marks by writing to the appearing candidates.

Also, there is a lot of hectic calculation which must be done during the process. This is also a big problem for the HRs and the recruiters. This is so time consuming and also very intimidating process.

So, for all of these problems we need to make a website on which all this manual work can be done online and also where we don’t really need to make calculations manual. This website would help a lot for the recruitment of the candidates.

**Proposed System:**

This website will help the recruiters to recruit students from any institution. It will provide them with the solution to the manual pen-paper procedure which includes and require a lot of hectic work.

It will require the company HRs or recruiters to login first. In case if they don’t have an ID on the website, they need to register first. After they register, they will be provided with their unique company ID and they can always login again with their company ID.

There will be separate databases for each company in which records will be maintained. The records will consist of the records of the candidates such as academics, communication skills, technical skills, etc. and the recruiters will be able to give marks to the candidates according to their abilities in various fields. After they will enter the marks of candidates, separate database will be arranged in ascending order, with less qualified candidates at the top.

There will be link through which HRs can also remove less qualified candidates. After maintaining the records as per their requirements they can send mails to candidates regarding joining date and training.

It will be most secure website as separate passwords or IDs are given to companies.

* 1. **Software Engineering Paradigm Applied:**

The Software Engineering Paradigm applied for this project is spiral model. The spiral model is used when we want the processes to be in repetition or iterative.

**2.2.1 Description of the Process model:**

The Spiral Process model includes the steps of Requirement Analysis, Planning, Design, Coding, Testing and Delivery. The motivation behind this process model was due to the fact that requirements given at the beginning of the project are incomplete where requirements analysis phase and the design phase are continuously evolving as the time passes. The Steps go through a cyclical motion as the requirements and design evolve.

The Spiral model is non-linear and cyclic. Each cycle of the spiral consists of 6 phases. The radius of spiral represents the cost accumulated so far in the process and the angular dimension represents the program in the process. Each phase starts with a design goal and ends with the client/customer reviewing the progress thus far.

The Spiral model focuses on constant re-evaluation of the design and the risks involved. It can be described as being ‘iterative’ and ‘sequential’. The model can accommodate any other process development model due to its generality. It is commonly used for large and complex projects which as a result, can also be described as being expensive.

**2.2.2 Advantages and Disadvantages:**

* Advantages:

1. Takes into consideration the change of requirements during the development process.
2. The model is flexible and can be tailored for a variety of situations, such as reuse, component based development and prototyping.
3. Can be adjusted to be used as any other process model.
4. Combines the best features of waterfall and prototyping models.
5. High amount of risk analysis hence the avoidance of risk is enhanced.
6. Additional functionality can be added at a later date.
7. Software is produced early in the Software life cycle.
8. Project estimates in terms of schedule, costs etc. become more and more realistic as the project moves forward and the loops in the spiral gets completed.

* Disadvantages:

1. Can be a costly model to use.
2. Risk analysis requires highly specific expertise.
3. Project’s success is highly dependent on the risk analysis phase.
4. Doesn’t work well for smaller projects.
5. It is not suitable for low risk projects.
6. May be hard to define objectives, verifiable milestones.
7. Spiral may continue indefinitely.

**2.2.3 Why Spiral model is used for this project:**

The reason for using the spiral model of project development is that initially the requirements were not complete and frequent. For instance, initially there was no plant to add the E-mail facility in the project but as the project developed, the requirement of E-mail facility was felt.

Changes were consistent and were required to be applied at any time in the development process of the project. Also, initially the requirements were unclear and complex and the releases of the project were frequent.

There was always a need to add the new functionality during each phase of the project and this was a medium risk project as it was required to be completed in a short time and long term commitment was not feasible in this project.

* 1. **Requirement Analysis:**

**2.3.1 Data Flow Diagram:**

**2.3.2 Identification of Actors:**

The Actors in this project are the Database Admin/Web Master and the Company.

**2.3.3 Identification of Use-Cases:**

The following Use-Cases are identified as per the requirement analysis:

* Login:

Login action is done by Company.

* Register:

Registration is done by the Company.

* Evaluate the candidates:

Evaluation is done by the Company (HR).

* Send mail to the candidates:

Company HR does the action of sending joining mails to the Candidates.

* Delete the candidates:

Company (HR) deletes the ineligible Candidates.

* Maintain the servers:

Servers are maintained by the Database Admin/Web Master

* Update the profile:

Company does the action of updating profile.

* View All Candidates:

Company (HR) can view all the candidates.

* Update the password:

Company can update its password.

* Generate and Send OTP:

Database Admin generates and sends the OTP.

* Logout:

This action is performed by the company.

