FULL STACK WEB DEVELOPMENT

ASSIGNMENT-I

1602-21-733-064

K.Vishal Reddy

CSE-A

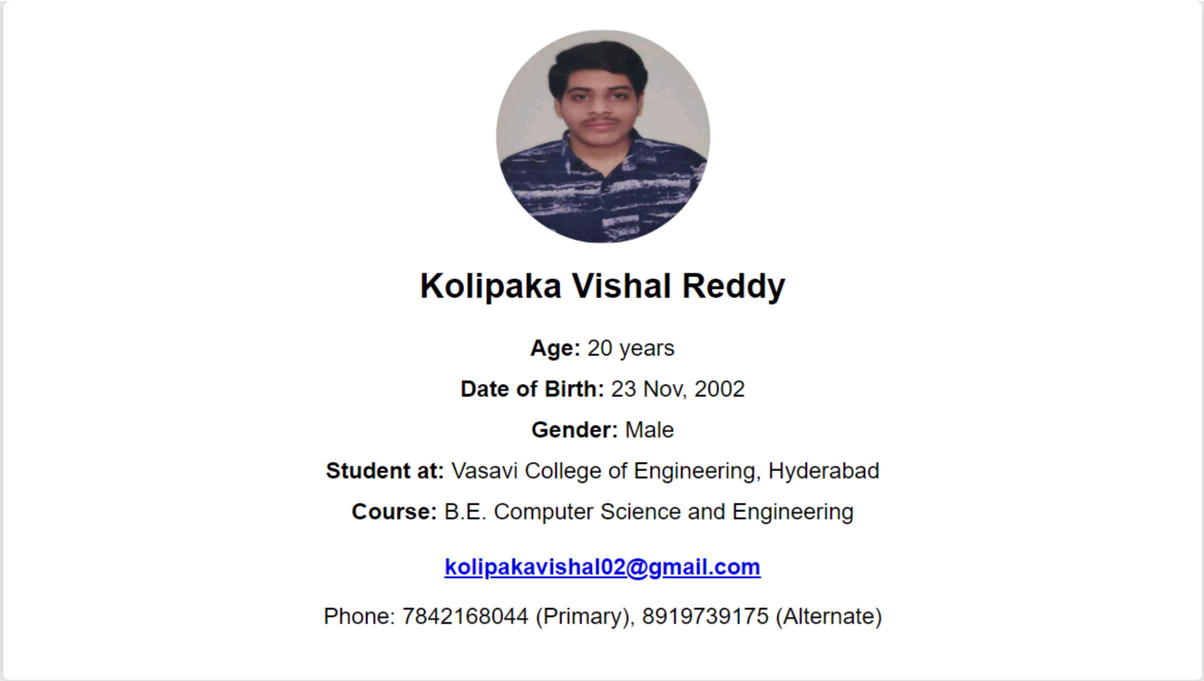
1. Every student irrespective of the group he belongs to is also required to create his/her own About Me page (Biodata) page with photo by applying appropriate CSS styles.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>About Me - Kolipaka Vishal Reddy</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-color: #f0f0f0;
      margin: 0;
      padding: 0;
    }
    .container {
      max-width: 800px;
      margin: 0 auto;
      padding: 20px;
      background-color: #fff;
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.2);
      border-radius: 5px;
    }
    h1 {   text-align: center;
      font-size: 24px;
    }
    img {display: block;
      margin: 0 auto;
      width: 150px;
      height: 150px;
      border-radius: 50%;
    }
    .personal-info {margin-top: 20px;
      text-align: center;
    }
    .info-item {
      margin-bottom: 10px;
    }
    .info-label { font-weight: bold; }
    .contact-info { text-align: center;
      margin-top: 20px;
    }
    .email { font-weight: bold;
      color: #007bff;
    }
  </style>
</head>
<body>
  <div class="container">
    <img src="My Pic.jpeg" alt="Kolipaka Vishal Reddy">
    <h1>Kolipaka Vishal Reddy</h1>
    <div class="personal-info">
      <div class="info-item">
        <span class="info-label">Age:</span> 20 years
      </div>
      <div class="info-item">
        <span class="info-label">Date of Birth:</span> 23 Nov, 2002
      </div>
      <div class="info-item">
```

```
          <span class="info-label">Gender:</span> Male
        </div>
        <div class="info-item">
          <span class="info-label">Student at:</span> Vasavi College of Engineering,
Hyderabad
        </div>
        <div class="info-item">
          <span class="info-label">Course:</span> B.E. Computer Science and
Engineering
        </div>
      </div>
      <div class="contact-info">
        <p class="email"><a
href="mailto:kolipakavishal02@gmail.com">kolipakavishal02@gmail.com</a></p>
        <p>Phone: 7842168044 (Primary), 8919739175 (Alternate)</p>
      </div>
    </div>
</body>
</html>
```

OUTPUT :



Kolipaka Vishal Reddy

Age: 20 years

Date of Birth: 23 Nov, 2002

Gender: Male

Student at: Vasavi College of Engineering, Hyderabad

Course: B.E. Computer Science and Engineering

kolipakavishal02@gmail.com

Phone: 7842168044 (Primary), 8919739175 (Alternate)

FULL STACK WEB DEVELOPMENT

2. Develop a web application which keeps track of data about students academic background and helps HR manager to generate eligibility list wrt a Company. There should be provision to capture the data of a candidate through multiple forms. Use the appropriate styling for creating rich interface and perform the necessary validation using JavaScript.

Code:

<>index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Student Registration</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <header><h1>Student Registration</h1></header>
  <nav>
    <ul>
      <li><a href="index.html">Dashboard</a></li>
      <li><a href="eligibility-list.html">Eligibility List</a></li>
      <li><a href="view-students.html">View Students</a></li>
      <li><a href="search-students.html">Search Students</a></li>
    </ul>
  </nav>
  <main>
    <form id="studentForm" onsubmit="saveStudentData()">
      <label for="rollNo">Roll Number:</label>
      <input type="text" id="rollNo" name="rollNo" required><br>
      <label for="name">Name:</label>
      <input type="text" id="name" name="name" required><br>
      <label for="cgpa">CGPA:</label>
      <input type="number" id="cgpa" name="cgpa" min="0" max="10" step="0.01" required><br>
      <label for="backlogs">Number of Backlogs:</label>
      <input type="number" id="backlogs" name="backlogs" min="0" required><br>
      <label for="branch">Branch:</label>
```

```
            <input type="text" id="branch" name="branch" required><br>
            <label for="languages">Programming Languages:</label>
            <input type="text" id="languages" name="languages" required><br>
            <input type="submit" value="Register">
        </form>
    </main>
    <script src="script.js"></script>
</body>
</html>
```

