

**CAREER MANAGEMENT & PLACEMENT SERVICE PORTAL**

**MINI PROJECT REPORT**

**18CSC311J – WEB DESIGN AND DEVELOPMENT**

**LABORATORY**

**(2018 Regulation)**

**III Year/ VI Semester**

**Academic Year: 2022 -2023**

**By**

**DURGA CHANDANA SREE MANDAPATI (RA2011028010099)**

**Under the guidance of**

**Dr.K.Deepa Thilak**

**Associate Professor**

**Department of Networking and Communications**



**DEPARTMENT OF NETWORKING AND COMMUNICATIONS**

**FACULTY OF ENGINEERING AND TECHNOLOGY**

**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY**

**Kattankulathur, Kancheepuram**

**MAY 2023**

## **BONAFIDE**

Certified that this Mini project report titled “**CAREER MANAGEMENT & PLACEMENT SERVICE PORTAL**” for the course **18CSC311J – WEB DESIGN AND DEVELOPMENT LABORATORY** is the bonafide work of **DURGA CHANDANA SREE MANDAPATI RA2011028010099** who undertook the task of completing the project within the allotted time.

### **Signature**

Dr. K. Deepa Thilak

### **Course Faculty**

Associate Professor  
Department of NWC

### **Signature**

Dr. Annapurani Panaiyappan K

### **Head of the Department**

Professor  
Department of NWC

## **Abstract**

The Career Management and Service Portal is a comprehensive platform designed to assist job seekers in advancing their careers. The portal offers users a login/signup page, directing them to a dashboard with multiple modules, including interview prep tips, aptitude, and verbal quizzes, and a vast range of resources such as ebooks, PPTs, and YouTube references. The portal also offers resume references to help job seekers create a professional and compelling resume.

The interview prep tips module provides users with comprehensive guidance on preparing for job interviews, covering areas such as appropriate dress, common interview questions, and making a good first impression. The aptitude and verbal quizzes modules help users practice and improve their skills in numerical reasoning, logical reasoning, and verbal reasoning. The resources module includes ebooks, PPTs, and YouTube references covering various career-related topics such as resume writing, cover letter writing, and job search strategies. Finally, the resume references module offers templates, sample resumes, and tips on how to tailor a resume to specific job requirements.

The primary aim of the Career Management and Service Portal is to streamline the job search process and provide job seekers with the necessary tools and resources to succeed in their job search endeavors.

## **Table of Contents**

<b>Chapter No.</b>	<b>Chapter Name</b>	<b>Page Number</b>
	Abstract	<b>3</b>
<b>1</b>	Introduction	<b>5</b>
<b>2</b>	System Architecture	<b>7</b>
<b>3</b>	Web Site Front End Implementation	<b>12</b>
3.1	Tools used	<b>12</b>
3.2	Web Page Design	<b>12-43</b>
3.3	Screenshots	<b>43-48</b>
<b>4</b>	Web Site Backend Implementation	<b>49</b>
4.1	Tools used	<b>49</b>
4.2	Code for backend	<b>50-55</b>
<b>5</b>	Database Design	<b>57</b>
5.1	Database used	
5.2	Database Schema and Tables Used in the Project	
<b>6</b>	Conclusion and Future Work	<b>61</b>

# **CHAPTER 1**

## **INTRODUCTION**

### **1.1 Existing Web Applications**

Several similar platforms already exist that offer job seekers tools and resources to enhance their career prospects. These platforms provide valuable services such as job listings, resume writing services, and interview preparation tips. Here are a few examples of existing works in the same domain as our Career Management and Service Portal:

- **LinkedIn Learning:** This is a platform that offers online courses and video tutorials on career-related topics such as job search strategies, resume writing and interviewing skills. LinkedIn Learning provides users with access to a vast library of courses and tutorials that are led by industry experts.
- **Glassdoor:** Glassdoor is a job search and company review platform that offers job seekers access to millions of job listings and company reviews. Glassdoor also provides users with interview questions and reviews to help them prepare for job interviews.
- **Indeed:** This popular job search platform aggregates job listings from multiple sources and allows users to search and apply for jobs directly on the platform. Indeed provides job seekers with a range of tools and resources, including resume-building tools and career advice.
- Other websites are Greeksforgreeks, Tutorialspoint, Interviewbit, Kahoot, Duolingo, and Prepinsta.

### **1.2 Drawbacks of existing application**

- Contains more advertisements.
- No single website covers quizzes, games, or resources.
- Need to navigate different sites for preparation.
- Limited interaction prevails as many websites need login to access the resources.
- No login/signup page which helps provide security to the user.
- Yet times few quiz applications take a long time to generate responses.
- While LinkedIn Learning offers a wide range of courses and video tutorials on career-related topics, the platform requires a paid subscription to access all content. Additionally, the courses and tutorials may not be tailored to the

specific needs of each user, and may not provide as comprehensive guidance on interview preparation, aptitude and verbal reasoning, and other career-related topics as a dedicated platform like the Career Management and Service Portal.