**2.3.4 Use-Case Diagram:**



**2.3.5 Sequence Diagram**

**Sequence Diagram for Login and Registration**

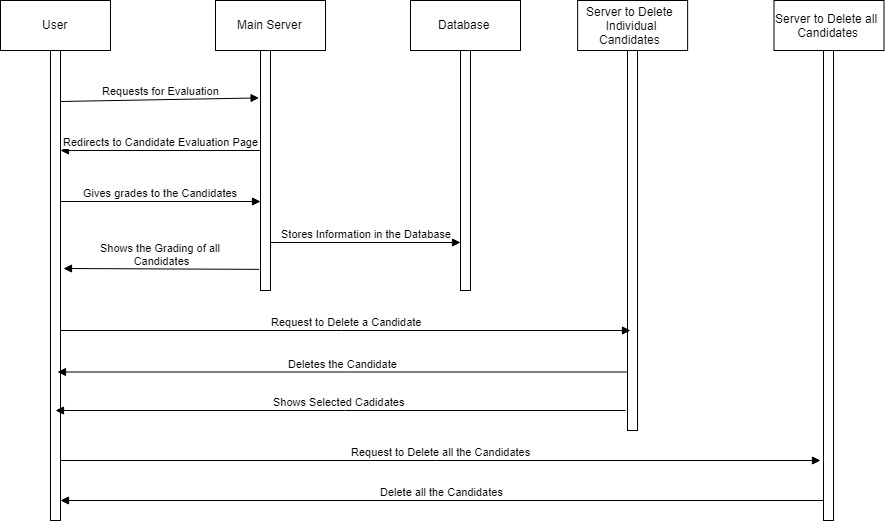
****

**Explanation:**

As the user requests to register on the website, the Registration Server redirects the User to the registration page where user fills all the details in the registration form. The registration Server Stores all the details in the database.

When the User fill login form and requests for login on the website, the Registration Server matches the details to the database. If the details are correct if redirects the User to User’s profile page.

**Sequence Diagram for Evaluation and Deletion of Candidates**



**Explanation:**

When the User requests for the evaluation of candidates, the Main Server redirects the User to Candidate Evaluation Page where User gives grades to the candidates. The Main Server stores all this information in the database and shows the grading of all the candidates to the User.

If the User requests the deletion of particular candidate, the Server to Delete Individual Candidates deletes that candidate and shows the remaining candidates. If the User requests to delete all the candidates, Server to delete all the Candidates deletes all the candidates.

**Sequence Diagram for Sending Mail to the Candidates**

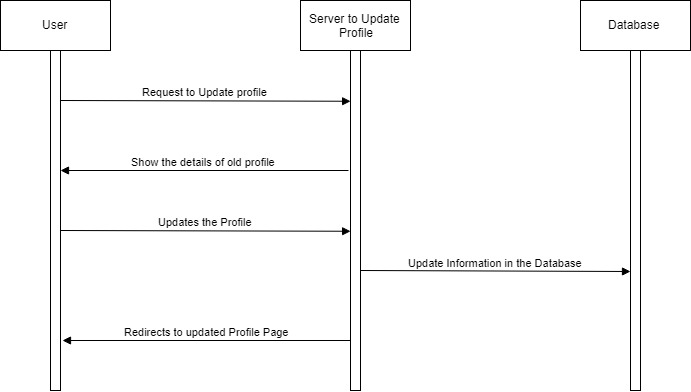


**Explanation:**

When the User requests to send mail to all the candidates, the Server to send Mail to all the Candidates fetches the e-mail IDs of the candidates from the database and sends mail to all the candidates and redirects the User to view candidate page.

When the user requests to send mail to a particular candidate, another server (Server to Send Mail to Individual Candidates) matches the e-mail ID that the User has entered to the database. If the ID is correct it sends joining mail to that particular candidate and redirects the user to view candidate page.

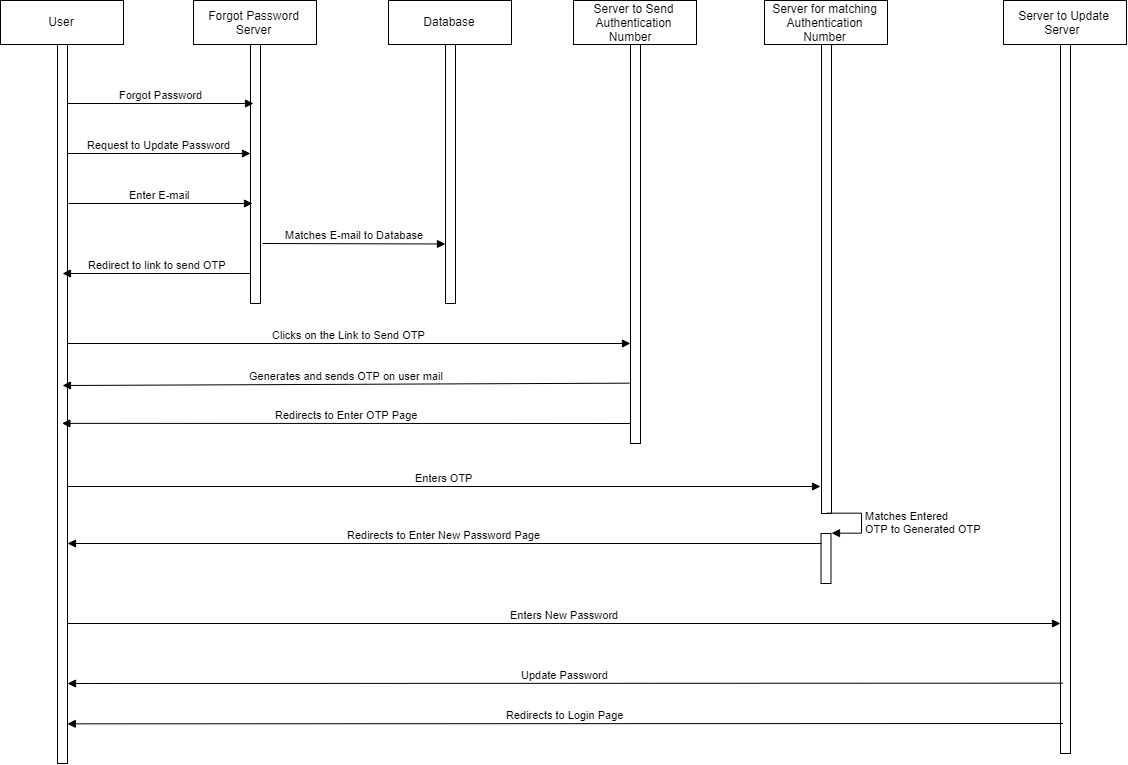
**Sequence Diagram to Update Profile**



**Explanation:**

When the User requests to update the profile, the Update Profile Server shows the details of old profile to the user. The User updates the profile and the Server updates the information in the database and redirects the user to the updated profile page.

