B. Tech. project Grace Marks Allocator– G 22

PROJECT REPORT-I

Submitted by

S. No	Name	Roll Number
1.	G Lokesh	CB.EN.U4CSE18117
2.	P Pruthve Rej	CB.EN.U4CSE18136
3.	N Pavan Kruthik	CB.EN.U4CSE18137
4.	P Upendra	CB.EN.U4CSE18138
5.	Sk Awez	CB.EN.U4CSE18148

is fulfilment of the requirements for the Course – 15CSE376 (Net Centric Programming)

BACHELOR OF TECHNOLOGY IN

COMPUTER SCIENCE AND ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

AMRITA SCHOOL OF ENGINEERING

AMRITA VISHWA VIDYAPEETHAM

COIMBATORE -641112

August 2021

Table of Contents

Abstract

- Every year the tracking of paper publications that provide students with grace marks and finding the right subject to which the marks can be added in order to increase the grade of the student in the best way possible is a very manually hectic job
- We have designed a software that eases the tedious process of handling the allocation of Grace marks to students belonging to a semester, impartially
- To do so, we set rules that govern the generation of grace marks and constrain them. Grace marks are generated on the basis of the proofs that have been submitted and verified
- The generated marks are then added to the existing marks and published.

Introduction to web designing:

What is HTML?

HTML is the language for describing the structure of Web pages. HTML gives authors the means to:

- Publish online documents with headings, text, tables, lists, photos, etc.
- Retrieve online information via hypertext links, at the click of a button.
- Design forms for conducting transactions with remote services, for use in searching for information, making reservations, ordering products, etc.
- Include spread-sheets, video clips, sound clips, and other applications directly in their documents

With HTML, authors describe the structure of pages using markup. The elements of the language label pieces of content such as “paragraph,” “list,” “table,” and so on

CSS:

Cascading Style Sheets (CSS) is a stylesheet language used to describe the presentation of a document written in HTML or XML. CSS describes how elements should be rendered on screen, on paper, in speech, or on other media. CSS is used to style and layout web pages — for example, to alter the font, colour, size, and spacing of your content, split it into multiple columns, or add animations and other decorative features. This module provides a gentle beginning to your path towards CSS mastery with the basics of how it works, what the syntax looks like, and how you can start using it to add styling to HTML.

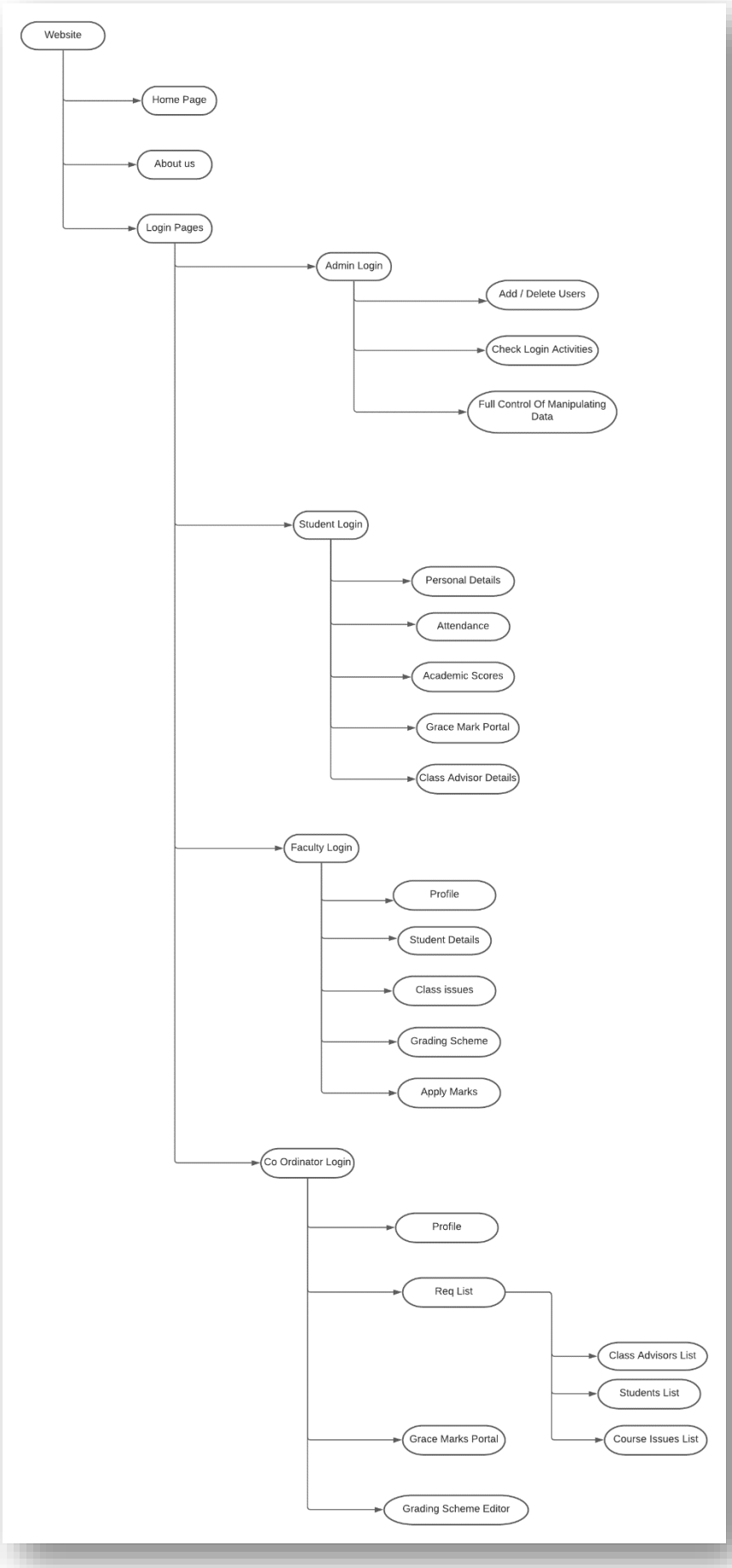
JavaScript:

JavaScript (JS) is a lightweight, interpreted, or just-in-time compiled programming language with first-class functions. While it is most well-known as the scripting language for Web pages, many non-browser environments also use it, such as Node.js, Apache CouchDB and Adobe Acrobat. JavaScript is a prototype-based, multi-paradigm, single-threaded, dynamic language, supporting object-oriented, imperative, and declarative styles.

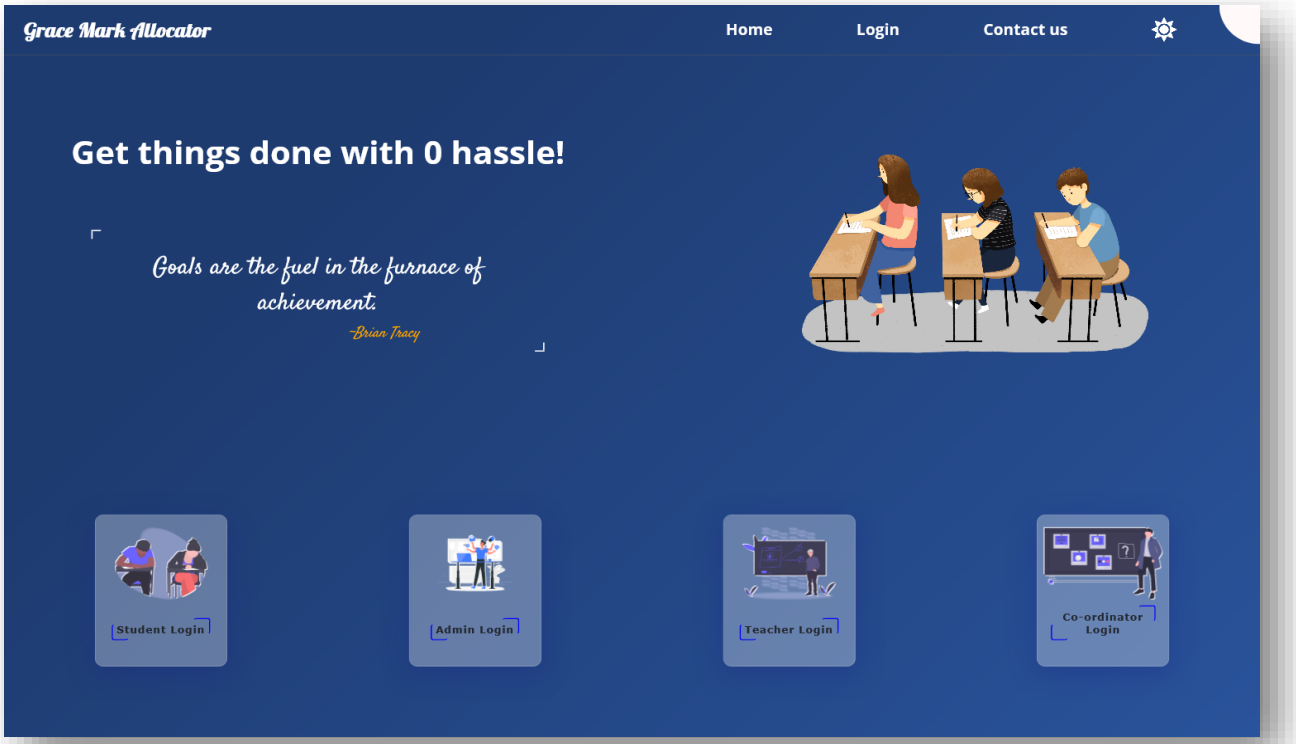
Embedded JavaScript:

You can embed JavaScript in an HTML document in the following ways: As statements and functions within a <SCRIPT> tag. ... See "Specifying a file of JavaScript code". By specifying a JavaScript expression as the value for an HTML attribute. See "Using JavaScript expressions as HTML attribute values".

Design diagram:



Implementation:



Description:

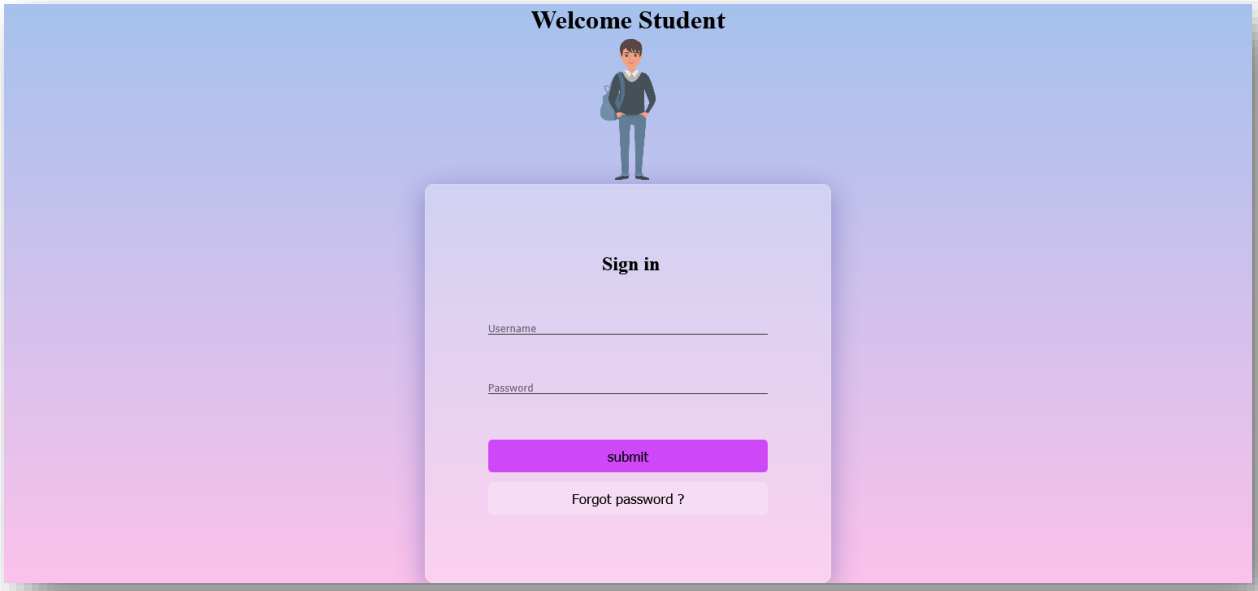
This page is the landing page of our website where users can choose their respective login page.

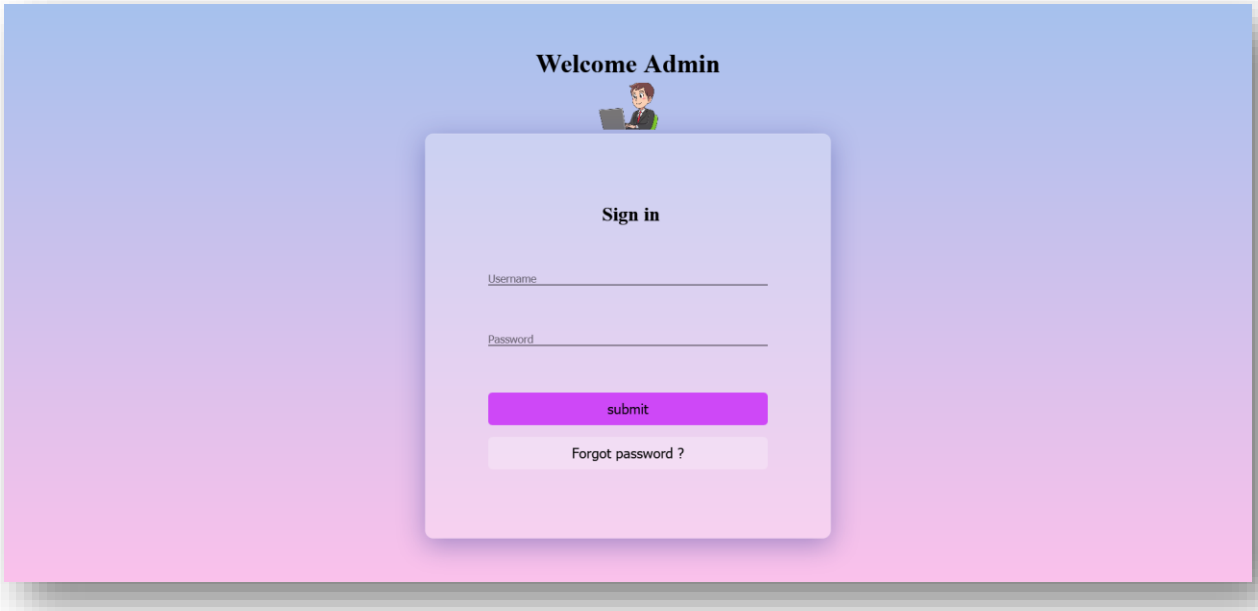
Code snippet:

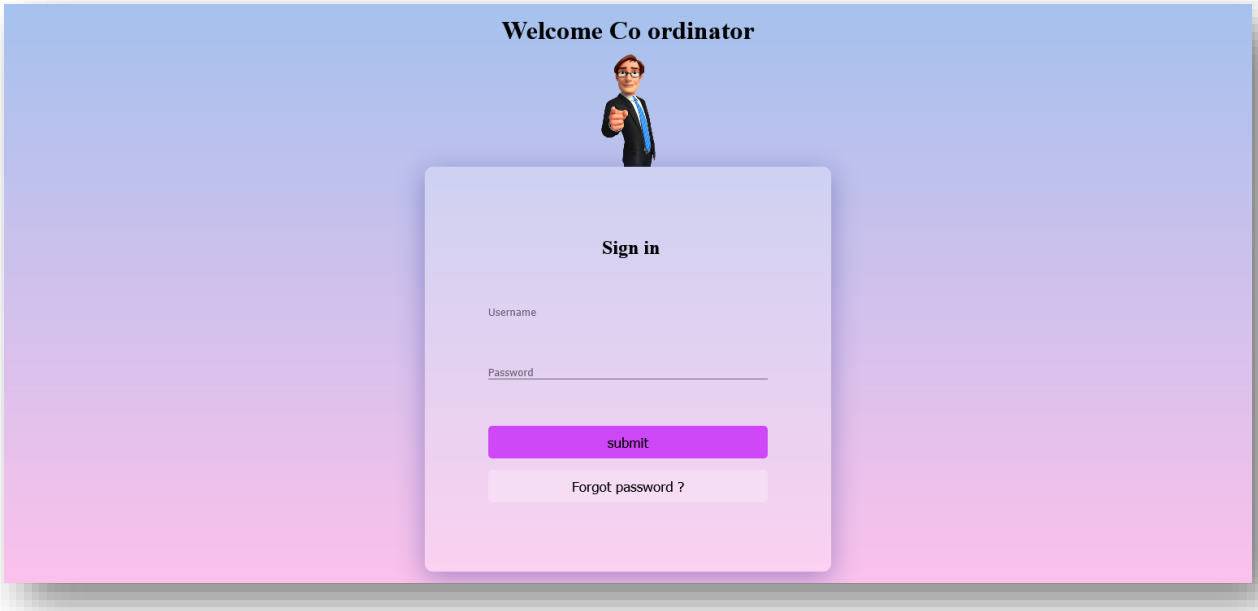
```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
    <link rel="stylesheet" href="http://localhost:5000/css/home.css" />
  </head>
  <form method="GET", action="http://localhost:8000/">
  <body>
    <div class="header">
      <h2 class="head">Grace Mark Allocator</h2>
```

```
      <div class="login">
        <a href="">Home</a>
        <a href="#box">Login</a>
        <a class="contact" href="">Contact us</a>
        <img class="mode" src="http://localhost:5000/img/sun.png" id="icon" />
        <!-- <a class="mode" href="">Mode</a> -->
      </div>
    </div>
    <div class="container">
      <div class="mini-cont">
        <h1>Get things done with 0 hassle!</h1>

        <div class="quotes">
          <blockquote class="quote"></blockquote>
          <div class="author"></div>
        </div>
      </div>
      <div class="image">
        <img
          src="http://localhost:5000/img/—Pngtree—
students at the college entrance_4519946.png"
          alt=""
        />
      </div>
    </div>
    <section id="section1">
      <div class="box" id="box">
        <img src="http://localhost:5000/img/undraw_exams_g4ow.svg" alt="" />
        <div class="btn-box">
          <button formaction="/Student">Student Login</button>
        </div>
      </div>
      <div class="box admin">
        <img src="http://localhost:5000/img/Admin-rafiki.svg" alt="" />
        <div class="btn-box">
          <button formaction="/Admin">Admin Login</button>
        </div>
      </div>
      <div class="box">
        <img src="http://localhost:5000/img/undraw_teaching_f1cm.svg" alt="" />
        <div class="btn-box">
          <button formaction="/Faculty">Teacher Login</button>
        </div>
      </div>
      <div class="box">
```

```
        <img src="http://localhost:5000/img/undraw_quiz_nlyh.svg" alt="" />
        <div class="btn-box">
          <button formaction="/Coordinator">Co-ordinator Login</button>
        </div>
      </div>
    </section>
    <nav class="nav-bar">
      <ul class="nav-links">
        <div class="ourTeam">
          <h3><a href="http://localhost:5000/contact.html"> Our Team </a></h3>
          <img src="http://localhost:5000/img/teams.png" alt="" />
        </div>
        <div class="git">
          <h3>
            <a
              href="https://github.com/awezrebel/Grace-marks-Calculator"
              target="blank"
              >Github</a
            >
          </h3>
          <img src="http://localhost:5000/img/git.png" alt="" />
        </div>
      </ul>
      <div class="contacts">
        <h2>Stay in touch!</h2>
      </div>
    </nav>

    <script src="http://localhost:5000/js/home.js"></script>
  </body>
</html>
```


Welcome Student
Sign in
Username
Password
submit
Forgot password ?


Welcome Admin
Sign in
Username
Password
submit
Forgot password ?



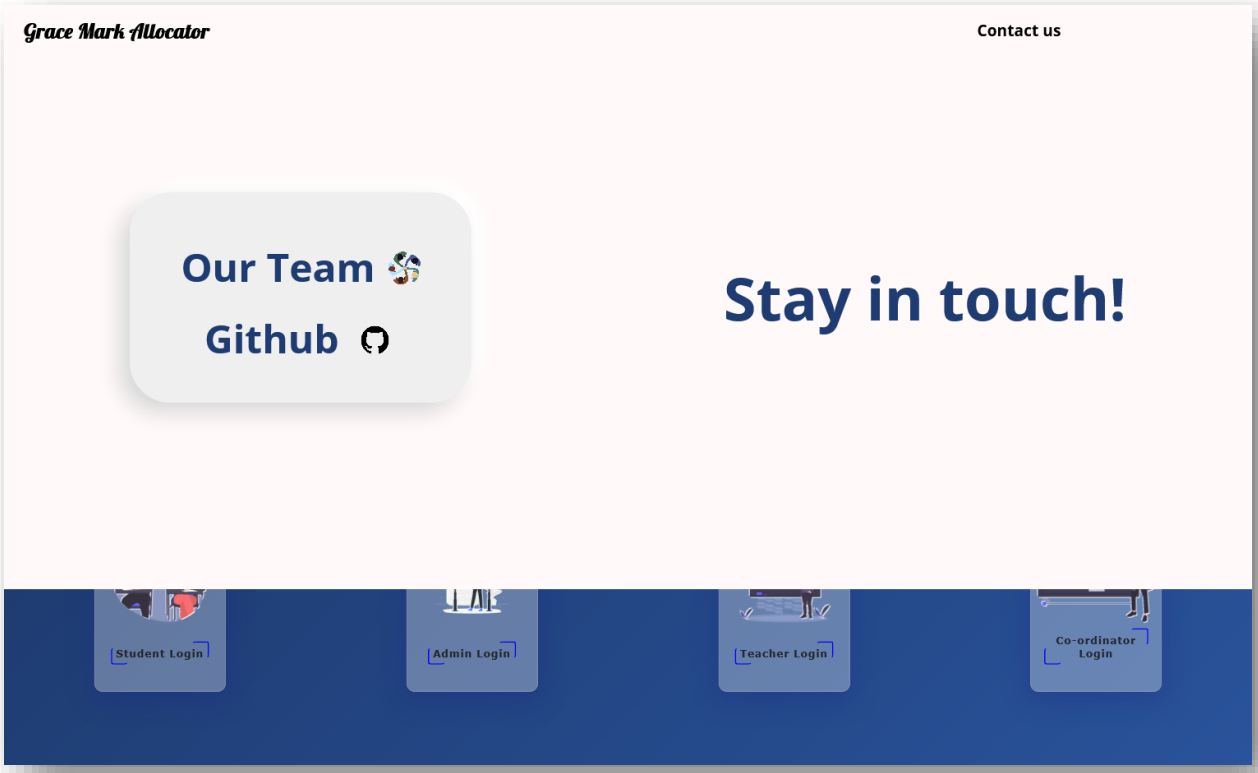
Description:

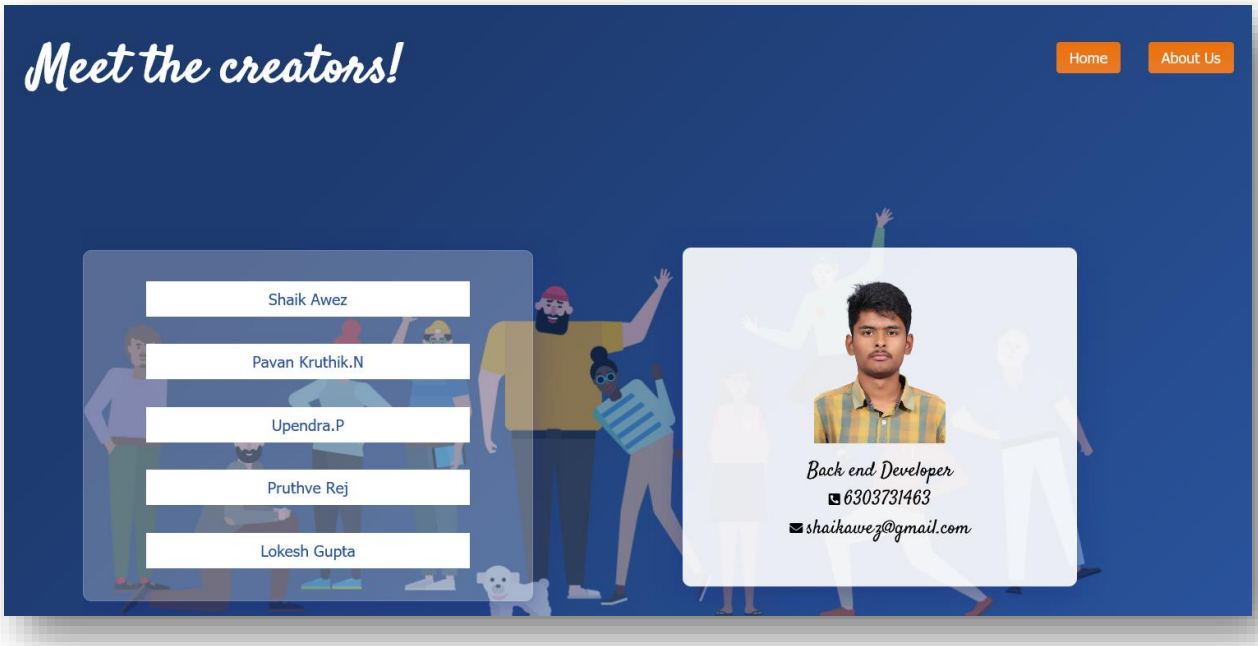
These are the different login pages done using single ejs file

Code snippet:

```
<!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>Document</title>
    <link rel="stylesheet"  type="text/css" href="http://localhost:5000/css/style
.css" />
</head>
<style>
       a{
        color: black;
    font-size: larger;
    text-align: center;
    margin-top: 5%;
    text-decoration: none;
    }
</style>

    <form method="GET", action="http://localhost:8000/">

<body>

    <div class="main-container">
        <div class="slider-container">
            <div class="left-slide">
                <div class="sign-in">
                    <h1>
                        <%= "Welcome "  +   welcome.Name %>
                    </h1>
        <img src ="  <%=welcome.image %>   "  width="100px">

                    <div class="sign-in-box">

                        <div class="head">

                            <h2>Sign in</h2>
                        </div>
                        <div class="username-field">
                            <i></i>
          <input type="text"  placeholder="Username"  name="username" />
```

```
                        </div>
                        <div class="password-field">
                            <i></i>
          <input type="password" placeholder="Password" name="password"/>
                        </div>
                        <button   formaction="/homelogin">submit</button>
                        <br><br>

<button  style="backgroundcolor:inherit;"  formaction="forgot">Forgot password ?<
/button>
                    </div>
                    </div>
            </div>
        </div>

    </div>
    <script src="http://localhost:5000/home.js"></script>
</body>

</html>
```


Grace Mark Allocator
Contact us
Our Team
Github
Stay in touch!
Student Login
Admin Login
Teacher Login
Co-ordinator Login


Meet the creators!
Home
About Us
Shaik Awez
Pavan Kruthik.N
Upendra.P
Pruthve Rej
Lokesh Gupta
Back end Developer
6303731463
shaikawez@gmail.com

Description:

About us and git hub link From Home page

Code snippet:

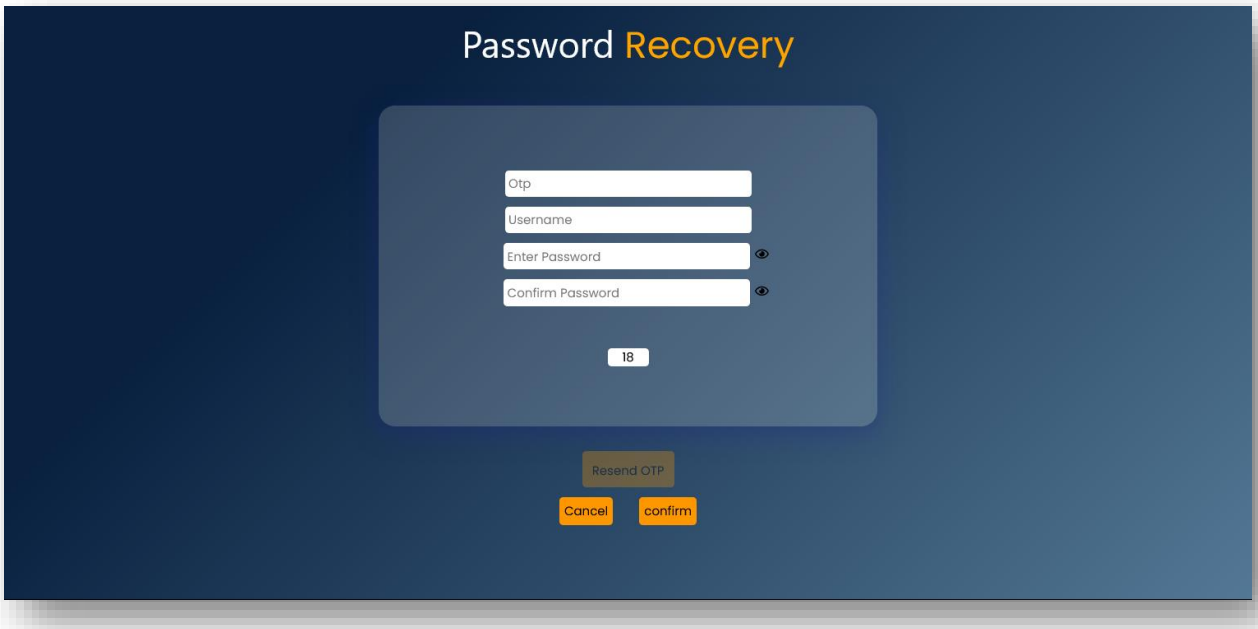
```
<!DOCTYPE html>
<html lang="en">

<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>Contact Us</title>
    <link rel="stylesheet" href="http://localhost:5000/css/contact.css" />
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/5.15.4/css/all.min.css" integrity="sha512-
1ycn6IcaQQ40/MKBW2W4Rhis/DbILU74C1vSrLJxCq57o941Ym01SwNsOMqvEBFlcgUa6xLiPY/NS5R+E
6ztJQ==" crossorigin="anonymous" referrerpolicy="no-referrer" />

</head>

<style>
.home {
position: absolute;
top: 0.8rem;
right: 10rem;
}


.home button {
background: #ed710c;
background-image: -webkit-linear-gradient(top, #ed710c, #e87c2a);

background-image: linear-gradient(to bottom, #ed710c, #e87c2a);

border-radius: 4px;
text-align: center;

color: #ffffff;
font-size: 18px;
padding: 0.5rem;
padding-right: 1rem;
padding-left: 1rem;
text-decoration: none;
```

```
border: none;
cursor: pointer;
transition: all 0.5s ease-in-out;
}

.home button:hover {
background: #3cb0fd;
text-decoration: none;
}



</style>

<body>
    <div class="header">
        <div class="messsage">
            <h1>Meet the creators!</h1>
        </div>
        <div class="about-us">
            <button>About Us</button>

        </div>
        <div class="home">
            <form method="get" action="http://localhost:8000/home">
                <button type="submit">Home</button>
            </form>

        </div>
    </div>

    <div class="img-container">
        <div class="image">
            <img src="http://localhost:5000/img/Teammates.png" />
        </div>

        <div class="creator-btns">
            <div class="click-btns">
                <a href="#item1">

                    <button>Shaik Awez</button>
                </a>
            </div>
            <div class="click-btns">
                <a href="#item2">
```

```
                <button>Pavan Kruthik.N</button>
            </a>

        </div>
        <div class="click-btns">
            <a href="#item3">

                <button>Upendra.P</button>
            </a>

        </div>
        <div class="click-btns">
            <a href="#item4">

                <button>Pruthve Rej</button>
            </a>

        </div>
        <div class="click-btns">
            <a href="#item5">

                <button>Lokesh Gupta</button>
            </a>

        </div>
    </div>


    <div class="creator-container">
        <div class="item" id="item1">
            <img src="http://localhost:5000/img/awez.png" alt="">
            <span>Back end Developer</span>
            <div class="details">
                <i class="fas fa-phone-square-alt"></i>
                <span>6303731463</span>
                <div class="mail">

                    <i class="fas fa-envelope"></i>
                    <span>shaikawez@gmail.com</span>
                </div>
            </div>


        </div>
```

```
            <div class="item" id="item2">
                <img src="http://localhost:5000/img/pavan.png" alt="">
                <span>Front end Developer</span>
                <div class="details">
                    <i class="fas fa-phone-square-alt"></i>
                    <span>8341900899</span>
                    <div class="mail">

                        <i class="fas fa-envelope"></i>
                        <span>kruthik24@gmail.com</span>
                    </div>
                </div>


            </div>
            <div class="item" id="item3">
                <img src="http://localhost:5000/img/upendra.png" alt="">
                <span>Front end Developer</span>
                <div class="details">
                    <i class="fas fa-phone-square-alt"></i>
                    <span>9182410276</span>
                    <div class="mail">

                        <i class="fas fa-envelope"></i>
                        <span>p_upendra@gmail.com</span>
                    </div>
                </div>
            </div>
            <div class="item" id="item4">
                <img src="http://localhost:5000/img/pruthve.png" alt="">
                <span>Tester</span>
                <div class="details">
                    <i class="fas fa-phone-square-alt"></i>
                    <span>9381044045</span>
                    <div class="mail">

                        <i class="fas fa-envelope"></i>
                        <span>p_pruthve@gmail.com</span>
                    </div>
                </div>
            </div>
            <div class="item" id="item5">
                <img src="http://localhost:5000/img/lokesh.png" alt="">
                <span>Tester</span>
                <div class="details">
```

```
                    <i class="fas fa-phone-square-alt"></i>
                    <span>8142838171</span>
                    <div class="mail">
                        <i class="fas fa-envelope"></i>

                        <span>g_lokesh@gmail.com</span>
                    </div>
                </div>
            </div>

        </div>
    </div>

    <script src="http://localhost:5000/js/contact.js"></script>
</body>

</html>
```