<>view-students.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>View Students</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <header> <h1>View Students</h1></header>
    <nav>
        <ul>
            <li><a href="index.html">Dashboard</a></li>
            <li><a href="eligibility-list.html">Eligibility List</a></li>
            <li><a href="view-students.html">View Students</a></li>
            <li><a href="search-students.html">Search Students</a></li>
        </ul>
    </nav>
    <main>
        <table id="studentsTable">
            <thead>
                <tr>
                    <th>Roll Number&emsp;</th>
                    <th>Name&emsp;</th>
                    <th>CGPA&emsp;</th>
                    <th>Backlogs&emsp;</th>
                    <th>Branch&emsp;</th>
                    <th>Programming Languages</th>
                </tr>
            </thead>
            <tbody id=studentsTableBody></tbody>
        </table>
    </main>
    <script src="script.js"></script>
</body>
</html>
```

<>edit.students.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Student Registration</title>
    <link rel="stylesheet" href="style.css">
  </head>
  <body>
    <header><h1>Student Registration</h1></header>
    <nav>
      <ul>
        <li><a href="index.html">Dashboard</a></li>
        <li><a href="eligibility-list.html">Eligibility List</a></li>
        <li><a href="view-students.html">View Students</a></li>
        <li><a href="search-students.html">Search Students</a></li>
      </ul>
    </nav>
    <main>
      <form id="editStudentForm" onsubmit="updateStudentData(); return false;">
        <label for="editName">Name:</label>
        <input type="text" id="editName" required /><br />
        <label for="editCgpa">CGPA:</label>
        <input type="number" step="0.01" id="editCgpa" required /><br />
        <label for="editBacklogs">Backlogs:</label>
    <input type="number" id="editBacklogs" required /><br />
    <label for="editBranch">Branch:</label>
    <input type="text" id="editBranch" required /><br />
    <label for="editLanguages">Programming Languages:</label>
    <input type="text" id="editLanguages" required /><br />
    <button type="submit">Update</button>
   </form>
  </main>
  <script src="script.js"></script>
 </body>
</html>
```

<>search-students.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Search Students</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <header><h1>Search Students</h1></header>
  <nav>
    <ul>
      <li><a href="index.html">Dashboard</a></li>
      <li><a href="eligibility-list.html">Eligibility List</a></li>
      <li><a href="view-students.html">View Students</a></li>
      <li><a href="search-students.html">Search Students</a></li>
    </ul>
  </nav>
  <main>
    <form id="searchForm">
      <label for="rollNumber">Enter Roll Number:</label>
      <input type="text" id="rollNumber" required>
      <button type="submit">Search</button>
    </form>
    <div id="searchResults"></div>
  </main>
  <script src="script.js"></script>
</body>
</html>
```

<>eligibility-list.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Eligibility List</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <header><h1>Eligibility List</h1></header>
  <nav>
    <ul>
     <li><a href="index.html">Dashboard</a></li>
     <li><a href="eligibility-list.html">Eligibility List</a></li>
     <li><a href="view-students.html">View Students</a></li>
     <li><a href="search-students.html">Search Students</a></li>
    </ul>
   </nav>
  <main>
    <label for="minimumCgpa">Minimum CGPA:</label>
    <input type="number" step="0.01" id="minimumCgpa" required>
    <button onclick="checkEligibility()">Check Eligibility</button>
    <div id="eligibleStudentsContainer"></div>
  </main>
  <script src="script.js"></script>
</body>
</html>
```

#style.css

```
body, h1, h2, h3, p {
 margin: 0;
```

```
  padding: 0;
}
body {
  background-color: #f4f4f4;
  color: #333;
  font-family: Arial, sans-serif;
}
header {
  background-color: #0074e4;
  text-align: center;
  color: #fff;
  padding: 10px;
}
nav ul {
  list-style-type: none;
  display: flex;
  justify-content: center;
  background-color: #333;
}
nav li {
  margin: 0 10px;
}
nav a {
  text-decoration: none;
  color: #fff;
  font-weight: bold;
}
main {
  text-align: center;
  padding: 20px;
}
form {
  background-color: #fff;
  padding: 20px;
  border-radius: 5px;
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.2);
```

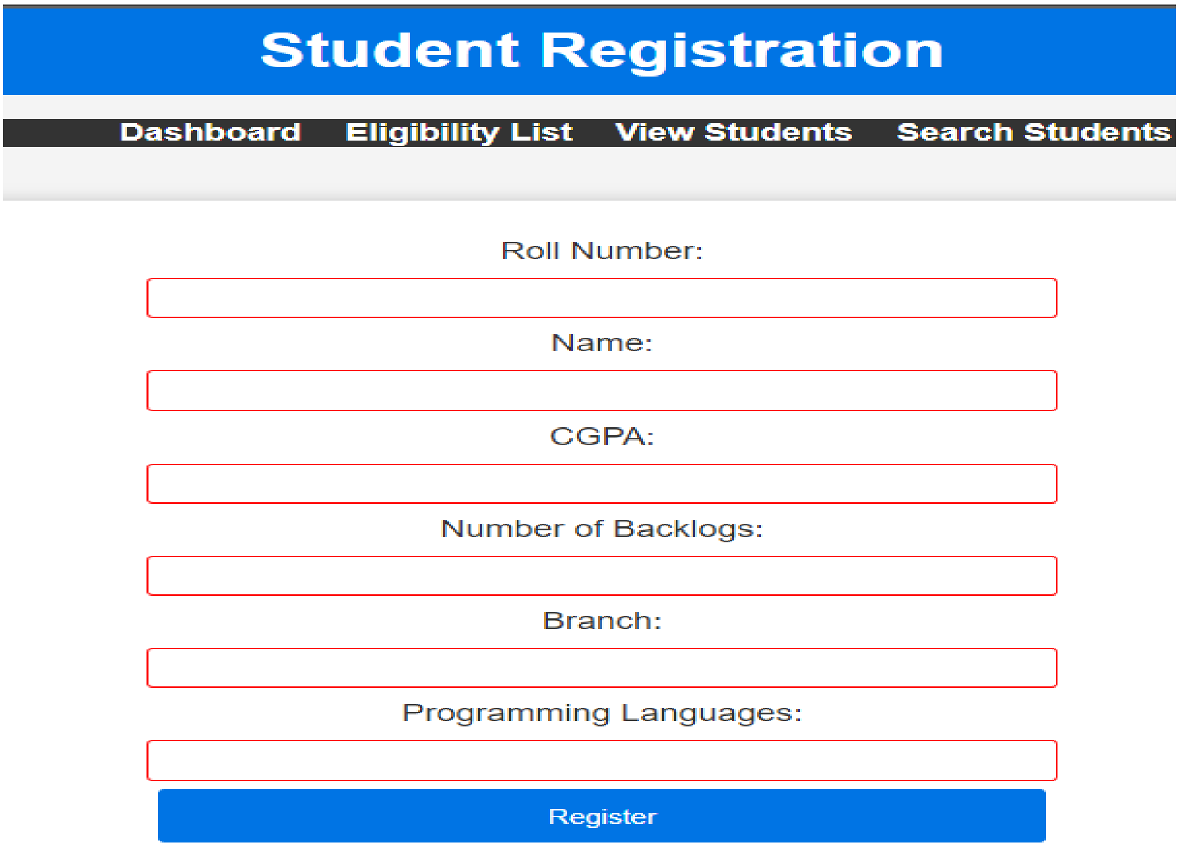
```
}
form label {
  display: block;
  margin: 5px 0;
}
form input {
  width: 25%;
  padding: 5px;
  margin: 3px 0;
  border: 1px solid #ccc;
  border-radius: 3px;
}
form input[type="submit"] {
  background-color: #0074e4;
  color: #fff;
  border: none;
  padding: 10px 20px;
  cursor: pointer;
  border-radius: 3px;
}
form input:invalid {
  border: 1px solid #f00;
}
form input[type="submit"]:hover {
  background-color: #0051a0;
}
table {
  width: 50%;
  border-collapse: collapse;
  margin: 20px auto;
}
table, th, td {
  border: 1px solid #ddd;
}
th, td {
  padding: 10px;
  text-align: left;
}
#searchResults{
  background-color: #fff;
  padding: 20px;
  border-radius: 5px;
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.2);
  margin: 20px auto;
}
#minimumCgpa{
  width: 25%;
  padding: 10px;
  margin: 5px 0;
  border: 1px solid #ccc;
  border-radius: 3px;
}
```