**Sequence Diagram to Update Password**

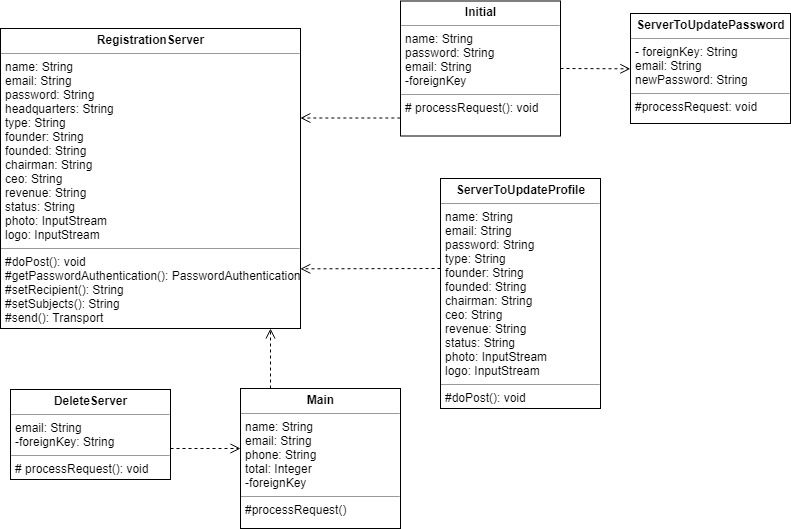


**Explanation:**

When the user forgets password, he/she requests to update the password. The Forgot Password server matches the e-mail of the user to the database and if the mail is correct the, the server redirects the User to the link to send OTP. As the User clicks on the link the Server to send Authentication number generates and send OTP on the User’s mail and redirects to Enter OTP page. The User enter the OTP and Server to match Authentication number checks whether OTP is correct or not. If it is correct, then it redirects the user to Enter new Password page where User enters the new password and password will be updated by Update Password Server.

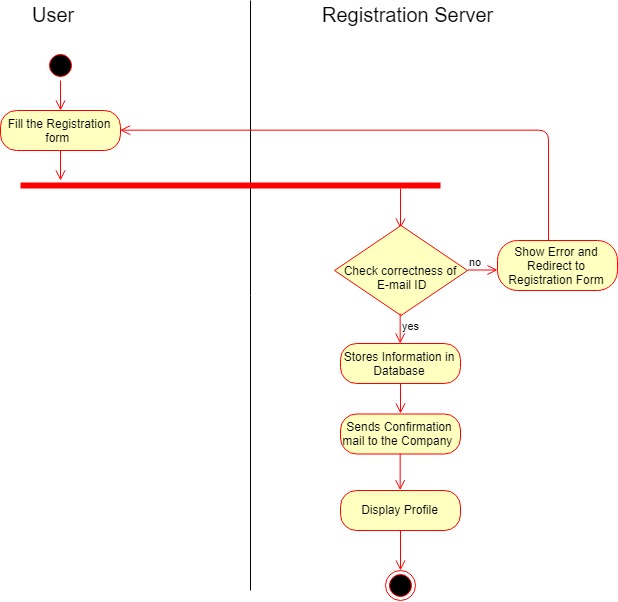
* + 1. **Class Diagram**

It is used to show the existence of classes and their relationship in the logical view of the system.