Description:

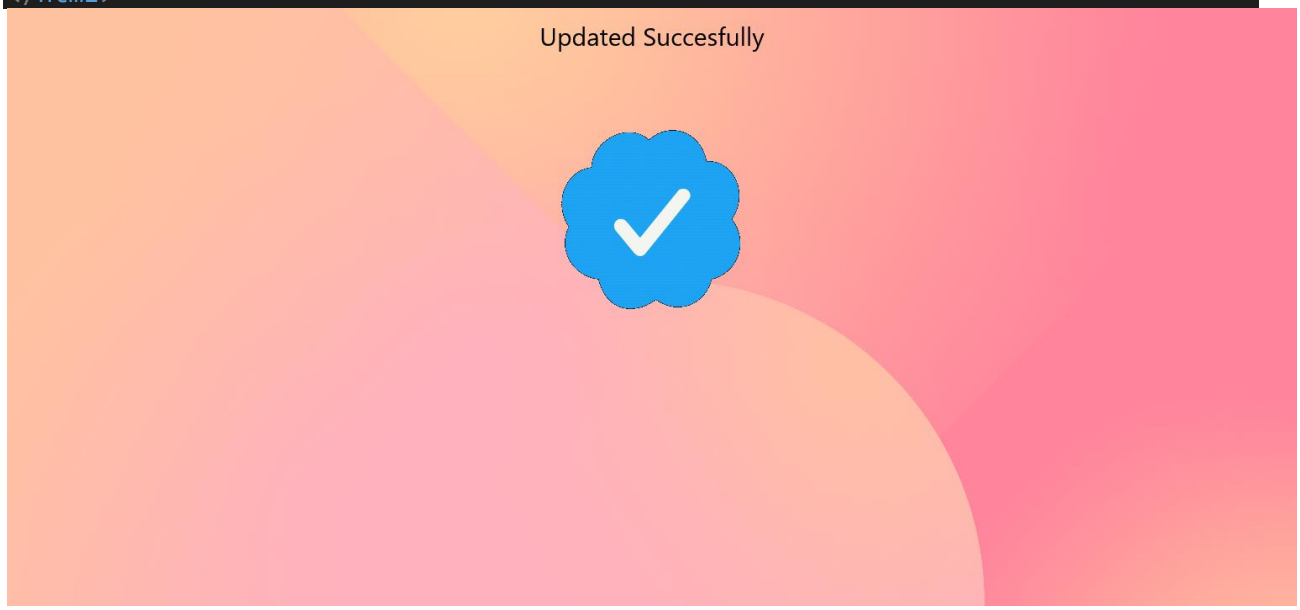
Forgot password page , with the help of otp generated to mobile can reset password , done validations for each input feild

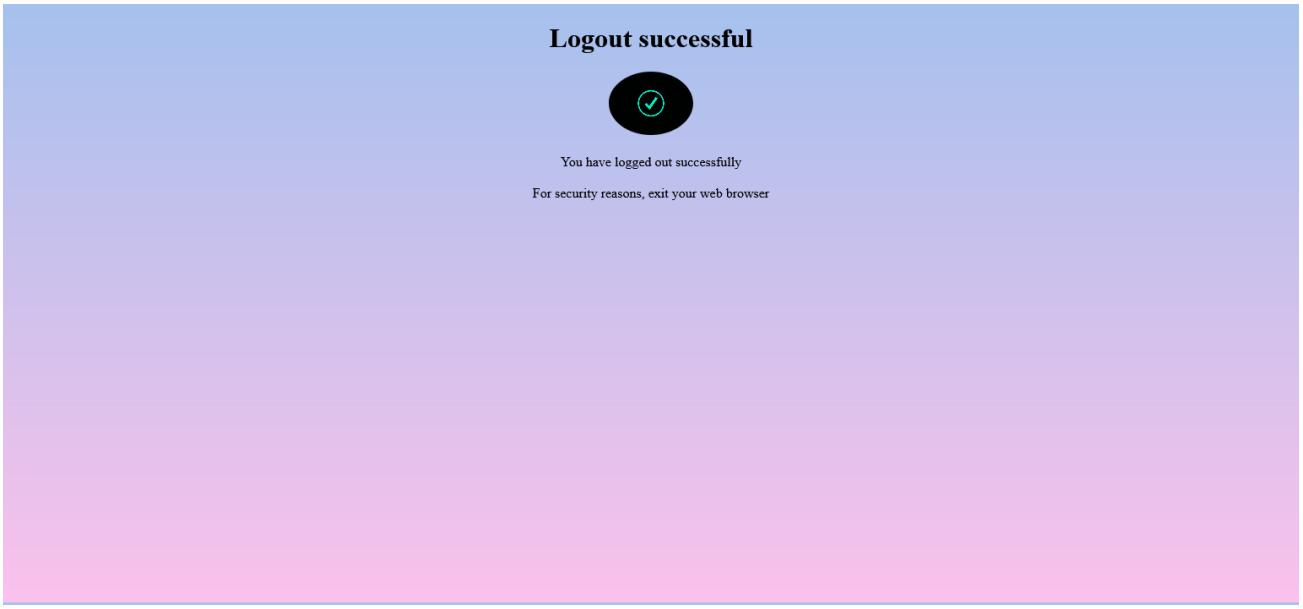
Code snippet:

```
<!DOCTYPE html>
<html lang="en">

<head>
  <title>OTP</title>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/5.13.0/css/all.min.css">

<body>

  <header>
    <h2>Password <span>Recovery</span></h2>
  </header>
  <form method="GET", action="http://localhost:8000/">

      <div class="inputs">


<input type="text" placeholder="Otp" name="otp">
<input type="text" placeholder="Username" name="uname1">
<div class="password">
<input type="password" placeholder="Enter Password"   name="pwd" id="pwd">
<em class="far fa-eye" onclick="password()"></em>

        </div>
        <div class="confirm-password">

<input type="password" placeholder="Confirm Password"   name="cpwd" id="cpwd">
          <em class="far fa-eye" onclick="confirmpassword()"></em>
        </div>
        <div id="countdown"></div>

      </div>
```

```
    <script>
      function password() {
        var x = document.getElementById("pwd");
        if (x.type === "password") {
          x.type = "text";
        } else {
          x.type = "password";
        }
      }

      function confirmpassword() {
        var x = document.getElementById("cpwd");
        if (x.type === "password") {
          x.type = "text";
        } else {
          x.type = "password";
        }
      }
    </script>

    <button class="w3-btn w3-
orange" id="mybutton" formaction="/otp">Resend OTP</button> <br><br>
    <p>


<button  formaction="/home" , class="w3-btn w3-
orange">Cancel</button>      
<button type="submit" , class="w3-btn w3-orange">confirm</button> <br><br>
    <p></p>
    <script>
      const htmlBtn = document.getElementById("mybutton");
      htmlBtn.disabled = true;
    </script>


    <script>
      alert("otp sent succesfully")
      var timeleft = 20;
      var downloadTimer = setInterval(function () {
        if (timeleft <= 0) {
          clearInterval(downloadTimer);
          document.getElementById("countdown").innerHTML = " ", htmlBtn.disabled
= false;
        } else {
          document.getElementById("countdown").innerHTML = timeleft;
```

```
        }
        timeleft -= 1;
      }, 1000);

    </script>

  </form>


</body>

</html>
```





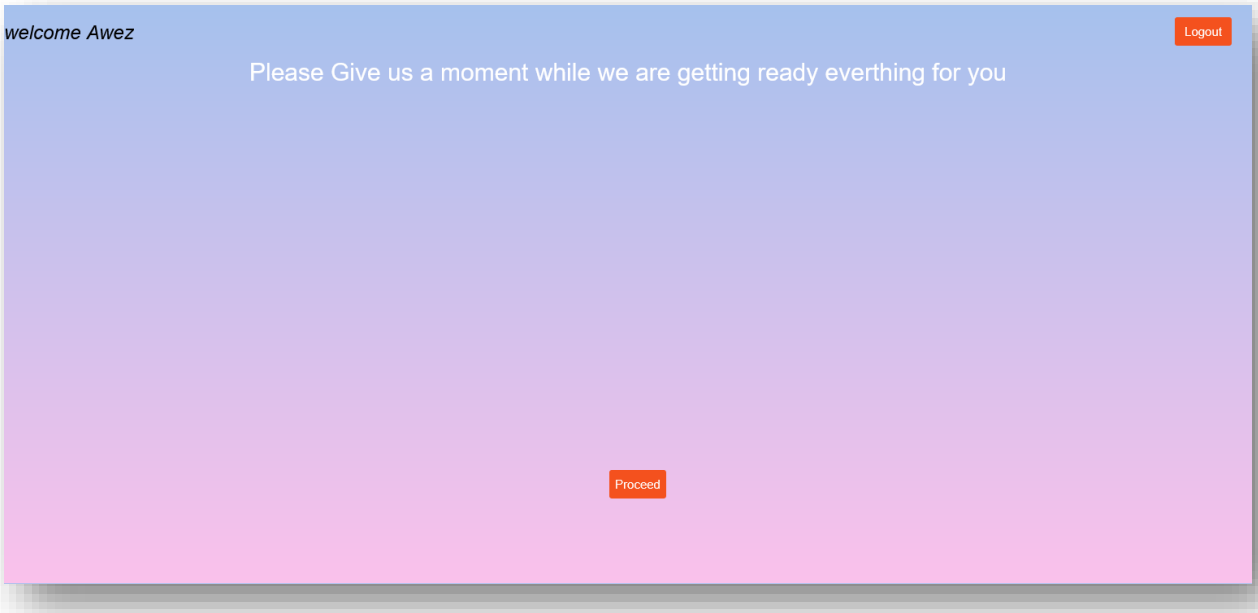
Description :

Basic logout page (Backward prevention script added), updation Acknowledge page after password reset

Code snippet:

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<style>
body{
    background-image: linear-gradient(to top, #fbc2eb 0%, #a6c1ee 100%);
    height: 95.5vh;

    justify-content: center;
    align-items: center;
     text-align: center;
}
img {
  border-radius: 50%;
}
</style>
<title>Logout</title>
<form method="GET", action="http://localhost:8000/">
</head>
<body>


    <script type="text/javascript">
        window.history.forward();
        function noBack()
        {

            window.onbeforeunload = function() { return "Your work will be lost."
; };
        }
    </script>

<body onLoad="noBack();" onpageshow="if (event.persisted) noBack();" onUnload="">
```

```
<h1>Logout successful </h1>
<img src="https://i.pinimg.com/originals/70/a5/52/70a552e8e955049c8587b2d7606cd6a6.gif" width="100" alt="image not available"></img></br><br>
<p1> You have logged out successfully </p1><br>
 <br>
<p1> For security reasons, exit your web browser </p1> <br> </br>



<script type = "text/javascript" >
function preventBack() { window.history.forward(); }
setTimeout("preventBack()", 0);
window.onunload = function () { null };
</script>
</body>
</html>
```



Description:

Welcome page with name , Fetched According to Login id with an Loading Animation

Code snippet:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <link rel="stylesheet" href="css/welcome.css">


<body>
<form method="GET", action="http://localhost:8000/">


        <% loginwelcome.forEach(function (login) { %>

           <h1>
             <%= "welcome  "  + login.name%>
            <% }) %>
         </h1>
       <h2> Please Give us a moment while  we are getting ready everthing for you
</h2>




</tbody>
</table>
<button class="logout" formaction="/warning">Logout</button>
<button class="proceed" id="proceed" formaction="/data">Proceed</button>
</body>
</html>
```

Evaluation sheet:

Roll No	Technology	Max Marks	Marks Awarded	Total (30)
18117	HTML CSS	10 10		
18136	HTML CSS	10 10		
18137	HTML CSS	10 10		
18138	HTML CSS	10 10		
18148	HTML CSS	10 10		
	Project Documentation	10		

B. Tech. project Grace Marks Allocator– G 22

PROJECT REPORT-II

Submitted by

S. No	Name	Roll Number
1.	G Lokesh	CB.EN.U4CSE18117
2.	P Pruthve Rej	CB.EN.U4CSE18136
3.	N Pavan Kruthik	CB.EN.U4CSE18137
4.	P Upendra	CB.EN.U4CSE18138
5.	Sk Awez	CB.EN.U4CSE18148

is fulfilment of the requirements for the Course – 15CSE376 (Net Centric Programming)

BACHELOR OF TECHNOLOGY IN

COMPUTER SCIENCE AND ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

AMRITA SCHOOL OF ENGINEERING

AMRITA VISHWA VIDYAPEETHAM

COIMBATORE -641112

October 2021

Table of Contents

Technologies Learnt:

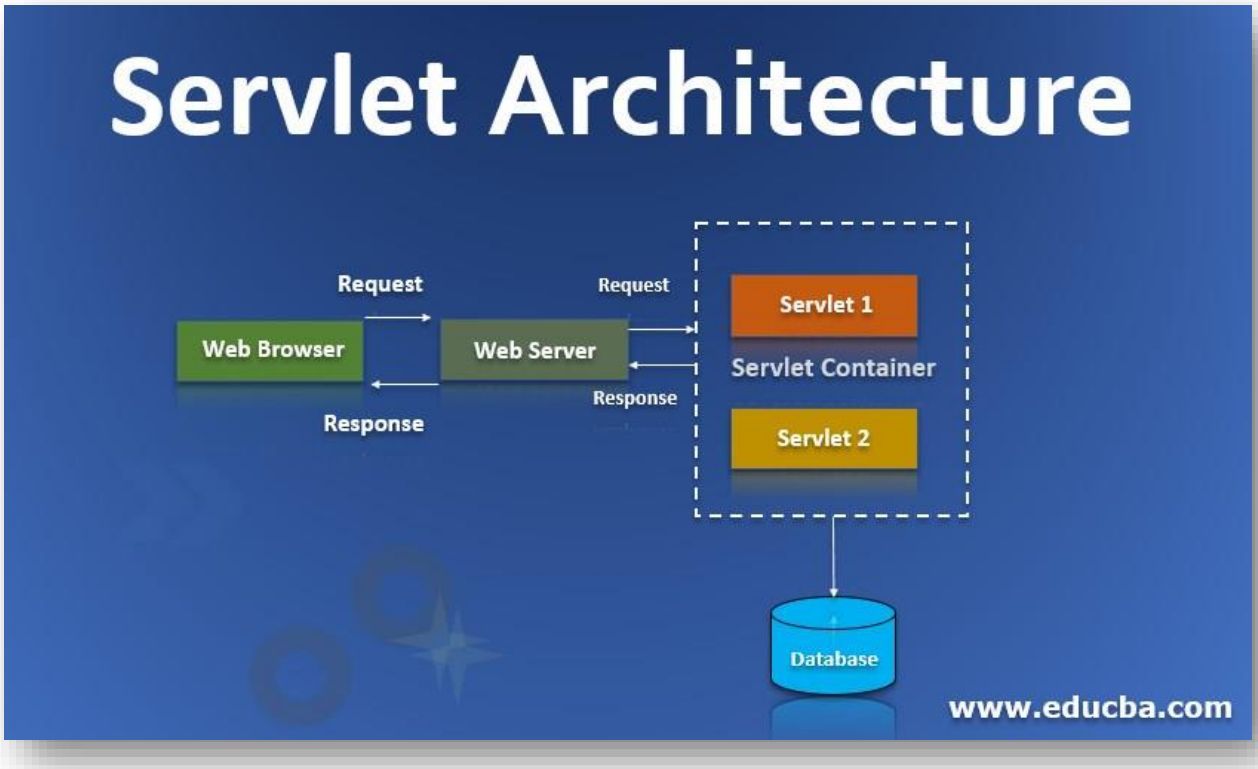
Servlets

Servlets are the Java programs that run on the Java-enabled web server or application server

They are used to handle the request obtained from the web server, process the request, produce the response, then send a response back to the web server.
Properties of Servlets are as follows:

● Servlets work on the server-side.