<script.js>

```
// Function to clear student data from local storage on homepage (student-registration.html) load
function clearStudentData() {
    localStorage.removeItem("students");
}
// Function to save student data to local storage (student-registration.html)
function saveStudentData() {
    const rollNo = document.getElementById("rollNo").value;
    const name = document.getElementById("name").value;
    const cgpa = document.getElementById("cgpa").value;
    const backlogs = document.getElementById("backlogs").value;
    const branch = document.getElementById("branch").value;
    const languages = document.getElementById("languages").value;
    const student = {
        rollNo: rollNo,
        name: name,
        cgpa: cgpa,
        backlogs: backlogs,
        branch: branch,
        languages: languages
    };
    let students = JSON.parse(localStorage.getItem("students")) || [];
    students.push(student);
    localStorage.setItem("students",JSON.stringify(students));
    alert("Student data saved successfully!");
    return false;
}
// Function to retrieve and display students on view-students.html
function displayStudents() {
    const studentsTableBody = document.getElementById("studentsTableBody");
    const students = JSON.parse(localStorage.getItem("students")) || [];
    studentsTableBody.innerHTML = ""; // Clear the existing table content
    students.forEach((student, index) => {
        const row = document.createElement("tr");
        row.innerHTML = `
            <td>${student.rollNo}</td>
            <td>${student.name}</td>
            <td>${student.cgpa}</td>
            <td>${student.backlogs}</td>
            <td>${student.branch}</td>
            <td>${student.languages}</td>
            <td>
                <button onclick="editStudent('${student.rollNo}')">Edit</button>
                <button onclick="deleteStudent('${student.rollNo}')">Delete</button>
            </td>`;
        studentsTableBody.appendChild(row);
    });
}
// Function to delete student data (updated to use rollNo as a string)
function deleteStudent(rollNo) {
    const students = JSON.parse(localStorage.getItem("students")) || [];
    const studentIndex = students.findIndex(student => student.rollNo === rollNo);
    if (studentIndex !== -1) {
        students.splice(studentIndex, 1);
        localStorage.setItem("students", JSON.stringify(students));
        displayStudents();
    } else {
        alert("Student not found.");
    }
}
// Call the displayStudents function when the page loads
document.addEventListener("DOMContentLoaded", function () {
    displayStudents();
});
function editStudent(rollNo) {
    window.location.href = `edit-student.html?rollNo=${rollNo}`;
}
// Function to load and pre-fill student details for editing
function loadStudentForEditing() {
    const urlParams = new URLSearchParams(window.location.search);
    const rollNo = urlParams.get("rollNo");
    const isDirect = urlParams.get("direct");
    if (isDirect === 'true') {
        alert("Access denied. Please use the 'Edit' button from 'view-students.html'.");
        window.location.href = 'view-students.html'; // Redirect back to view-students.html
        return;
    }
    const students = JSON.parse(localStorage.getItem("students")) || [];
    const student = students.find(student => student.rollNo === rollNo);
    if (student) {
        document.getElementById("editName").value = student.name;
        document.getElementById("editCgpa").value = student.cgpa;
        document.getElementById("editBacklogs").value = student.backlogs;
        document.getElementById("editBranch").value = student.branch;
        document.getElementById("editLanguages").value = student.languages;
    } else {
        alert("Student not found.");
    }
}
// Function to update student data and redirect back to view-students.html
function updateStudentData() {
    const urlParams = new URLSearchParams(window.location.search);
    const rollNo = urlParams.get("rollNo");
    const students = JSON.parse(localStorage.getItem("students")) || [];
    const studentIndex = students.findIndex(student => student.rollNo === rollNo);
    if (studentIndex !== -1) {
        students[studentIndex].name = document.getElementById("editName").value;
        students[studentIndex].cgpa = document.getElementById("editCgpa").value;
        students[studentIndex].backlogs = document.getElementById("editBacklogs").value;
        students[studentIndex].branch = document.getElementById("editBranch").value;
        students[studentIndex].languages = document.getElementById("editLanguages").value;
        localStorage.setItem("students", JSON.stringify(students));
        window.location.href = "view-students.html";
    } else {
        alert("Student not found.");
    }
}
// Call the loadStudentForEditing function when the edit-student.html page loads
document.addEventListener("DOMContentLoaded", function () {
    loadStudentForEditing();
});
// Function to handle student search
function searchStudent() {
    const rollNumber = document.getElementById("rollNumber").value;
    const students = JSON.parse(localStorage.getItem("students")) || [];
    const foundStudent = students.find(student => student.rollNo === rollNumber);
    const searchResults = document.getElementById("searchResults");
    if (foundStudent) {
        searchResults.innerHTML = `
            <h2>Search Results</h2>
            <p><strong>Roll Number:</strong> ${foundStudent.rollNo}</p>
            <p><strong>Name:</strong> ${foundStudent.name}</p>
            <p><strong>CGPA:</strong> ${foundStudent.cgpa}</p>
            <p><strong>Backlogs:</strong> ${foundStudent.backlogs}</p>
            <p><strong>Branch:</strong> ${foundStudent.branch}</p>
            <p><strong>Programming Languages:</strong> ${foundStudent.languages}</p>`;
    } else {
        searchResults.innerHTML = "Student not found.";
    }
}
// Add event listener for form submission
const searchForm = document.getElementById("searchForm");
searchForm.addEventListener("submit", function (e) {
    e.preventDefault();
    searchStudent();
});
// Function to load student data from local storage
function loadStudentData() {
    return JSON.parse(localStorage.getItem("students")) || [];
}
function checkEligibility() {
    const minimumCgpa = parseFloat(document.getElementById("minimumCgpa").value);
    const students = loadStudentData(); // Load student data
    const eligibleStudents = students.filter(student => {
        return parseFloat(student.cgpa) >= minimumCgpa;
    });
    displayEligibleStudents(eligibleStudents);
}
function displayEligibleStudents(eligibleStudents) {
    const eligibleStudentsContainer = document.getElementById("eligibleStudentsContainer");
    eligibleStudentsContainer.innerHTML = ""; // Clear previous data
    if (eligibleStudents.length > 0) {
        const table = document.createElement("table");
        table.innerHTML = `
            <thead>
                <tr>
                    <th>Roll Number</th>
                    <th>Name</th>
                    <th>CGPA</th>
                    <th>Backlogs</th>
                    <th>Branch</th>
                    <th>Programming Languages</th>
                </tr>
            </thead>
            <tbody>
            </tbody>`;
        eligibleStudents.forEach(student => {
            const row = document.createElement("tr");
            row.innerHTML = `
                <td>${student.rollNo}</td>
                <td>${student.name}</td>
                <td>${student.cgpa}</td>
                <td>${student.backlogs}</td>
                <td>${student.branch}</td>
                <td>${student.languages}</td>`;
            table.querySelector("tbody").appendChild(row);
        });
        eligibleStudentsContainer.appendChild(table);
    } else {
        eligibleStudentsContainer.textContent = "No eligible students found.";
    }
}
```