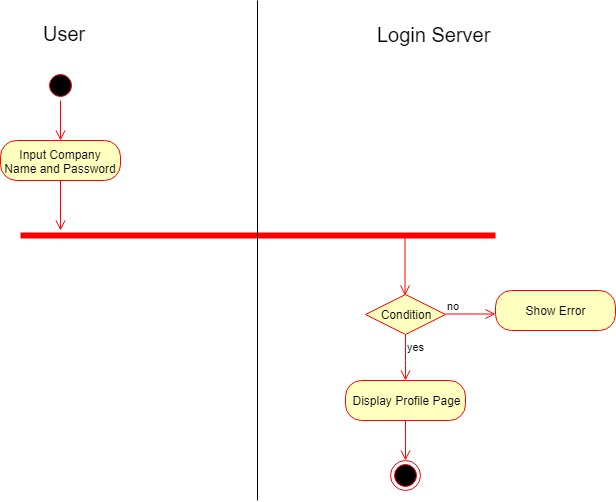


* + 1. **Activity Diagram**

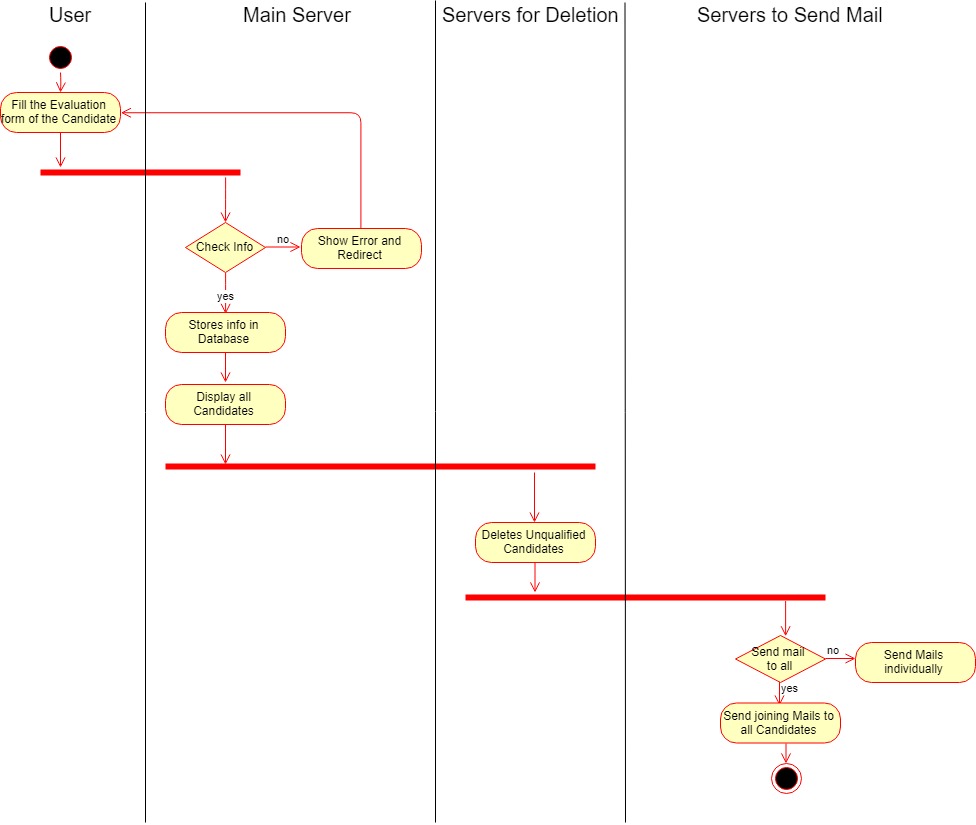
**Activity Diagram for Registration**



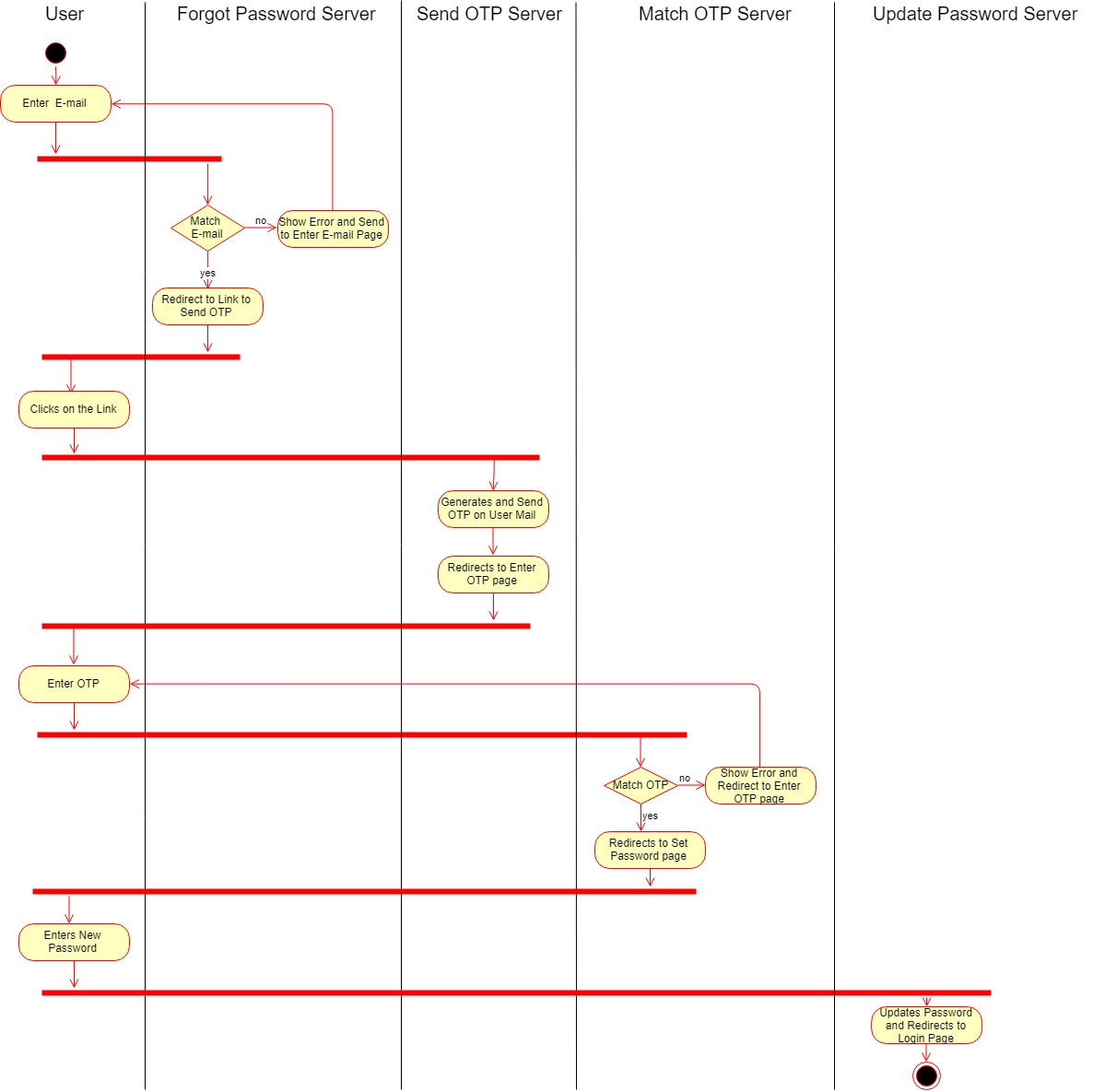
**Activity Diagram for Login**

****

**Activity Diagram for Evaluation**

****

**Activity Diagram for Updating Password**



1. **Design**

**3.1 Design Classes:**

**Registration:**

public class RegistrationServer extends HttpServlet {

private Connection con, PreparedStatement ps, ResultSet rs, Statement st;