● Servlets are capable of handling complex requests obtained from the web server.



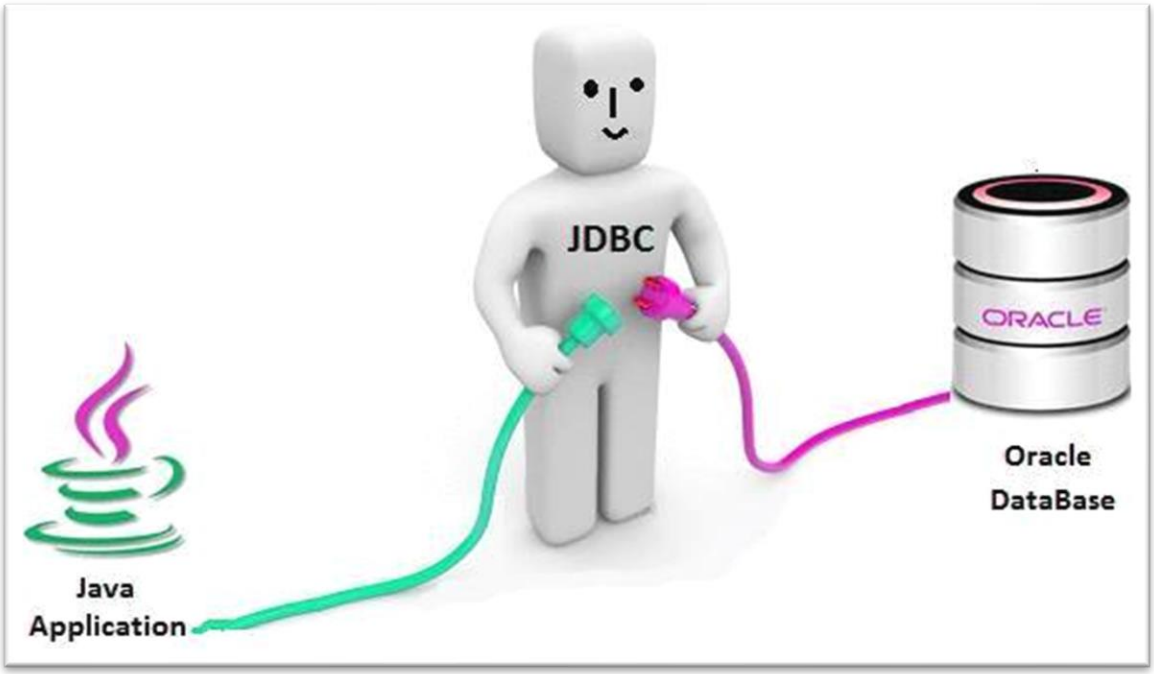
Execution of Servlets involves six basic steps:

1. The clients send the request to the web server.
2. The web server receives the request.
3. The web server passes the request to the corresponding servlet.
4. The servlet processes the request and generates the response in the form of output.
5. The servlet sends the response back to the web server.
6. The web server sends the response back to the client and the client browser displays it on the screen.

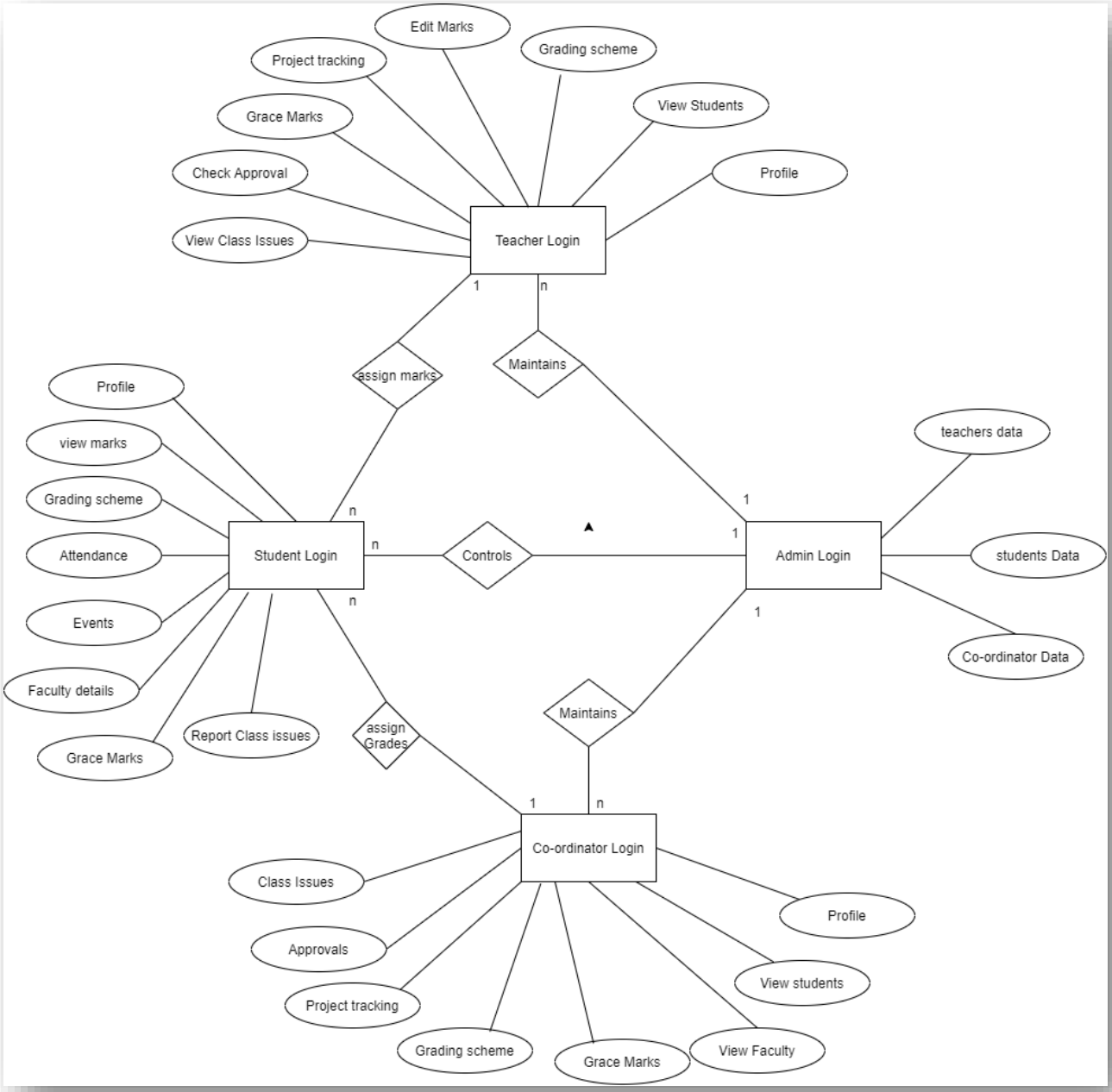
JDBC

JDBC stands for Java Database Connectivity. JDBC is a Java API to connect and execute the query with the database. It is a part of Java SE (Java Standard Edition). JDBC API uses JDBC drivers to connect with the database.

We can use JDBC API to access tabular data stored in any relational database. With the help of JDBC API, we can save, update, delete and fetch data from the database.
We can use JDBC API to handle database using Java program and can perform the following activities:
1. Connect to the database
2. Execute queries and update statements to the database
3. Retrieve the result received from the database.



ER Diagram:


Edit Marks
Project tracking
Grading scheme
Grace Marks
View Students
Check Approval
Profile
View Class Issues
Teacher Login
1
n
assign marks
Maintains
Profile
view marks
Grading scheme
Attendance
Events
Faculty details
Grace Marks
Report Class issues
assign Grades
Student Login
n
n
n
Controls
1
1
1
Admin Login
teachers data
students Data
Co-ordinator Data
Maintains
1
n
Co-ordinator Login
Class Issues
Approvals
Project tracking
Grading scheme
Grace Marks
View Faculty
View students
Profile

Architecture diagram:

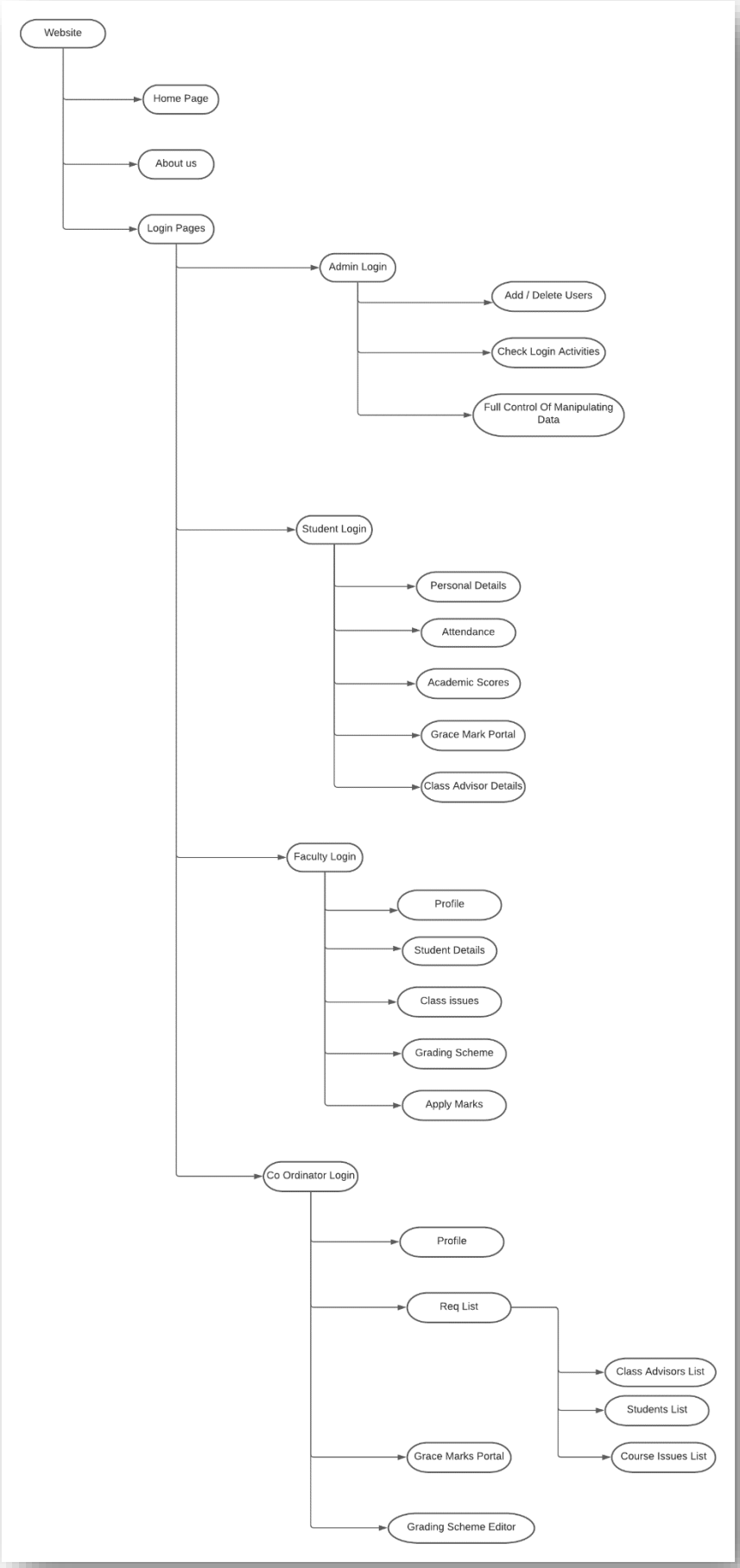
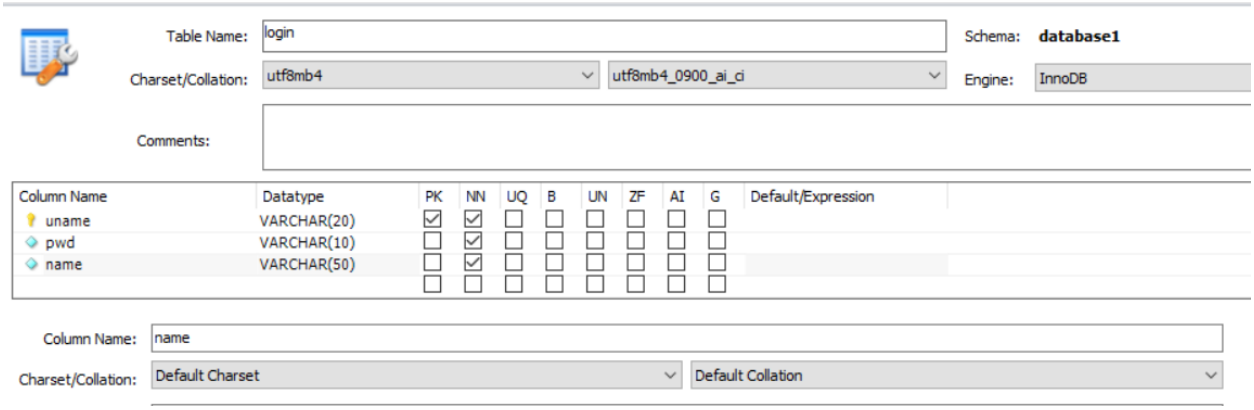
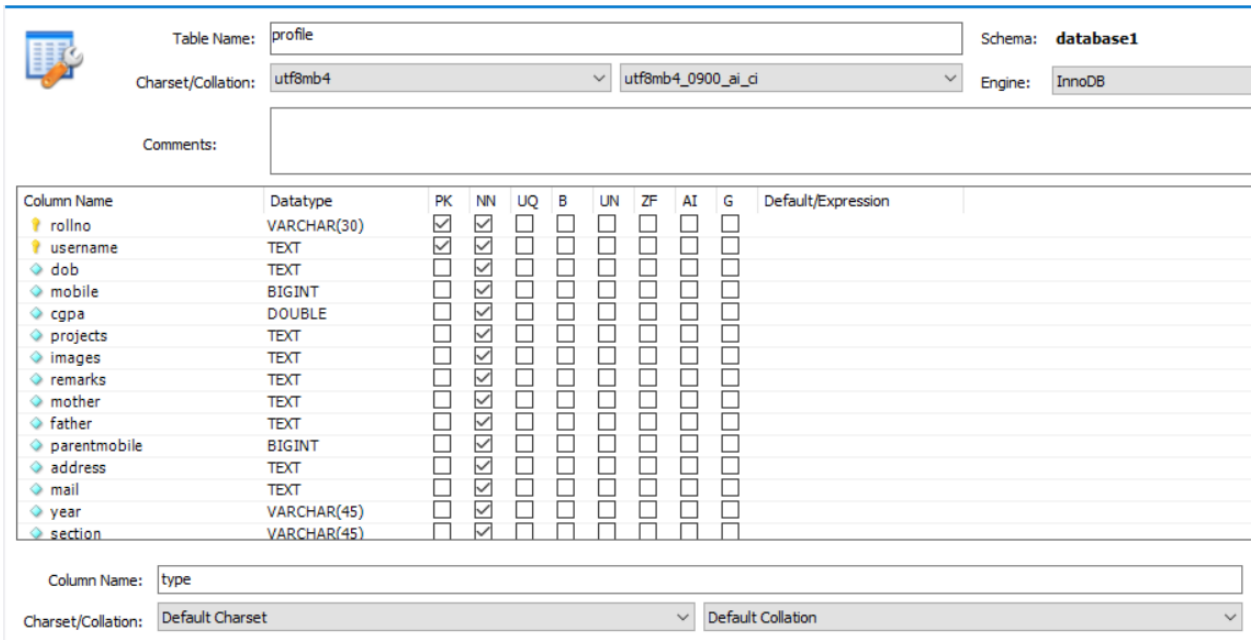