OUTPUT :

FULL STACK WEB DEVELOPMENT



Roll Number:

Name:

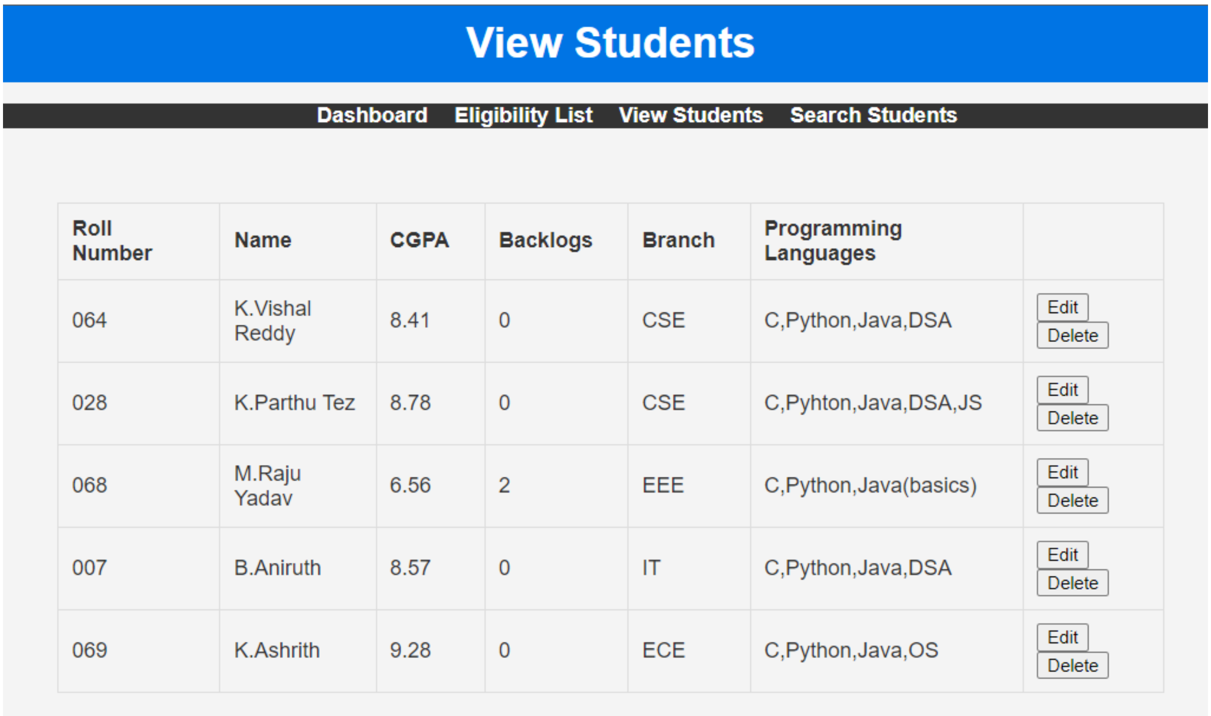
CGPA:

Number of Backlogs:

Branch:

Programming Languages:

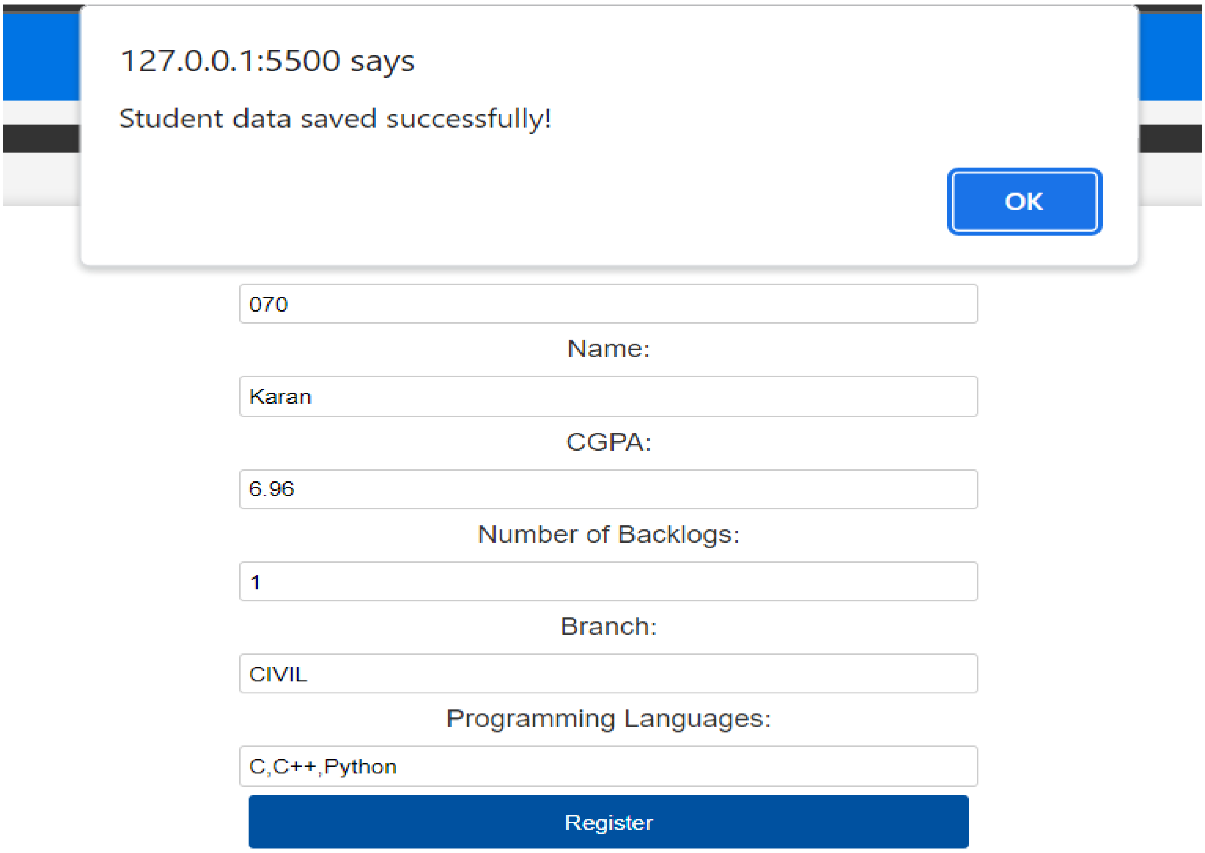
Already existed Student details



Roll Number	Name	CGPA	Backlogs	Branch	Programming Languages	
064	K.Vishal Reddy	8.41	0	CSE	C,Python,Java,DSA	Edit Delete
028	K.Parthu Tez	8.78	0	CSE	C,Pyhton,Java,DSA,JS	Edit Delete
068	M.Raju Yadav	6.56	2	EEE	C,Python,Java(basics)	Edit Delete
007	B.Aniruth	8.57	0	IT	C,Python,Java,DSA	Edit Delete
069	K.Ashrith	9.28	0	ECE	C,Python,Java,OS	Edit Delete

We can add new student details from dashboard

FULL STACK WEB DEVELOPMENT



127.0.0.1:5500 says

Student data saved successfully!

070

Name:

Karan

CGPA:

6.96

Number of Backlogs:

1

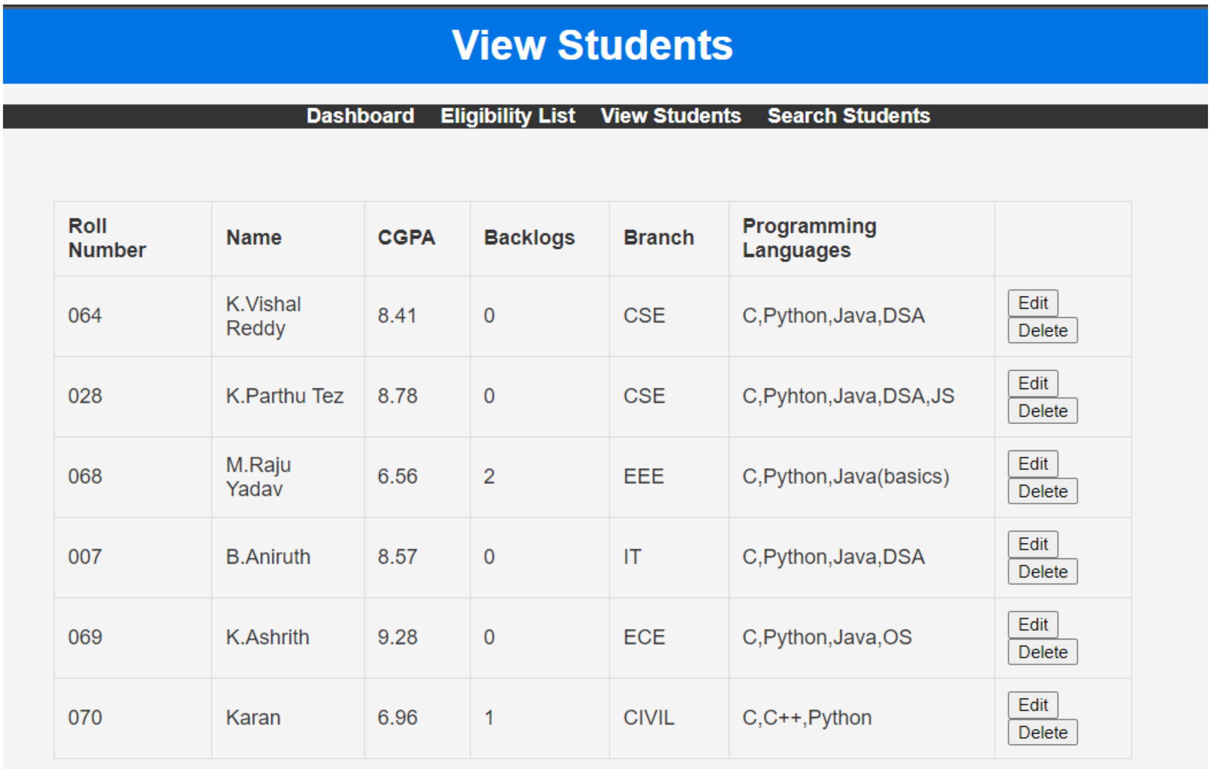
Branch:

CIVIL

Programming Languages:

C,C++,Python

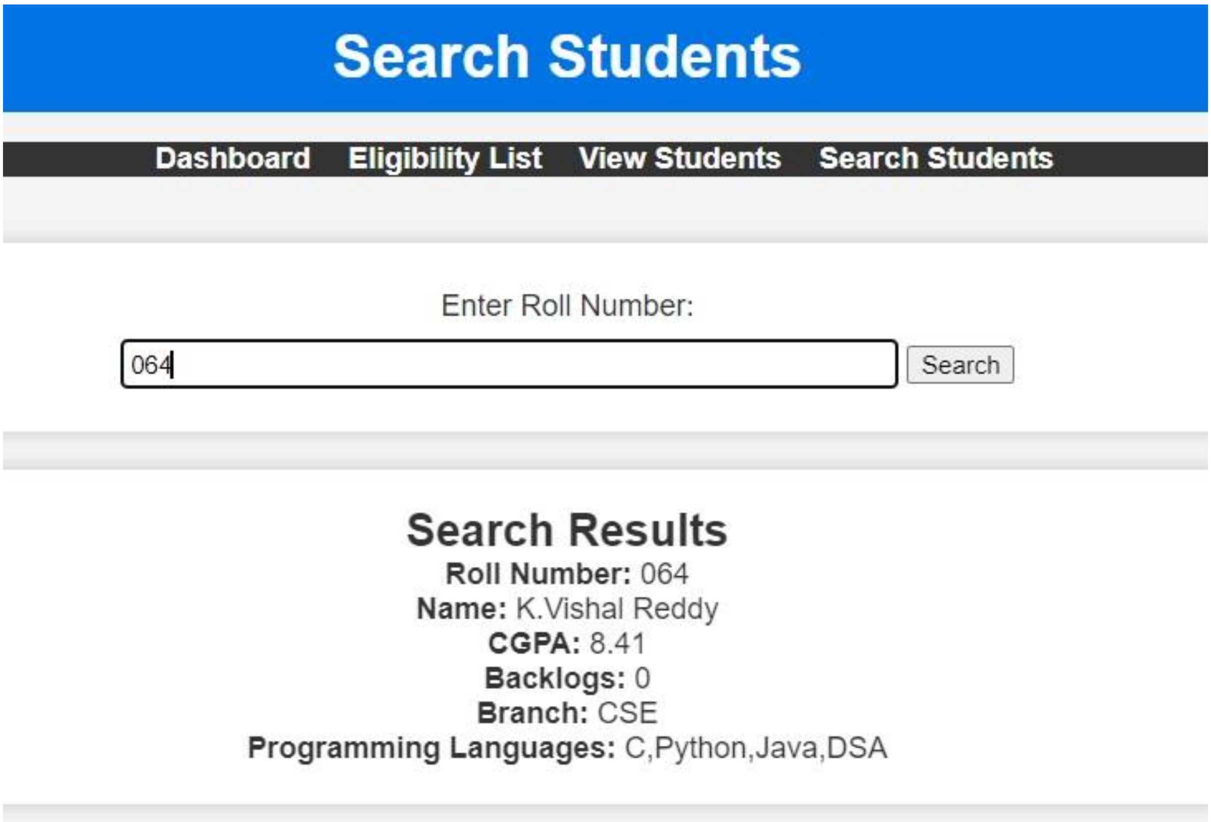
After adding new student details(updated)



View Students

Roll Number	Name	CGPA	Backlogs	Branch	Programming Languages	
064	K.Vishal Reddy	8.41	0	CSE	C,Python,Java,DSA	Edit Delete
028	K.Parthu Tez	8.78	0	CSE	C,Pyhton,Java,DSA,JS	Edit Delete
068	M.Raju Yadav	6.56	2	EEE	C,Python,Java(basics)	Edit Delete
007	B.Aniruth	8.57	0	IT	C,Python,Java,DSA	Edit Delete
069	K.Ashrith	9.28	0	ECE	C,Python,Java,OS	Edit Delete
070	Karan	6.96	1	CIVIL	C,C++,Python	Edit Delete

We can search the students through their roll nos

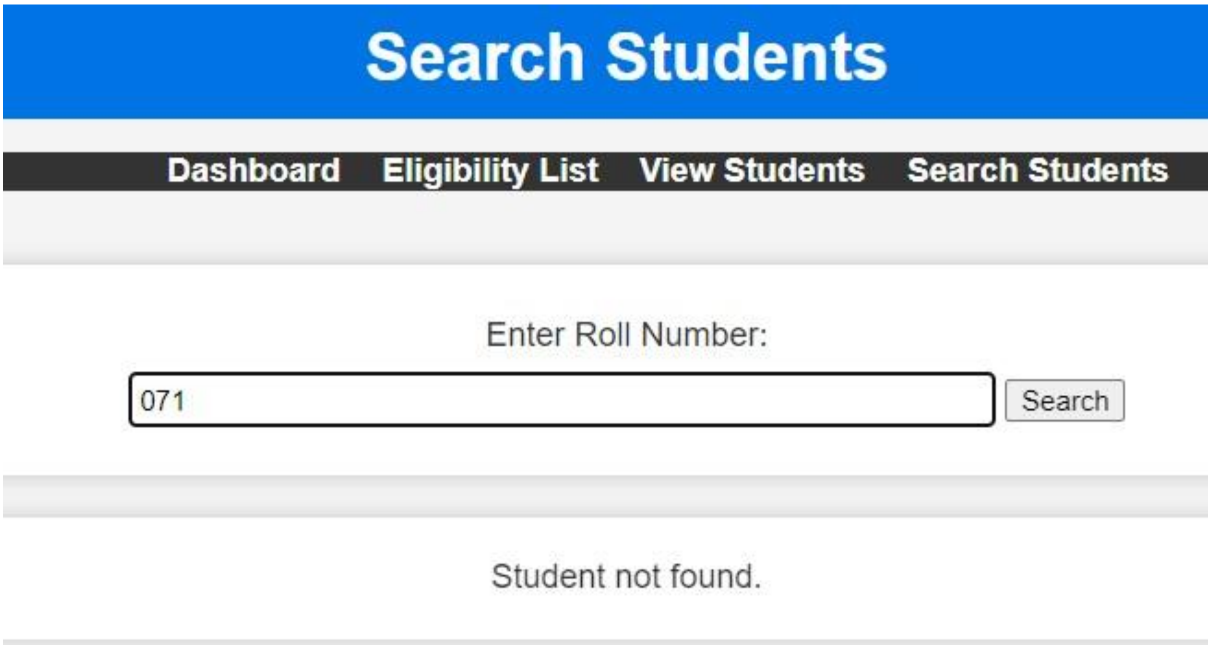


Enter Roll Number: 064

Search Results

Roll Number: 064
Name: K.Vishal Reddy
CGPA: 8.41
Backlogs: 0
Branch: CSE
Programming Languages: C,Python,Java,DSA