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

String name, email, password, headquarters, type, founded, founder, chairman, ceo, revenue, status;

int lastserialno=0;

int lastotp=0;

final String from="recruitus.website@gmail.com";

final String pass="############”;

Session s=Session.getInstance(p,

new javax.mail.Authenticator(){

protected PasswordAuthentication getPasswordAuthentication()

{

return new PasswordAuthentication(username,pass);

}

}

);

try{

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

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

st=con.createStatement();

rs =st.executeQuery("select max(serialno) from allcompanies");

while(rs.next())

{

lastserialno=rs.getInt(1);

}

int newserialno=lastserialno+1;

ps=con.prepareStatement("insert into allcompanies (serialno ,name ,email, password, location, type, founded, founder, chairman, ceo, revenue, photo, status, logo) values(?,?,?,?,?,?,?,?,?,?,?,?,?,?)"; ps.setInt(1, newserialno);

ps.setString(2, name);

ps.setString(3, email);

ps.setString(4, password);

ps.setString(5, headquarters);

ps.setString(6, type);

ps.setString(7,founded);

ps.setString(8,founder);

ps.setString(9, chairman);

ps.setString(10, ceo);

ps.setString(11, revenue);

ps.setBlob(12, photo);

ps.setString(13, status);

ps.setBlob(14, logo);

int a=ps.executeUpdate();

if(a>0)

{

//send mail to company’s mail and redirect to profile page

}

}

catch(ClassNotFoundException | SQLException | MessagingException ex) {

//redirect error to registration page

}

}

}

**Login:**

public class Initial extends HttpServlet {

private Connection con, ResultSet rs, Statement st;

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

String password, name;

int c1=0,c2=0;

int foreignkey;

String email;

String s3 = null;

try{

//connection and driver loading

String q1="select \* from allcompanies";

st=con.createStatement();

rs=st.executeQuery(q1);

while(rs.next())

{

String s1;

s1 =rs.getString(4);

String s2;

s2 = (rs.getString(2).toLowerCase());

if(password.equals(s1))

{

c1++;

s3=s1;

}

if(name.equals(s2))

{

c2++;

s3=s1;

}

}

if(c1==0||c2==0){

//redirect error to login page

}

else

{

String q2="select serialno from allcompanies where password='"+password+"'and name='"+name+"'";

st=con.createStatement();

rs=st.executeQuery(q2);

while(rs.next())

{

//redirect to profile page along with foreign key

}

}

}

catch(ClassNotFoundException | SQLException ex){

//error }

}

}

}

**Evaluate Candidates:**

public class Main extends HttpServlet {

private Connection con, PreparedStatement p;

private int foreignkey;

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

String name ,email, phone, ts, cs, ms, acad, remarks;

int c=0;

if(Integer.parseInt(ts)>10)

{

c++;

//error marks should be less than 10

}

if(Integer.parseInt(ms)>10)

{

c++; //error marks should be less than 10

}

if(Integer.parseInt(cs)>10)

{

c++; //error marks should be less than 10

}

if(Integer.parseInt(acad)>10)

{

c++; //error marks should be less than 10

}

try{

tm=Integer.parseInt(ts)+Integer.parseInt(cs)+Integer.parseInt(ms)+Integer.parseInt(acad);

}

catch(NumberFormatException e)

{

c++;//error enter marks

}

try{

if(c==0){

// connection and driver loading

ps=con.prepareStatement("insert into table2(email, name, phone, techsk, mansk, commsk, acad, remarks, total,foreignkey)values(?,?,?,?,?,?,?,?,?,?)");

ps.setString(1, email);

ps.setString(2, name);

ps.setString(3, phone);

ps.setString(4, ts);

ps.setString(5, ms);

ps.setString(6, cs);

ps.setString(7, acad);

ps .setString(8, remarks);

ps.setString(9, Integer.toString(tm));

ps.setInt(10, foreignkey);

int a=ps.executeUpdate();

if(a>0)

{

//redirect to show all candidates page

}

}}

catch(ClassNotFoundException | SQLException ex){

{

//error candidate already exist } }

}

**Send Joining Mail:**

public class SendMail extends HttpServlet {

String name;

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

String to,name;

Private int foreignkey;

final String from="recruitus.website@gmail.com";

final String password="############";

Session s=Session.getInstance(p,

new javax.mail.Authenticator(){

protected PasswordAuthentication getPasswordAuthentication() {

return new PasswordAuthentication(username,password);

}

} );

try{

//send joining mail

}

catch(MessagingException e){

if(to==""){

//error enter email

}

else{

//error email doesn’t exist

}

} }

**Update Password:**

public class UpdatePassword extends HttpServlet {

private Connection con;

private Statement st;

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

HttpSession session=request.getSession(true);