Table Specifications:

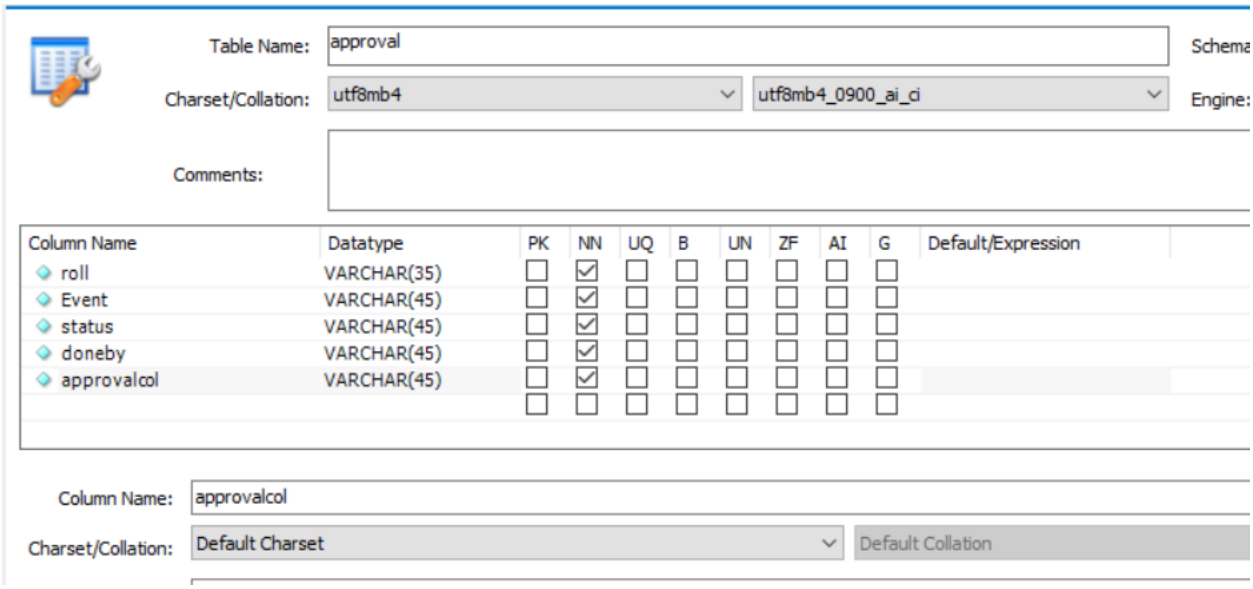
1)Users



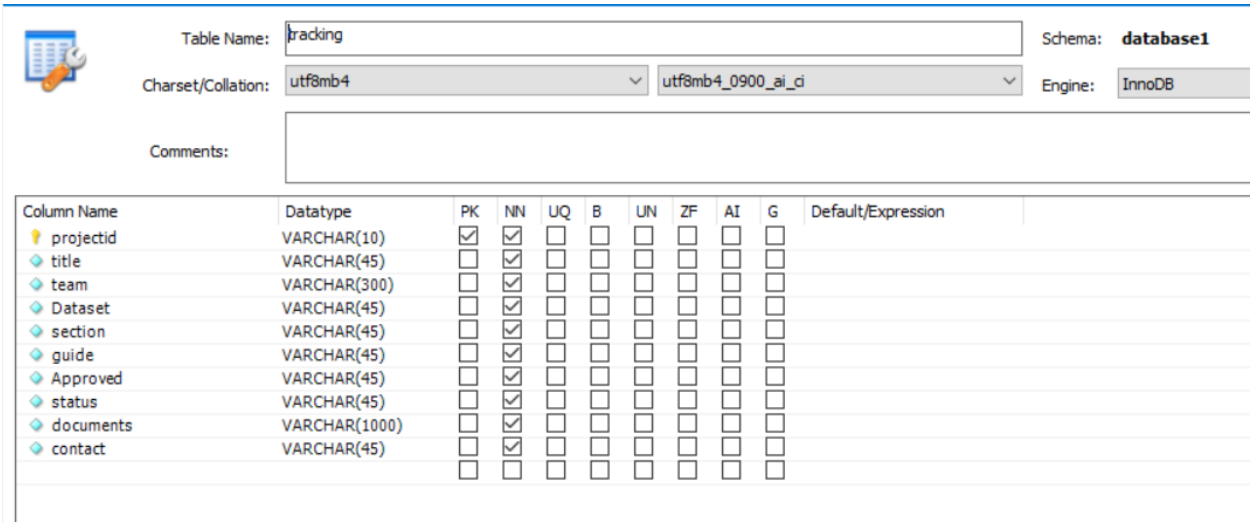
2)Profile



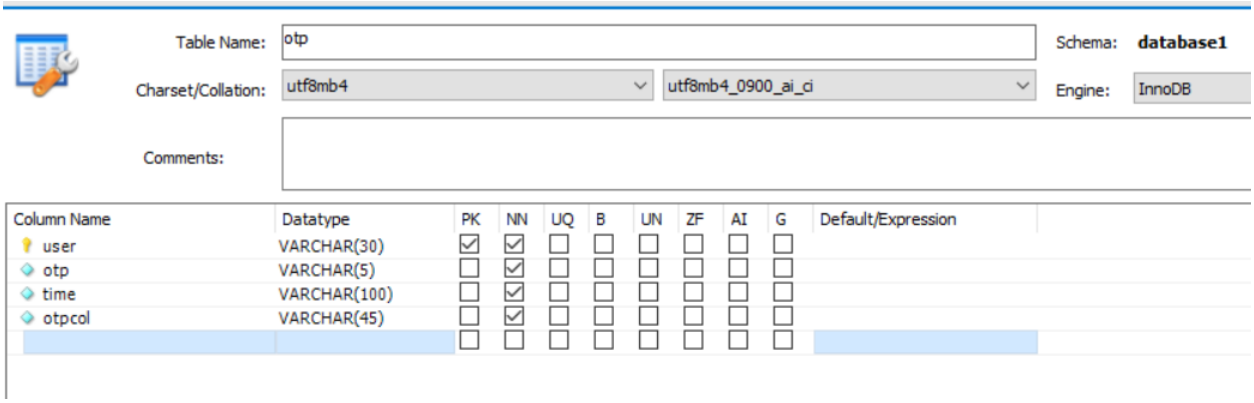
3) Approval



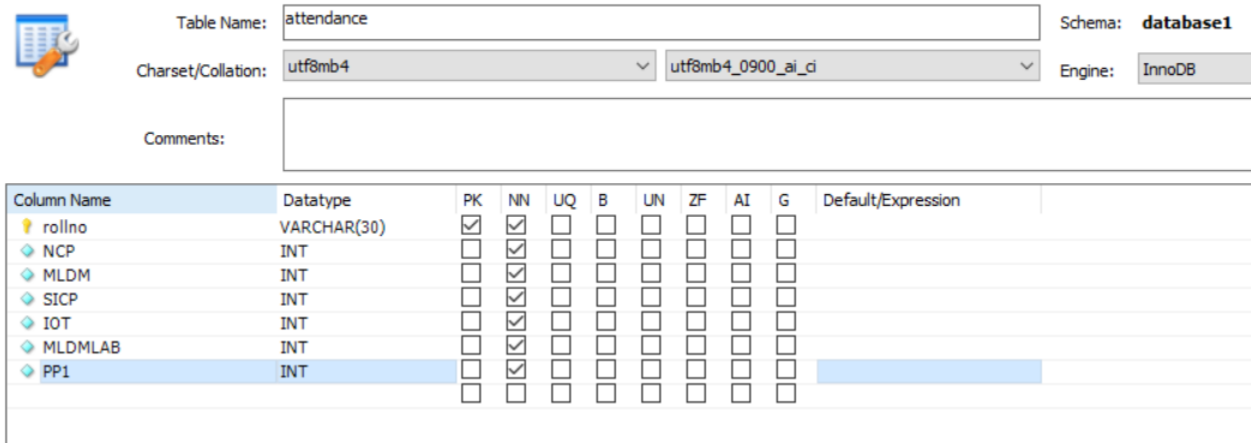
4)Project Tracking



5)Otp



6)Attendance



Functionalities using servlets

1) **Login Action**
   a) Path :/login
   b) Method: Get
   c) Input: username, password

2) **Enroll Student**

   a)Path:/enroll

   b)Method:Get

   c)Input:Students Data

3) **Marks Re enter**

   a)Path:/editmarks

   b)Method:Get

   c)Input:Marks Record

4) **Approval**
   a)Path:/approval
   b)Method:Get
   c)Input:Co Ordinator Approval

5) **Class Issues**
   a)Path:/issue
   b)Method:Get
   c)Input:Students Feedback

6) **Admin Access**
   a)Path:/admin
   b)Method:Get
   c)Input:Login Credentials

7) **Block User**
   a)Path:/block
   b)Method:Get
   c)Input:Reamark , Rollno

8) **Event Mark**
   a)Path:/graceedit
   b)Method:Get
   c)Input:Mark , Eventype

9) **Grade**
   a)Path:/Grade
   b)Method:Get
   c)Input:Mark , Subcode

Database connectivity

Java:

```
System.out.println("\n\n***** MySQL JDBC Connection Testing *****");
        Connection conn = null;
        try
        {
        Class.forName ("com.MySQL.jdbc.Driver").newInstance ();
            String userName = "root";
            String password = "Awez@0987";
            String url = "jdbc:MySQL://localhost/database1";
            conn = DriverManager.getConnection (url, userName, password);
            System.out.println ("\nDatabase Connection Established...");
        }
       catch (Exception ex)
        {
            System.err.println ("Cannot connect to database server");
            ex.printStackTrace();
        }
```

Javascript:

```
var mysql = require('mysql');
var con = mysql.createConnection({
  host: "localhost",
  user: "root",
  password: "Awez@0987"
});

con.connect(function(err) {
  if (err) throw err;
  console.log("Connected!");
});
```

Insertion:

```
String query="insert into approval (roll , Event , status) values (?,?,?,?,?,?)";
   PreparedStatement st=con.prepareStatement(query);
   st.setString(1,rollno);
   st.setString(2,event);
   st.setString(3,status);
```

Updation:

```
String query="UPDATE database1.marks SET  "+ subcode + "= ? WHERE (rollno = ?) ";
   PreparedStatement st=con.prepareStatement(query);

   st.setString(1,mark);
   st.setString(2,rollno);
   int check=st.executeUpdate();
   return check;
```

Validation:

```
String username=  req.getParameter("username");
String password= req.getParameter("password");

LoginDao dao = new LoginDao();

try {
    if(dao.connectdb(username, password)) {
        HttpSession session = req.getSession();
        res.sendRedirect("http://localhost:8000/Student");
    }else {
        res.sendRedirect("http://localhost:8000/?username=+&password=+");

    }
```

Validation

```
if(username != undefined && password != undefined) {
fs.truncate('currentlogin.txt', 0, function() {
console.log("File Content Deleted");
});

fs.readFile('block.txt', 'utf-8', (err, data) => {  if (err) throw err;

if(username==data){
res.send("Your account is blocked ..pls login after 24hrs");
}else{

var test = 0;
var pd=password;
fs.readFile('logintype.txt', 'utf-8', (err, data) => {
var sqlquery='SELECT * from ';
sqlquery +=data;

mycon.query(sqlquery, function (error,login, fields) {
if (error) throw error;
var length = login.length;
var m=null;
for(var i = 0; i < length; i++){
if (login[i].uname==username){
m=i;
}
}
if(m==null){
res.sendfile("oops.html");
}else{
if(login[m].pwd==password){
test=1
fs.appendFile('currentlogin.txt', username , (err) => {
if (err) throw err;
const sql5=`INSERT INTO database1.currentlogin (user) VALUES ('${username}')`;
mycon.query(sql5, function (err1, result) {
if (err1) throw err1;
console.log(result);
});
})
console.log(test);
if(test==1){       res.redirect("/loginwelcome");  }
```

Evaluation sheet:

Roll No	Technology	Max Marks	Marks Awarded	Total (30)
18117	Servlet JDBC	10 10		
18136	Servlet JDBC	10 10		
18137	Servlet JDBC	10 10		
18138	Servlet JDBC	10 10		
18148	Servlet JDBC	10 10		
	Project Documentation	10		

B. Tech. project Grace Marks Allocator– G 22

PROJECT REPORT-III

Submitted by

S. No	Name	Roll Number
1.	G Lokesh	CB.EN.U4CSE18117
2.	P Pruthve Rej	CB.EN.U4CSE18136
3.	N Pavan Kruthik	CB.EN.U4CSE18137
4.	P Upendra	CB.EN.U4CSE18138
5.	Sk Awez	CB.EN.U4CSE18148

is fulfilment of the requirements for the Course – 15CSE376 (Net Centric Programming)

BACHELOR OF TECHNOLOGY IN

COMPUTER SCIENCE AND ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

AMRITA SCHOOL OF ENGINEERING

AMRITA VISHWA VIDYAPEETHAM

COIMBATORE -641112

November 2021

Table of Contents

Abstract

Technologies Used

HTML

HTML is the language for describing the structure of Web pages. HTML gives authors the means to:

● Publish online documents with headings, text, tables, lists, photos, etc.
● Retrieve online information via hypertext links, at the click of a button.
● Design forms for conducting transactions with remote services, for use in searching for information, making reservations, ordering products, etc.
● Include spread-sheets, video clips, sound clips, and other applications directly in their
documents

With HTML, authors describe the structure of pages using markup. The elements of the
language label pieces of content such as “paragraph,” “list,” “table,” and so on

CSS:

Cascading Style Sheets (CSS) is a stylesheet language used to describe the presentation of a document written in HTML or XML. CSS describes how elements should be rendered on screen, on paper, in speech, or on other media. CSS is used to style and layout web pages — for example, to alter the font, colour, size, and spacing of your content, split it into multiple columns, or add animations and other decorative features. This module provides a gentle beginning to your path towards CSS mastery with the basics of how it works, what the syntax looks like, and how you can start using it to add styling to HTML.

JavaScript:

JavaScript (JS) is a lightweight, interpreted, or just-in-time compiled programming language with first-class functions. While it is most well-known as the scripting language for Web pages, many non-browser environments also use it, such as Node.js, Apache CouchDB and Adobe Acrobat. JavaScript is a prototype-based, multi-paradigm, single-threaded, dynamic language, supporting object-oriented, imperative, and declarative styles.

Embedded JavaScript:

You can embed JavaScript in an HTML document in the following ways: As statements and functions within a <SCRIPT> tag. ... See "Specifying a file of JavaScript code". By specifying a JavaScript expression as the value for an HTML attribute. See "Using JavaScript expressions as HTML attribute values".