If the student is not found it

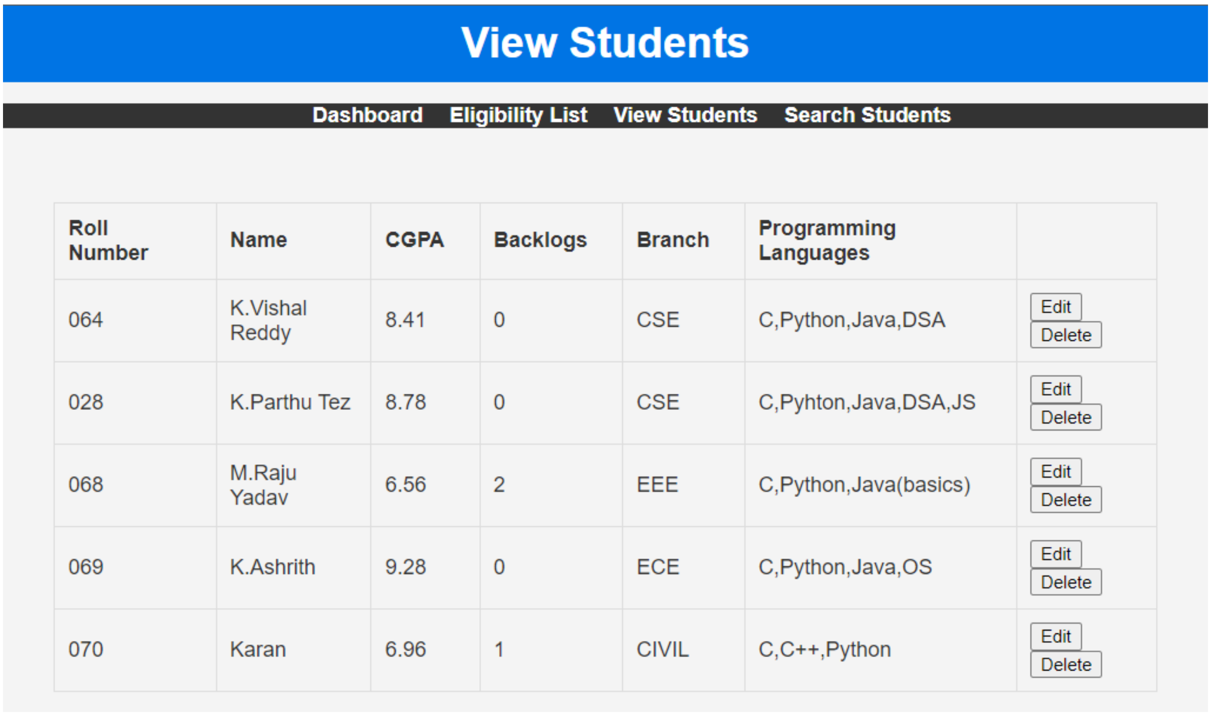


Enter Roll Number: 071

Student not found.

After deleting student with roll no 007(B.Aniruth)

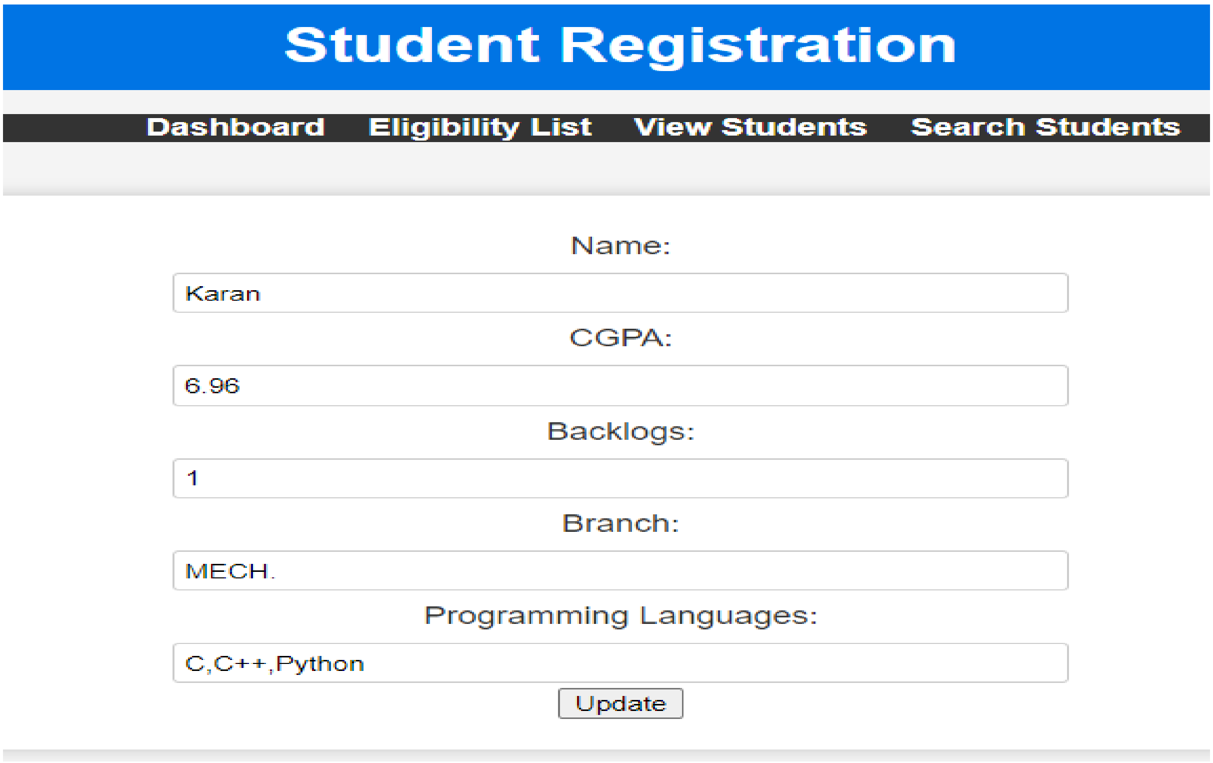
FULL STACK WEB DEVELOPMENT



View Students

Roll Number	Name	CGPA	Backlogs	Branch	Programming Languages	
064	K.Vishal Reddy	8.41	0	CSE	C,Python,Java,DSA	Edit Delete
028	K.Parthu Tez	8.78	0	CSE	C,Pyhton,Java,DSA,JS	Edit Delete
068	M.Raju Yadav	6.56	2	EEE	C,Python,Java(basics)	Edit Delete
069	K.Ashrith	9.28	0	ECE	C,Python,Java,OS	Edit Delete
070	Karan	6.96	1	CIVIL	C,C++,Python	Edit Delete

Pressing edit on particular student we can make changes and update.