String new newpassword, email=null;

int foreignkey;;

try{

//Connection and Driver Loading

st=con.createStatement();

int i=st.executeUpdate("update allcompanies set password='"+newpassword+"' where serialno='"+foreignkey+"' and email='"+email+"'");

if(i>0)

{

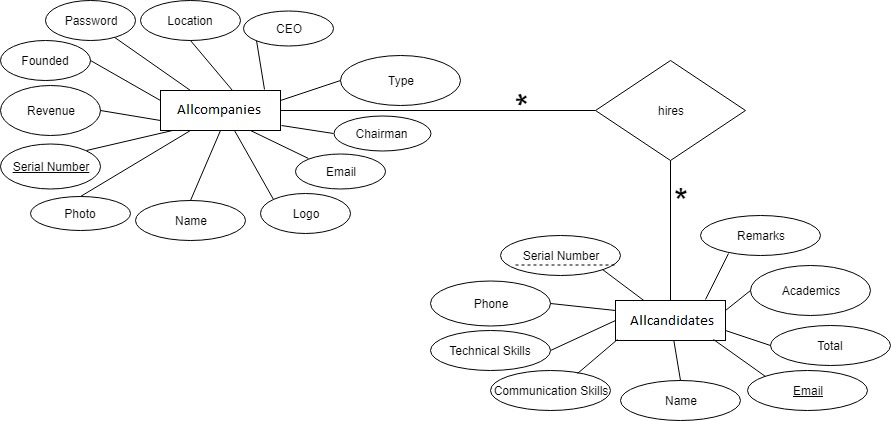
//update password in database and redirect to login page