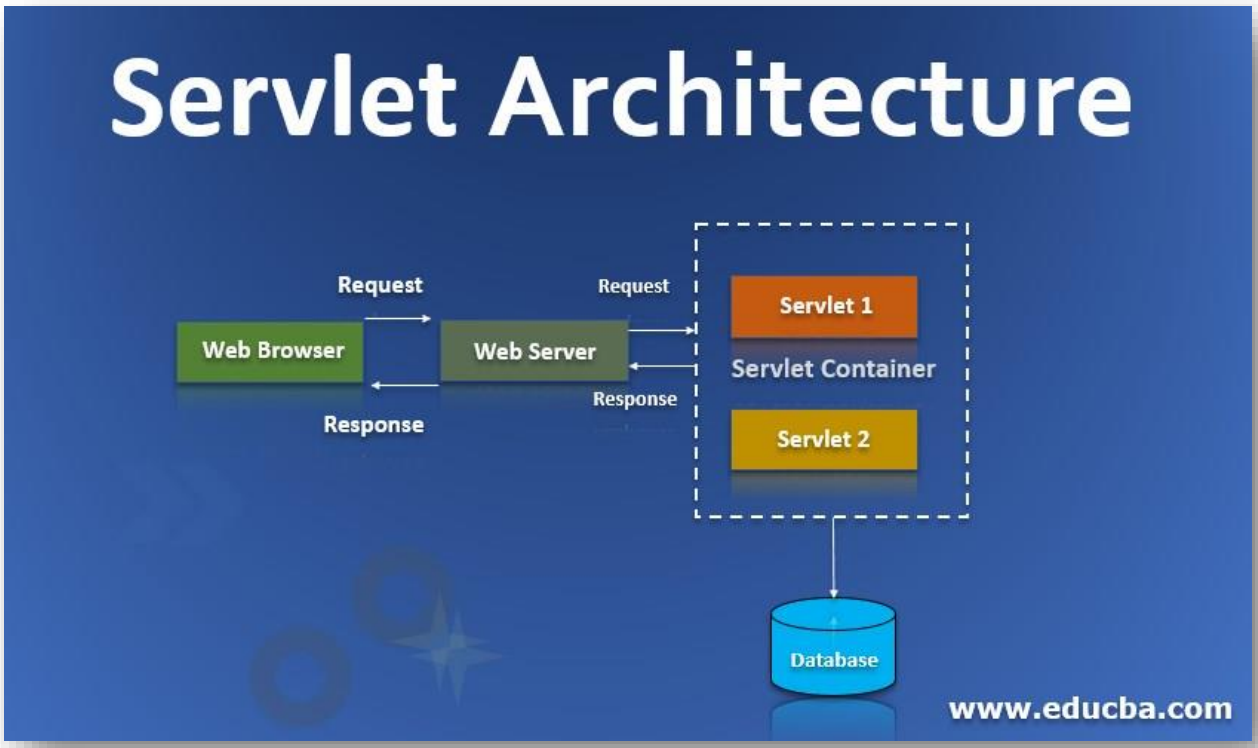
Servlets

Servlets are the Java programs that run on the Java-enabled web server or application server

They are used to handle the request obtained from the web server, process the request, produce the response, then send a response back to the web server.
Properties of Servlets are as follows:

● Servlets work on the server-side.

● Servlets are capable of handling complex requests obtained from the web server.



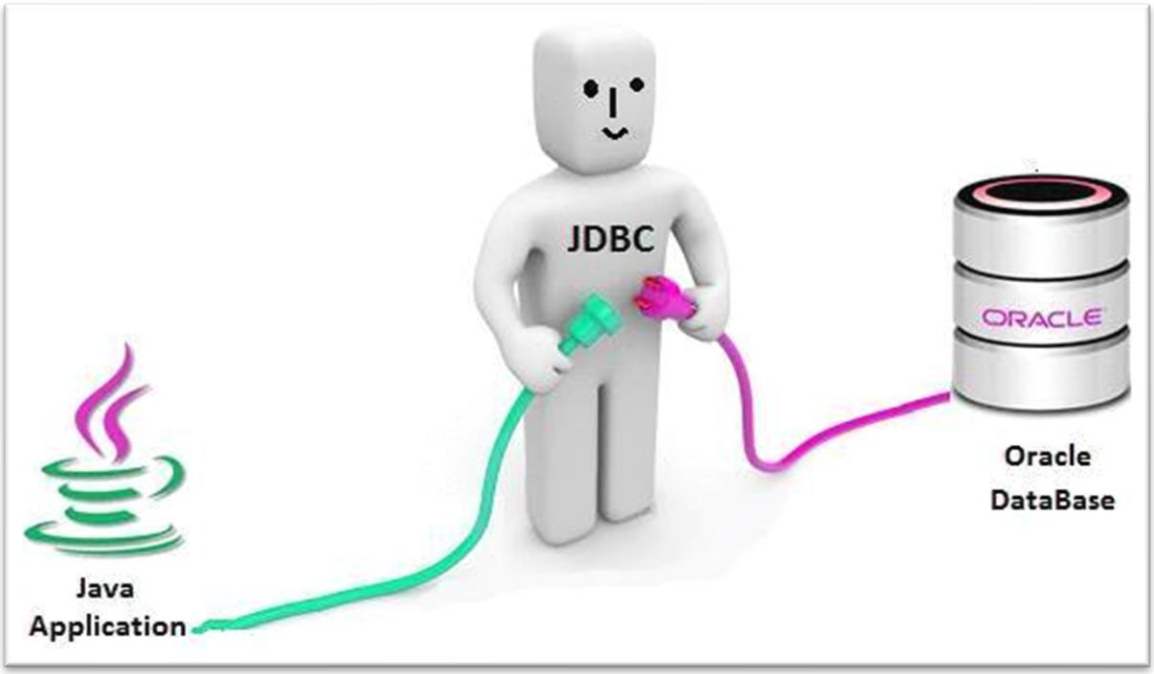
Execution of Servlets involves six basic steps:

1. The clients send the request to the web server.
2. The web server receives the request.
3. The web server passes the request to the corresponding servlet.
4. The servlet processes the request and generates the response in the form of output.
5. The servlet sends the response back to the web server.
6. The web server sends the response back to the client and the client browser displays it on the screen.

JDBC

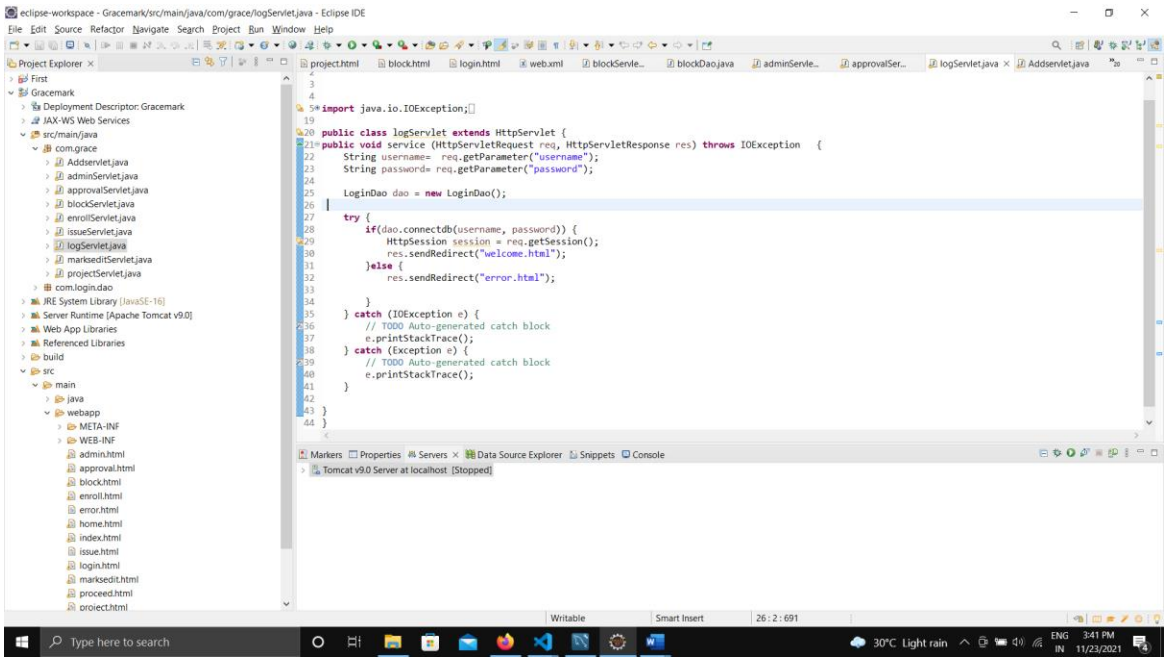
JDBC stands for Java Database Connectivity. JDBC is a Java API to connect and execute the query with the database. It is a part of Java SE (Java Standard Edition). JDBC API uses JDBC drivers to connect with the database.

We can use JDBC API to access tabular data stored in any relational database. With the help of JDBC API, we can save, update, delete and fetch data from the database.
We can use JDBC API to handle database using Java program and can perform the following activities:
1. Connect to the database
2. Execute queries and update statements to the database
3. Retrieve the result received from the database.

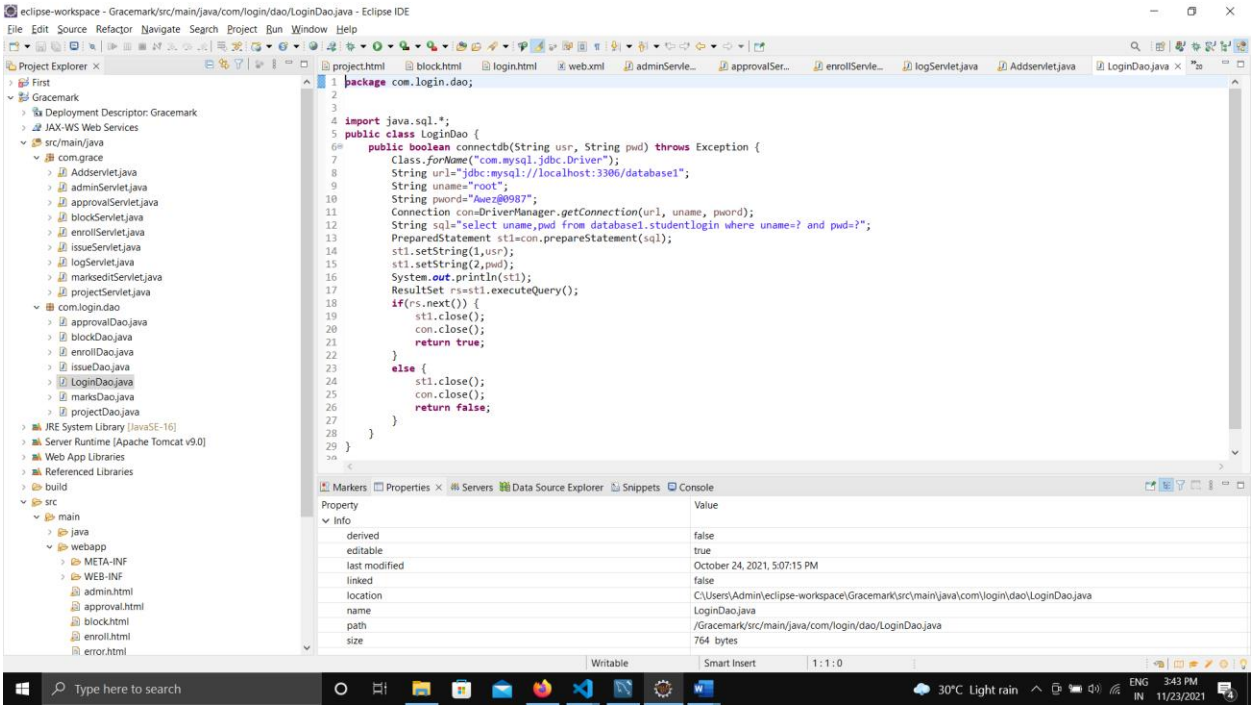


Implementation

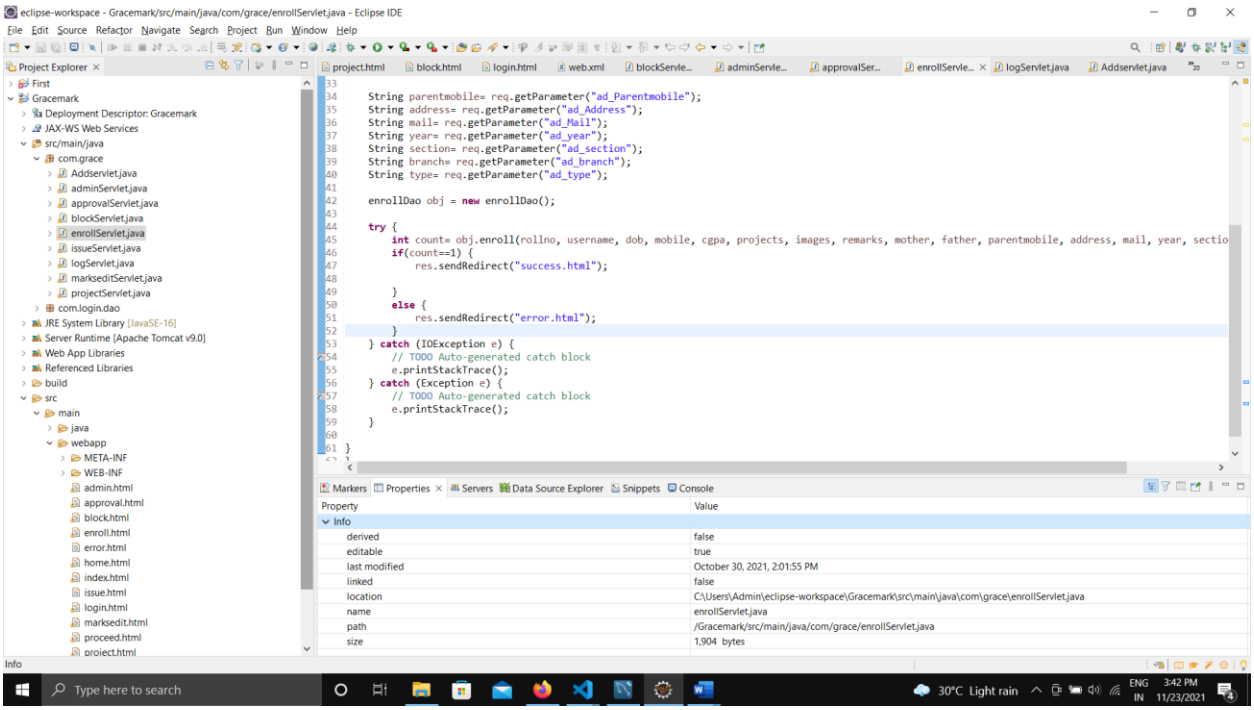
Login Servlet :



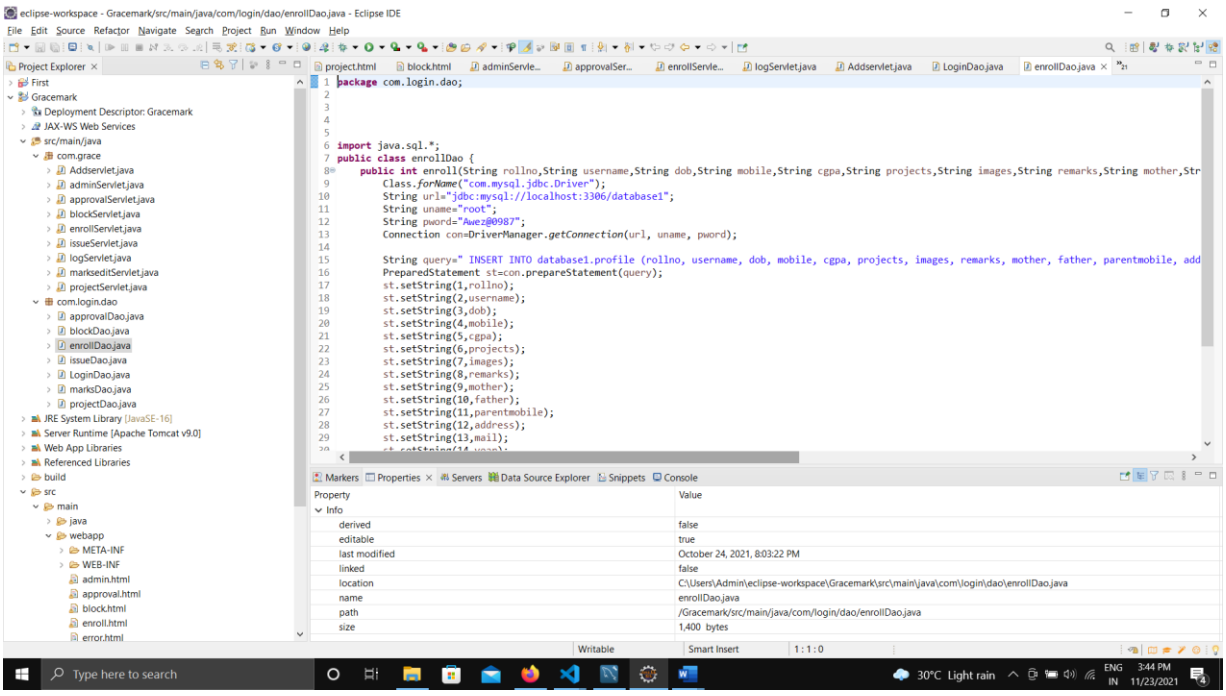
JDBC:



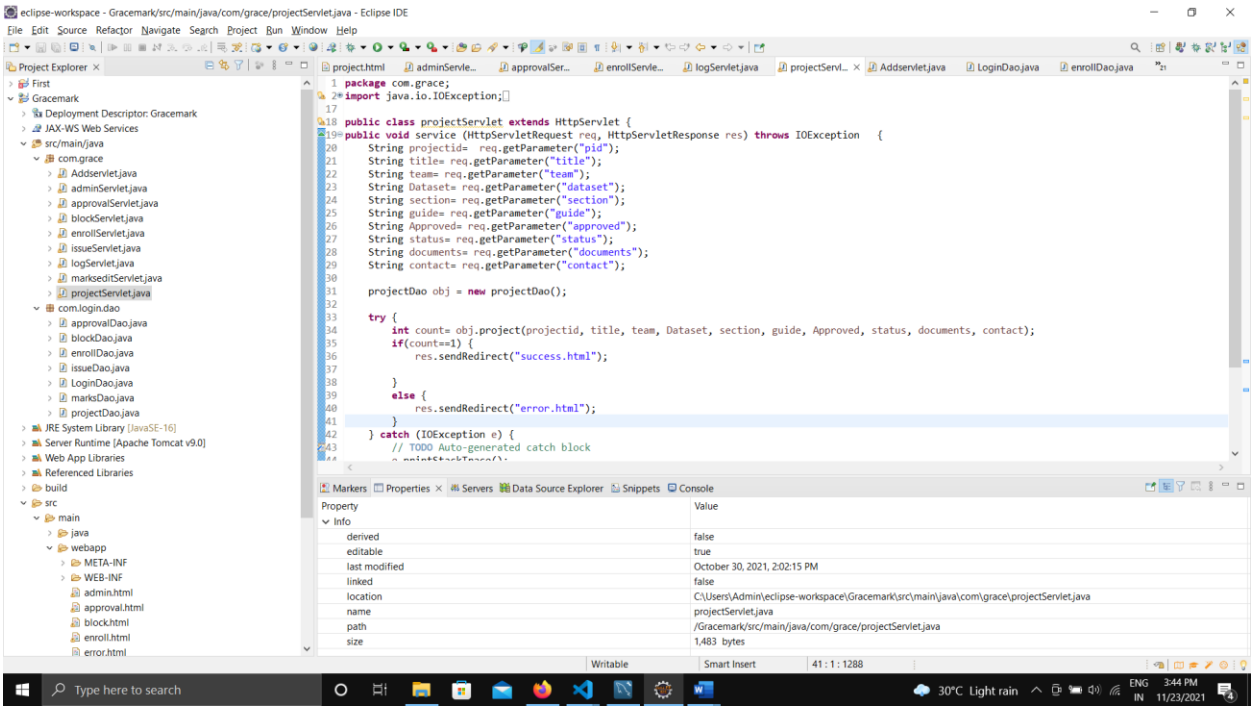
Enroll Servlet :



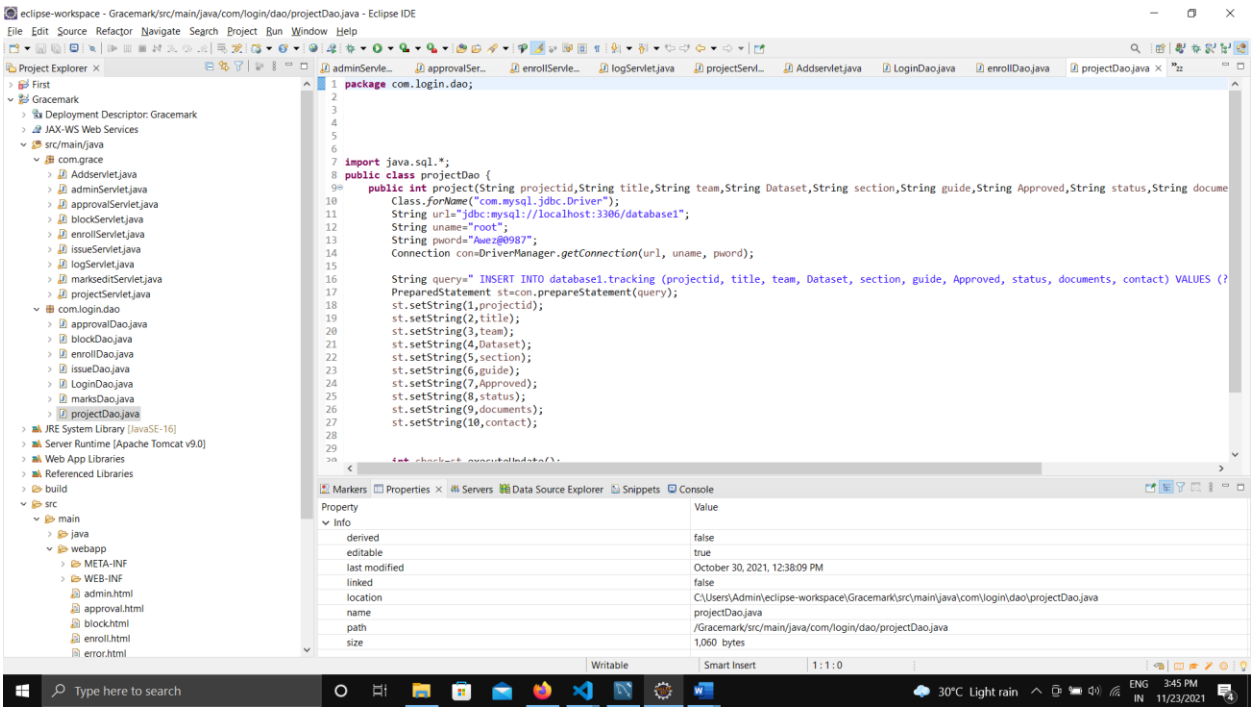
JDBC:



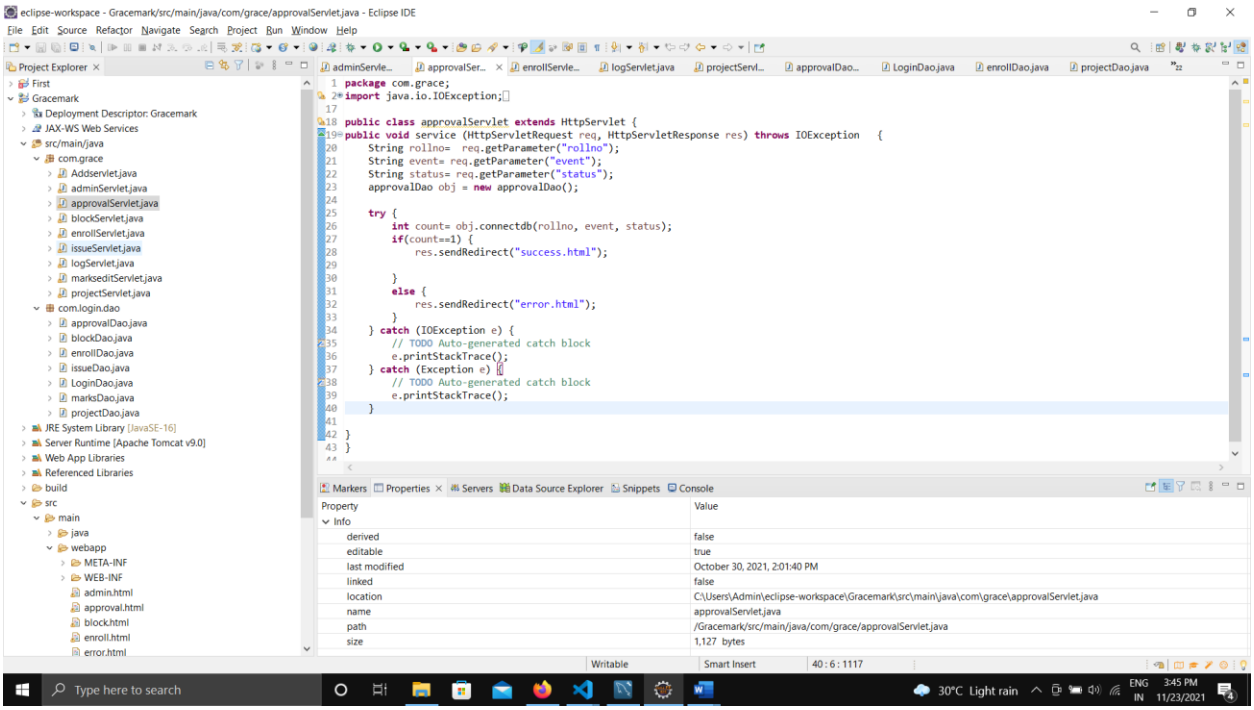
Project Servlet :



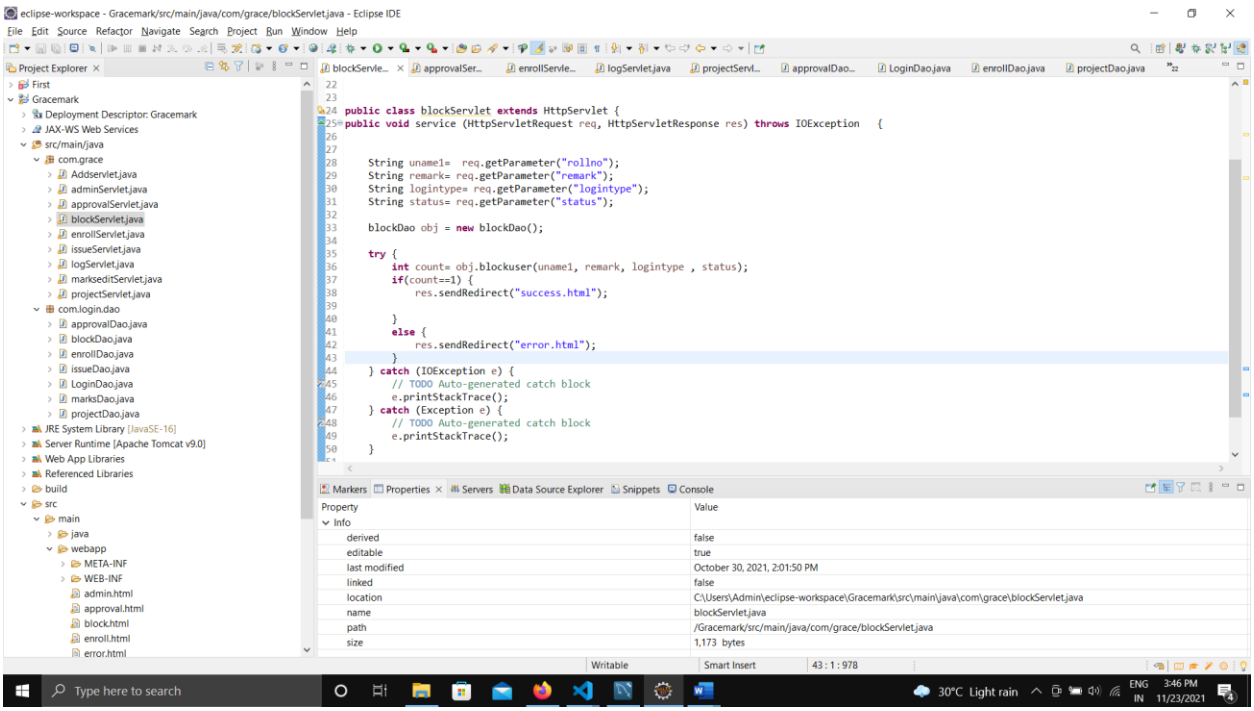
JDBC:



Approval



Block User



Active Login :

```
            <div class="container">
        <table class="table table-striped">
        <% activelogins.forEach(function (login) { %>
       <tr>
        <td><%=  login.username %>     </td>
        <td><%=  login.rollno %>     </td>
        <td> <%= login.mobile %>      </td>
        <td> <%= login.mail %>      </td>
        <td>  <img src="<%=login.images%>" width="150" alt="image not
available"></img> </td>
      </tr>

        <% }) %>
```

Attendance:

```
        <% attendance.forEach(function (students) { %>
      <tr>
          <th>  Course Code   </th>
          <th> Course Name </th>
          <th> Attendace in % </th>
      </tr>
      <tr>
          <td>15CSE376</td>
          <td>Net Centric Programming</td>
          <td ><%=students.NCP%> </td></tr>

          <td>15CSE401</td>
          <td>Machine Learning and Data Mining</td>
          <td ><%=students.MLDM%> </td></tr>

          <td>15CSE402</td>
          <td>Structure and Interpretation of Computer Programs</td>
          <td ><%=students.SICP%> </td></tr>
          <td>15CSE480</td>
          <td>Internet of Things</td>
          <td ><%=students.IOT%> </td></tr>
          <td>15CSE481</td>
          <td>Machine Learning and Data Mining Lab.</td>
          <td ><%=students.MLDMLAB%> </td></tr>
```

```
          <td>15CSE495</td>

          <td>Project Phase I</td>
          <td ><%=students.PP1%> </td></tr> <% }) %>
        </center>
```

Retriving Student Details

```
var obj1 = {};
app.set('views',path.join(__dirname,'views'));
app.set('view engine', 'ejs');
app.get('/students', function(req, res) {
fs.readFile('currentlogin.txt', 'utf-8', (err, data) => {
if (err) throw err;
mycon.query(`  SELECT * FROM database1.profile where type="student"; `,
function(err, result) {
        if(err){
        throw err;
        } else {
        obj1 = {students_details: result};
        res.render('students_details', obj1);
       // console.log(obj1);
        }
        });
});
});
```

Project Dashboard

```
app.get('/trackingdashboard', function(req, res) {
fs.readFile('currentlogin.txt', 'utf-8', (err, data) => {
if (err) throw err;
mycon.query(`  SELECT * FROM database1.tracking , profile where (profile.username
= tracking.guide) `, function(err, result) {
if(err){
throw err;
} else {
obj1 = {trackingdashboard: result};
res.render('trackingdashboard', obj1);
// console.log(obj1);
}
});
});
});
```

Faculty Details

```
app.get('/faculty_details', function(req, res) {
var obj2={}
mycon.query(` SELECT * FROM database1.profile where type="faculty"; `,
function(err, result) {
if(err){
throw err;
} else {
obj2 = {faculty_details: result};
res.render('faculty_details', obj2);
}});

});
```

Profile

```
fs.readFile('logintype.txt', 'utf-8', (err, data1) => {
fs.readFile('currentlogin.txt', 'utf-8', (err, data) => {
if (err) throw err;
mycon.query( `SELECT * FROM  profile where  ( profile.rollno = '${data}') ` ,
function(err, result) {

if(err){
throw err;
} else {
obj = {upload: result};

res.render('upload', obj);
}
});
});
})

})
```

Erasing Current user from database when logged out

```
app.get('/logout', function(req, res){
res.sendfile("logout.html");
fs.readFile('currentlogin.txt', 'utf-8', (err, data) => {
if (err) throw err;
mycon.query(  `truncate table database1.currentlogin `,function(err,result){
if(err) throw err;
// console.log(result);
});
```