Student Registration

Name:

Karan

CGPA:

6.96

Backlogs:

1

Branch:

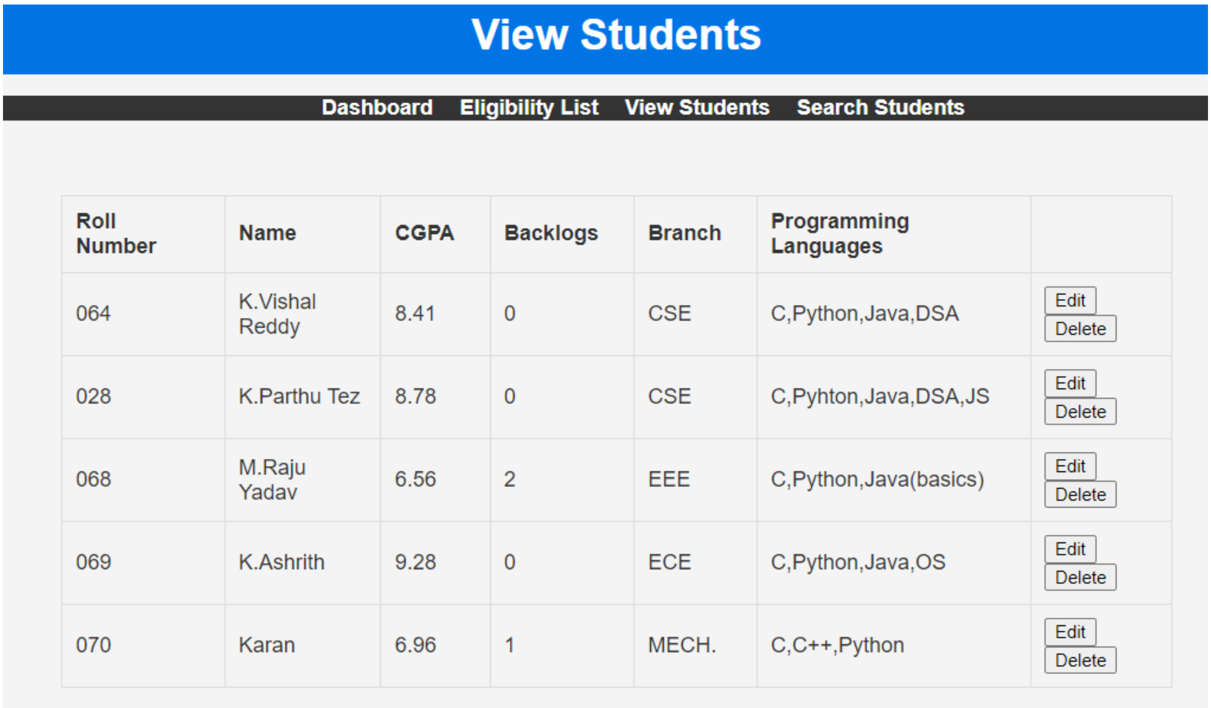
MECH.

Programming Languages:

C,C++,Python

Student Karan’s branch is changed from civil to mech.

FULL STACK WEB DEVELOPMENT

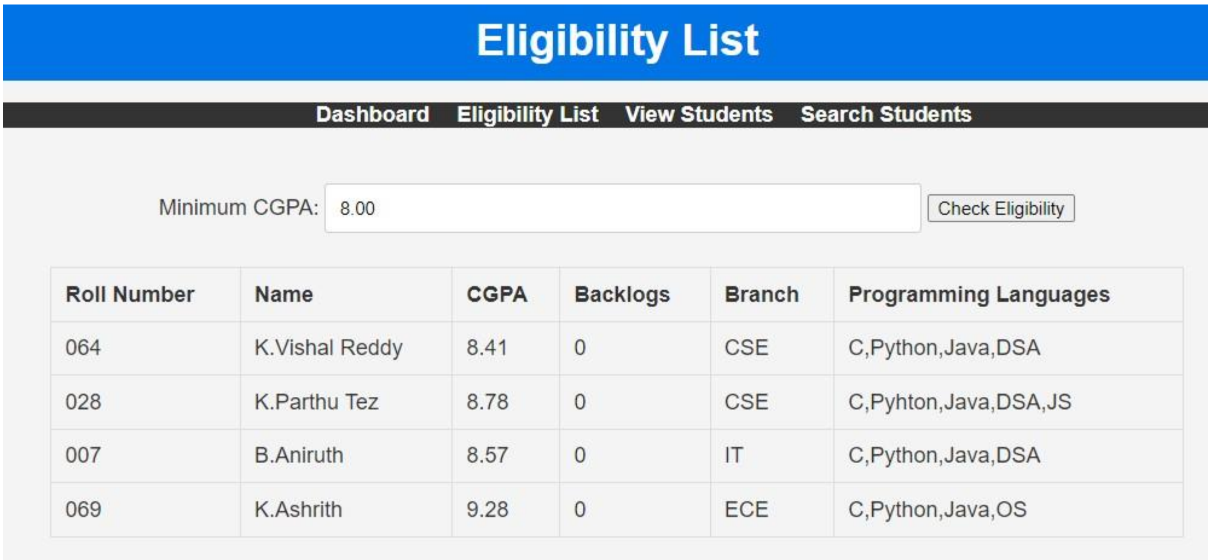


View Students

Roll Number	Name	CGPA	Backlogs	Branch	Programming Languages	
064	K.Vishal Reddy	8.41	0	CSE	C,Python,Java,DSA	Edit Delete
028	K.Parthu Tez	8.78	0	CSE	C,Pyhton,Java,DSA,JS	Edit Delete
068	M.Raju Yadav	6.56	2	EEE	C,Python,Java(basics)	Edit Delete
069	K.Ashrith	9.28	0	ECE	C,Python,Java,OS	Edit Delete
070	Karan	6.96	1	MECH.	C,C++,Python	Edit Delete

Min CGPA requied to hire by a company is entered here.

Here it displays the eligible students details as per criteria.



Eligibility List

Minimum CGPA: 8.00

Roll Number	Name	CGPA	Backlogs	Branch	Programming Languages
064	K.Vishal Reddy	8.41	0	CSE	C,Python,Java,DSA
028	K.Parthu Tez	8.78	0	CSE	C,Pyhton,Java,DSA,JS
007	B.Aniruth	8.57	0	IT	C,Python,Java,DSA
069	K.Ashrith	9.28	0	ECE	C,Python,Java,OS