}

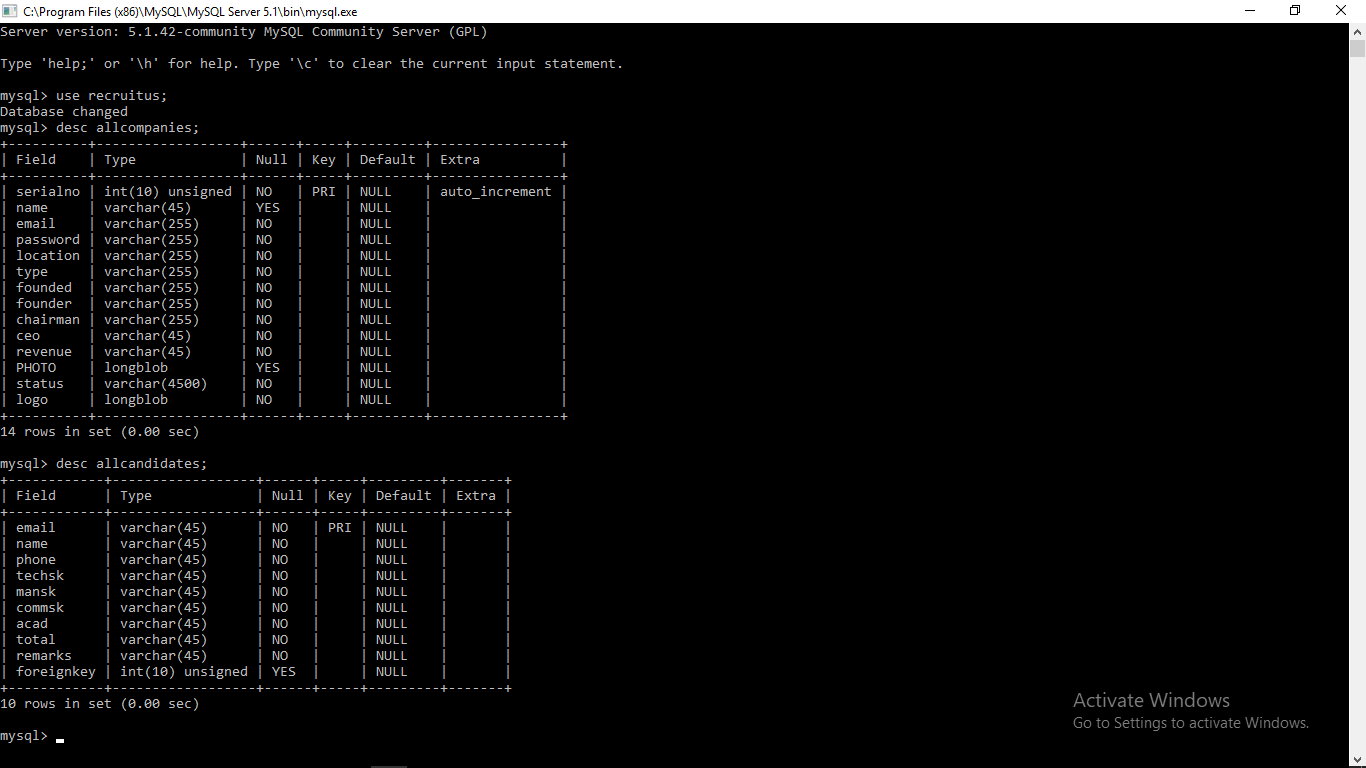
else{ //error }

catch(Exception ex) {//error} }}

**3.2 ER-Diagram**

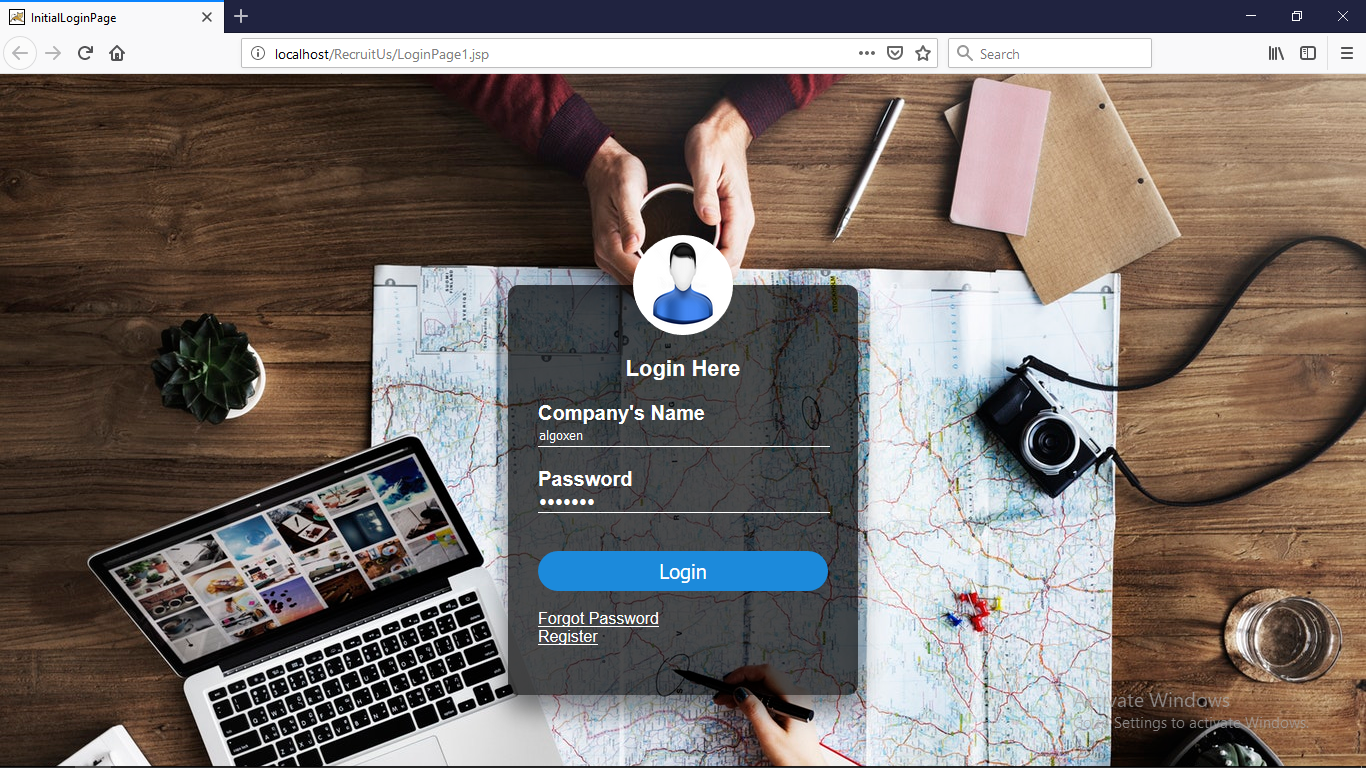
****

* 1. **Database tables:**

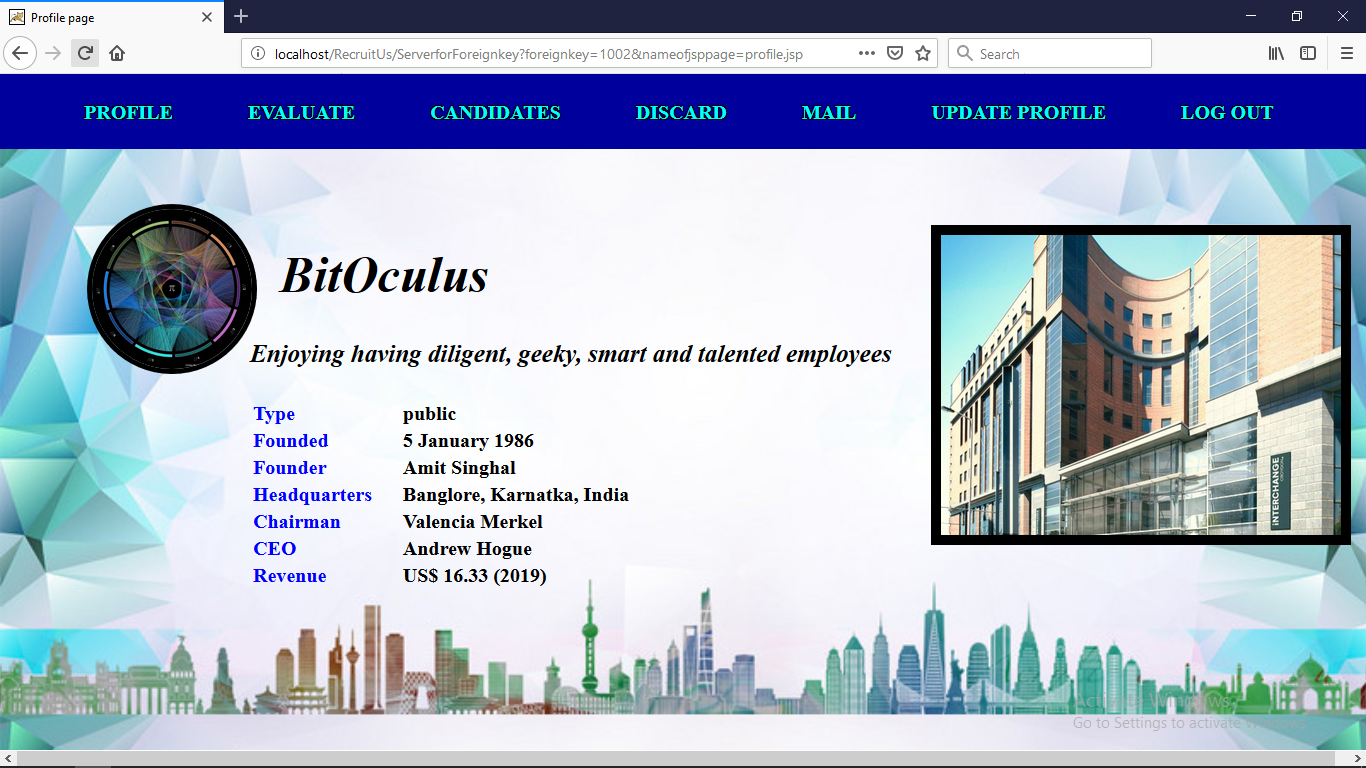


**3.4 Snapshots:**

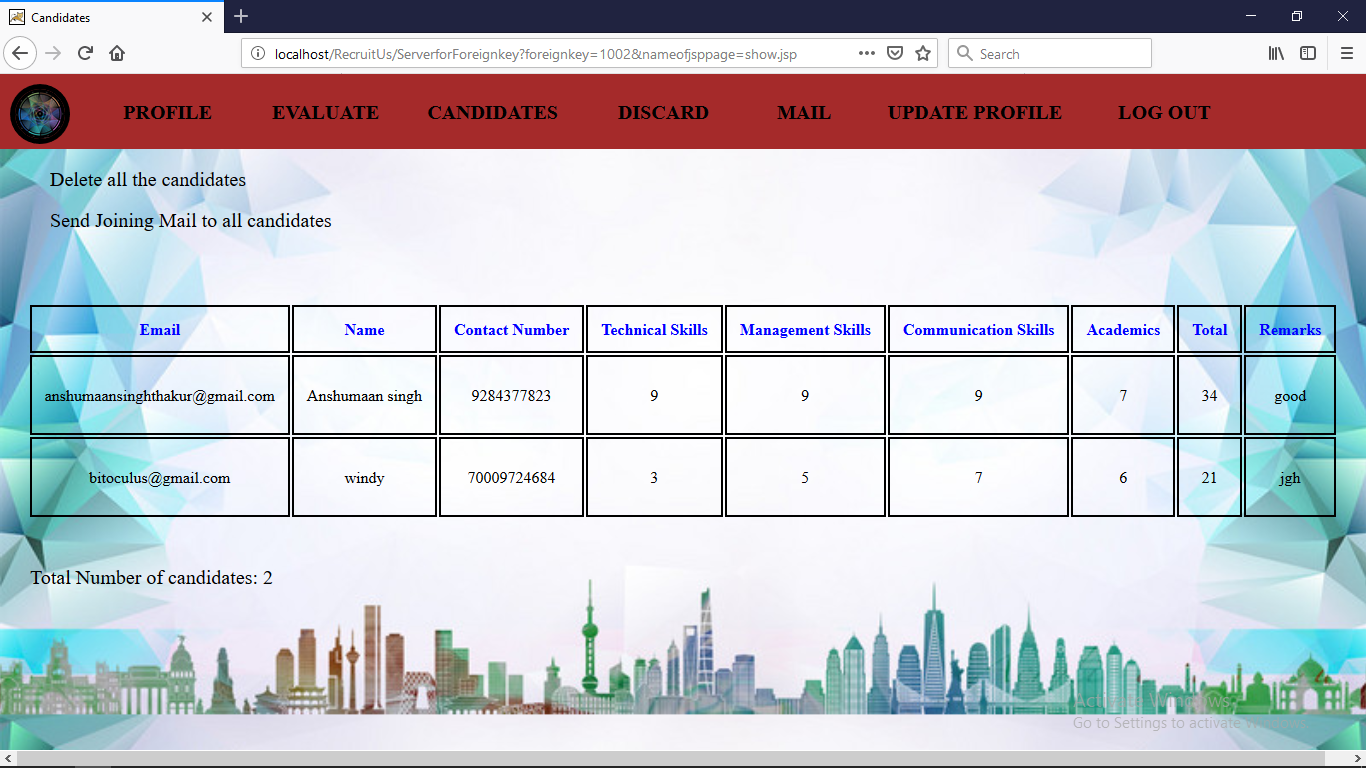
**Login page**



**Profile page**



**Show all candidates**

****

1. **Conclusion**

In this report, specifying the web project named RecruitUs, the details of the project have been provided. This website is particularly made for the HRs and the Recruiters of the various companies or the institutions so that they don’t have to do the normal paper-pen procedure for the candidate evaluation process during the recruitment. This website has some noteworthy features such as automatic grade calculation, mail sending facility and also forgot password functionality.

RecruitUs is so designed that any person can use it easing locanse of its simplicity. It is equipped with excellent GUI for good user experience.

1. **Future Scope**

In future, the project will be advanced in which companies HRs, directors will interact with each other. They will be able to see number of employees in other companies. In advanced version of project, they can also maintain records of employees and store information regarding services or products of the company. Also they can interact with each other on advanced website.

1. **References**
2. [**www.wikipedia.org**](http://www.wikipedia.org)
3. [**www.geeksforgeeks.org**](http://www.geeksforgeeks.org)
4. [**www.tutorialspoint.com**](http://www.tutorialspoint.com)