Retriving Marks

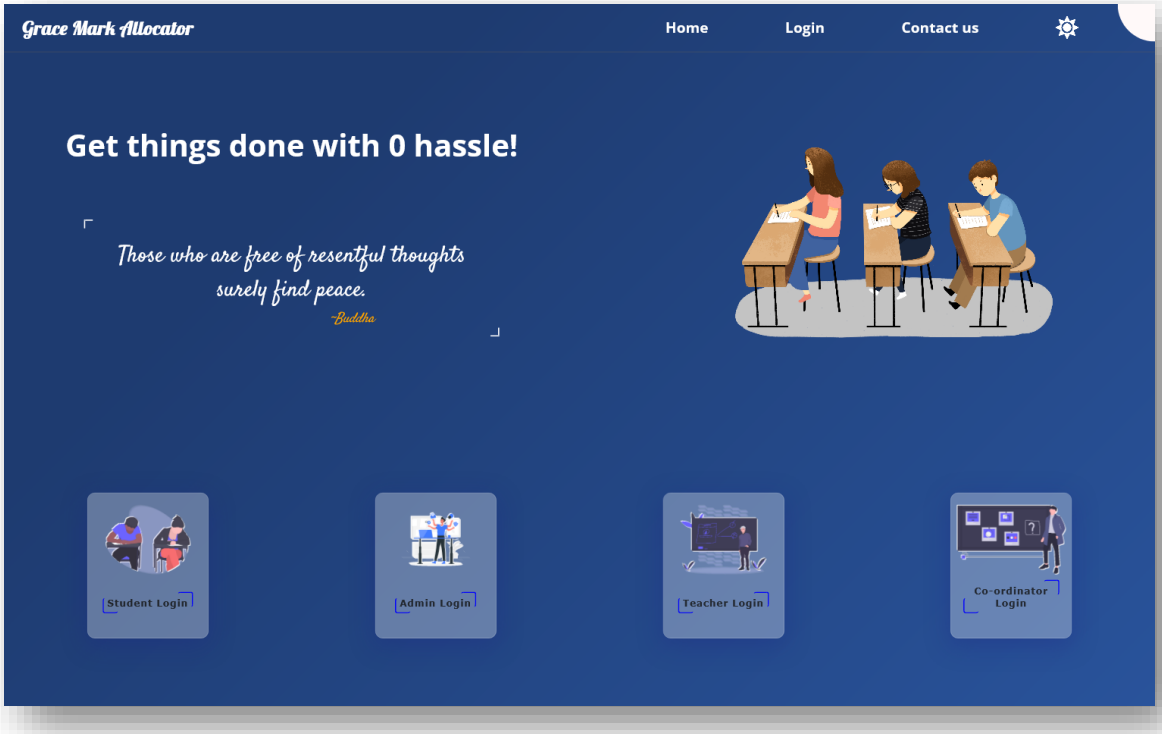
```
app.get('/selfmarks', function(req, res) {
var obj2={}
fs.readFile('currentlogin.txt', 'utf-8', (err, data) => {
if (err) throw err;
mycon.query(` SELECT * FROM database1.marks where rollno =  '${data}'; `,
function(err, result) {
if(err){
throw err;
} else {
obj2 = {marks: result};
res.render('marks', obj2);
}});
});
});
```

Uploading Documents Function

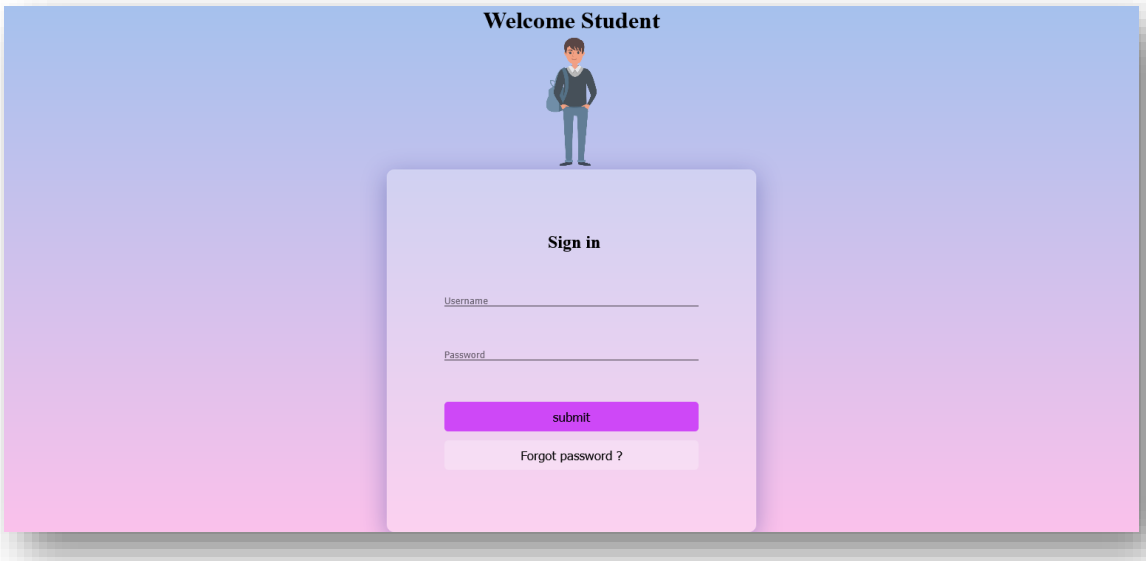
```
http.createServer(function (req, res) {
  if (req.url == '/fileupload') {
    var form = new formidable.IncomingForm();
    form.parse(req, function (err, fields, files) {
      var oldpath = files.filetoupload.path;
      var newpath = 'C:/Users/Admin/Desktop/Ncp/Project/Upload/' +
files.filetoupload.name;
      fs.rename(oldpath, newpath, function (err) {
        if (err) throw err;
        res.write('File uploaded and moved!');
        res.end();
      });
 });
  } else {
    res.writeHead(200, {'Content-Type': 'text/html'});
    res.write('<form action="fileupload" method="post" enctype="multipart/form-
data">');
    res.write('<input type="file" name="filetoupload"><br>');
    res.write('<input type="submit">');
    res.write('</form>');
    return res.end();
  }
}).listen(8080);
```

Output

Home page:



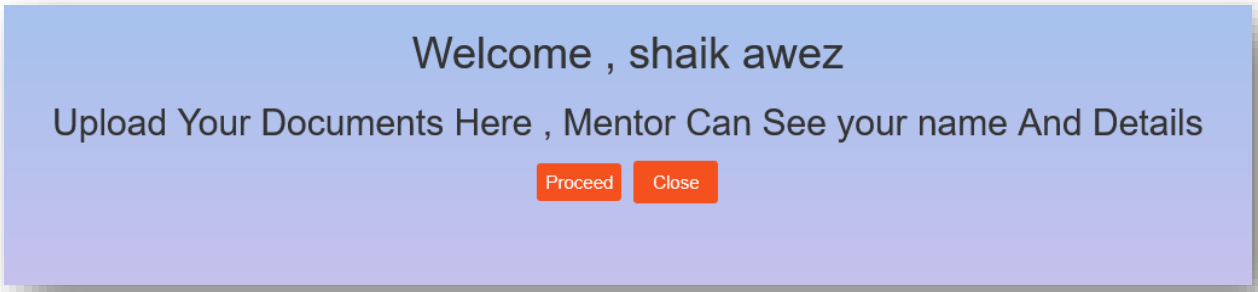
Login :



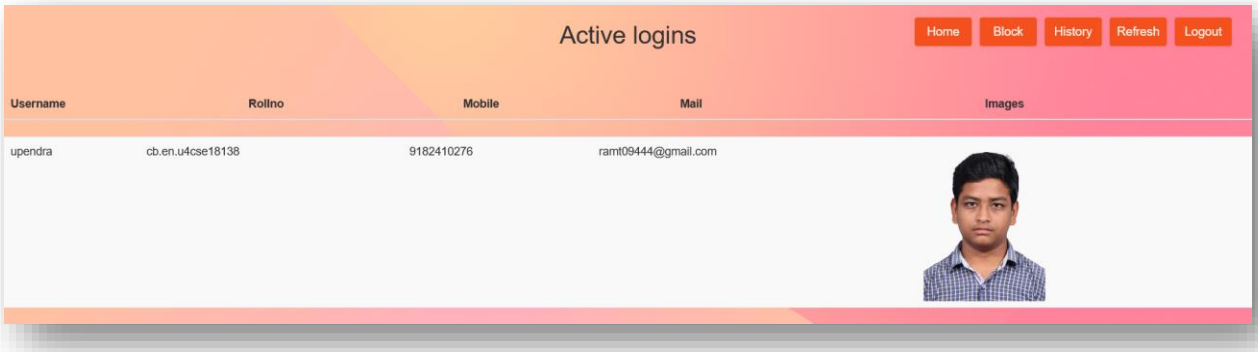
Welcome page:

Menu:

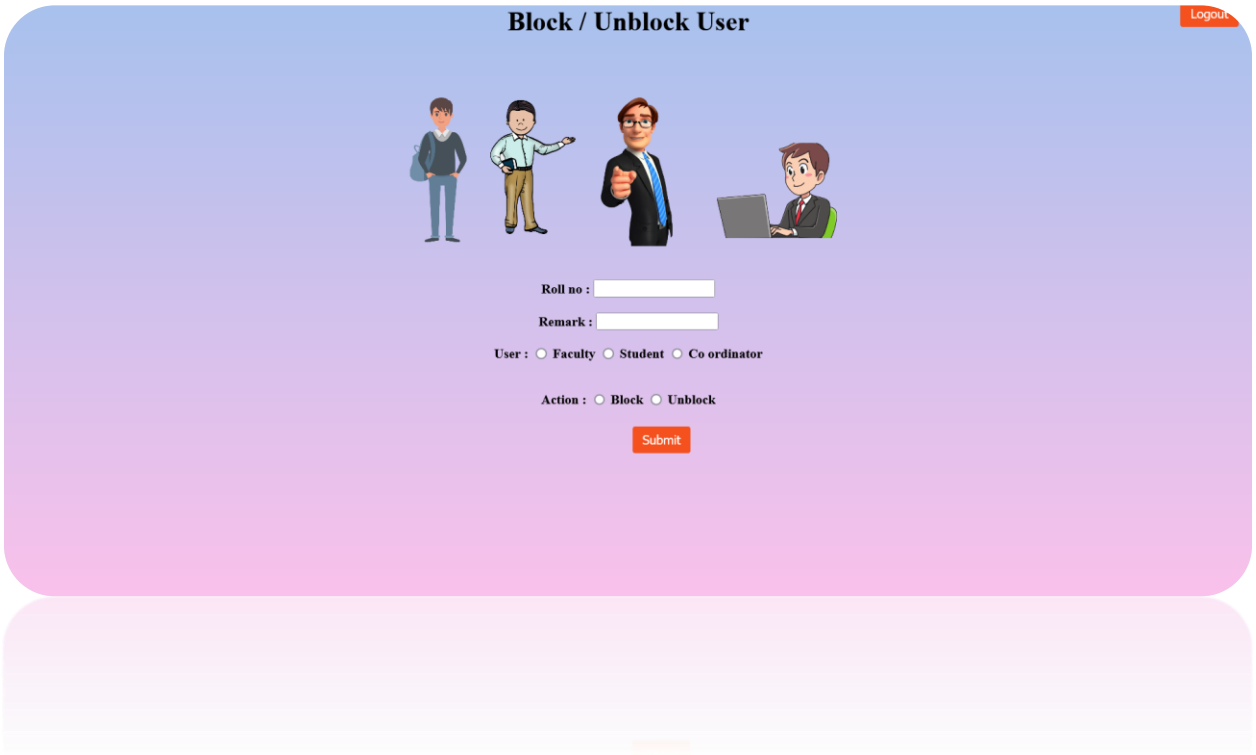
Doc Uploading:



Active logins:



User Block



Login History:

Students Data:

Students Data

Refresh Close

Roll no	Name	DOB	Mobile	CGPA	Mother	Father	Parent Mobile	Address	G-Mail ID	Photo
cb.en.u4cse18117	Lokesh	12/9/2000	8142838171	7.5	sudha rani	srinivasulu	9246477785	Narasaraopet,Guntur,AP	garlapatilokesh16@gmail.com	
cb.en.u4cse18136	pruthve	31-08-2001	9381044045	7	Raji	Ravi	9032702362	trvineni talkies ,Ananthpur ,AP	pruthverej1947@gmail.com	
cb.en.u4cse18137	pavan	24-10-2000	834190089	7.2	Geetha	Vinod	9440924291	Rajeev nagar extension,Tirupati,Andhra Pradesh	Kruthikpavan24@gmail.com	
cb.en.u4cse18138	upendra	20-10-2000	9182410276	7.3	devi	Sivaiah	9704108768	nrt , Guntur district Andhra Pradesh	ramt09444@gmail.com	
cb.en.u4cse18148	shaik awez	23-12-2000	6303731463	7	shaik Habeeb	shaik Jaleel	9441234780	Buchi reddy palem , Nellore,Ap	shaikjalil123@gmail.com	

Project tracking :

Tracking of Paper Publication

Project id	Tiltle	Team	Dataset	Section	Guide	Approved	Status	Supporting Documents	Contact	More Details
D231	Covid 19 Prediction Based On Comorbidities	1. Shaik Awez CB.EN.U4CSE18148 2. N pavan Kruthik CB.EN.U4CSE18137 3 P Pruthve Rej CB.EN.U4CSE18136 4 P Upendra CB.EN.U4CSE18138	Kaggale	CSE B	DR.Arun Kumar	Yes	Pre proceesing the data set	Click here	6303731463	Click Here

Close

More details on Project :

Abstract

A health crisis of massive proportion such as the current COVID-9 pandemic provides us with an op- portunity to ponder and reflect over what we can better in the way we deal with healthcare to make us humans be more prepared and enabled to combat such an event in the future. During the entire course of the pandemic, one of the main problems that healthcare providers have faced is the shortage of medical resources and a proper plan to efficiently distribute them. They have been in the dark failing to understand how much resource they could even in the very next week as the COVID-19 curve has swayed very un- predictably. In these tough times, being able to predict what kind of resource an individual might require at the time of being tested positive or even before that will be of great help to the authorities as they would be able to procure and arrange for the resources necessary to save the life of that patient

Problem Definition

To design a web application for predicting the Covid 19 virus using sampling techniques and Feature selection Methods . The Objective of the project is to design a web based application and to apply AlgorThe Objective of the project is to design a web based application and to apply AlgoThe Objective of the project is to design a web based application and to apply Algorithms to include an optimal sampling technique to accurately predict the virus

Introduction

A health crisis of massive proportion such as the current COVID-9 pandemic provides us with an op- portunity to ponder and reflect over what we can better in the way we deal with healthcare to make us humans be more prepared and enabled to combat such an event in the future. During the entire course of the pandemic, one of the main problems that healthcare providers have faced is the shortage of medical resources and a proper plan to efficiently distribute them. They have been in the dark failing to understand how much resource they could even in the very next week as the COVID-19 curve has swayed very un- predictably. In these tough times, being able to predict what kind of resource an individual might require at the time of being tested positive or even before that will be of great help to the authorities as they would be able to procure and arrange for the resources necessary to save the life of that patient

Guide

Roll No	COOD21
Name	DR.Arun Kumar
Alloted Sections	All
Branch	CSE,ECE
Graduation Year	2004
Mobile	1234568901
Projects	Prediction of cancer using customised fuzzy rough machine learning approaches ,Rough Set Based Variable Tolerance Attribute Selection on High Dimensional Microarray Imbalanced Data
Mail	c_arunkumar@gmail.com
Address	Coimbatore , Tamil Nadu

Website

Code

Faculty Welcome page:

Menu :

Validation

Evaluation sheet:

Roll No	Technology	Max Marks	Marks Awarded	Total (30)
18117	Business Logic Integration	10 10		
18136	Business Logic Integration	10 10		
18137	Business Logic Integration	10 10		
18138	Business Logic Integration	10 10		
18148	Business Logic Integration	10 10		
	Project Documentation	10		