- Glassdoor provides users with valuable company reviews and interview questions, but the platform may not have as comprehensive job listings or as many career-related resources as other job search platforms. Additionally, some users may find the review system on Glassdoor to be biased or unreliable.

### **1.3 Project Objectives**

The main objective of the Career Management and Service Portal is to provide job seekers with a comprehensive platform that offers a range of tools and resources to enhance their career prospects. The project aims to address some of the limitations of existing job search platforms by providing job seekers with a centralized platform that offers a wide range of services, including job listings, resume writing services, interview preparation tips, and resources on aptitude and verbal reasoning.

Specifically, the project aims to achieve the following objectives:

1. To develop a user-friendly login/signup page that allows job seekers to register and create an account on the platform.
2. To create a dashboard for registered users that includes modules for interview preparation tips, aptitude and verbal reasoning quizzes, and resources such as e-books, PPTs, and YouTube references.
3. To provide job seekers with access to a range of job listings from multiple sources, including company websites and job search engines.
4. To offer resume writing services that help job seekers create professional and customized resumes that stand out to potential employers.
5. To provide users with a range of resources and tools that help them prepare for job interviews, including sample interview questions and tips on how to answer them effectively.
6. To offer resources and tools that help job seekers improve their aptitude and verbal reasoning skills, such as practice quizzes and tutorials.
7. To continually update the platform with the latest job listings and resources to ensure that users have access to the most relevant and up-to-date information.

Overall, the Career Management and Service Portal aims to provide job seekers with a comprehensive platform that meets all their career-related needs and helps them achieve their career goals.

## CHAPTER 2

### SYSTEM ARCHITECTURE

#### 2.1 System Architecture

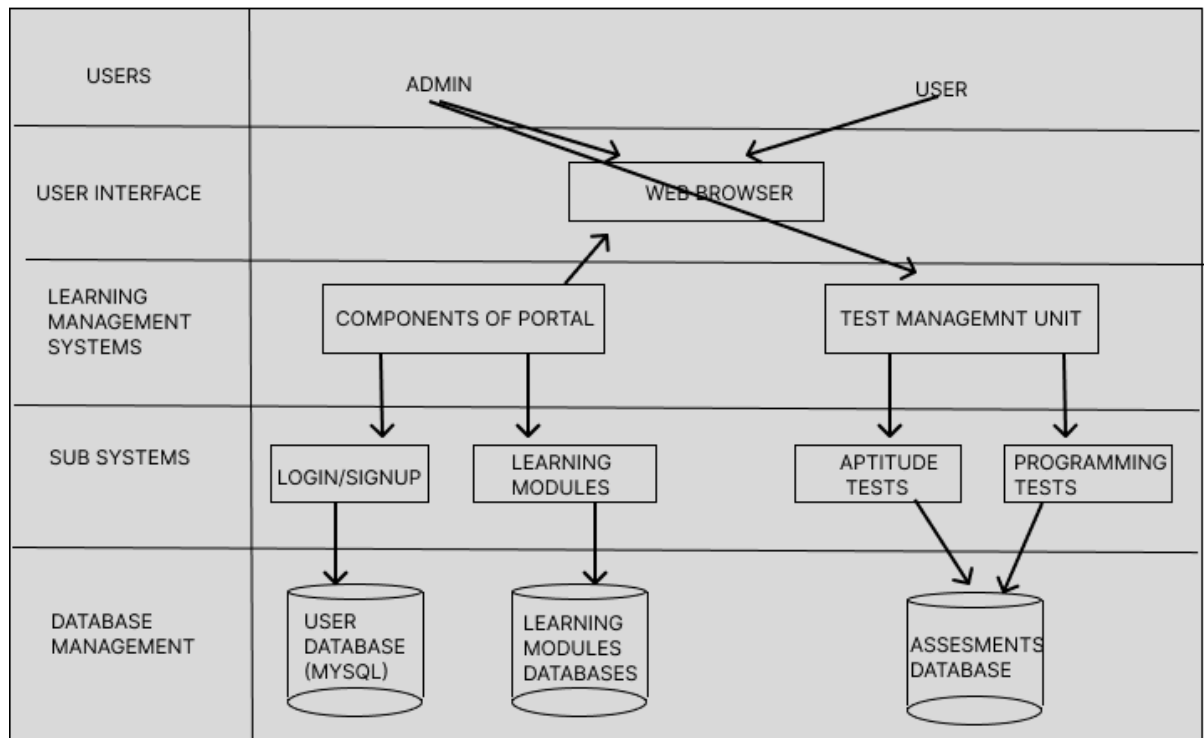


Figure-2.1 System Architecture Diagram for CMPS Portal

**User & User Interface:** This component includes the design and layout of the web application, which provides users with an interface to interact with the system. The user interface should be intuitive, user-friendly, and accessible to all users.

**Authentication and Authorization:** This component is responsible for authenticating users and authorizing their access to different parts of the system-login/signup. This component should ensure that only authorized users have access to sensitive data and functions.

**Learning management system:** This component enables users to set their career goals, develop a career plan, and track their progress toward their goals. It may also include assessments and resources for users to explore various career options.

## 2.2 Use Case Diagram

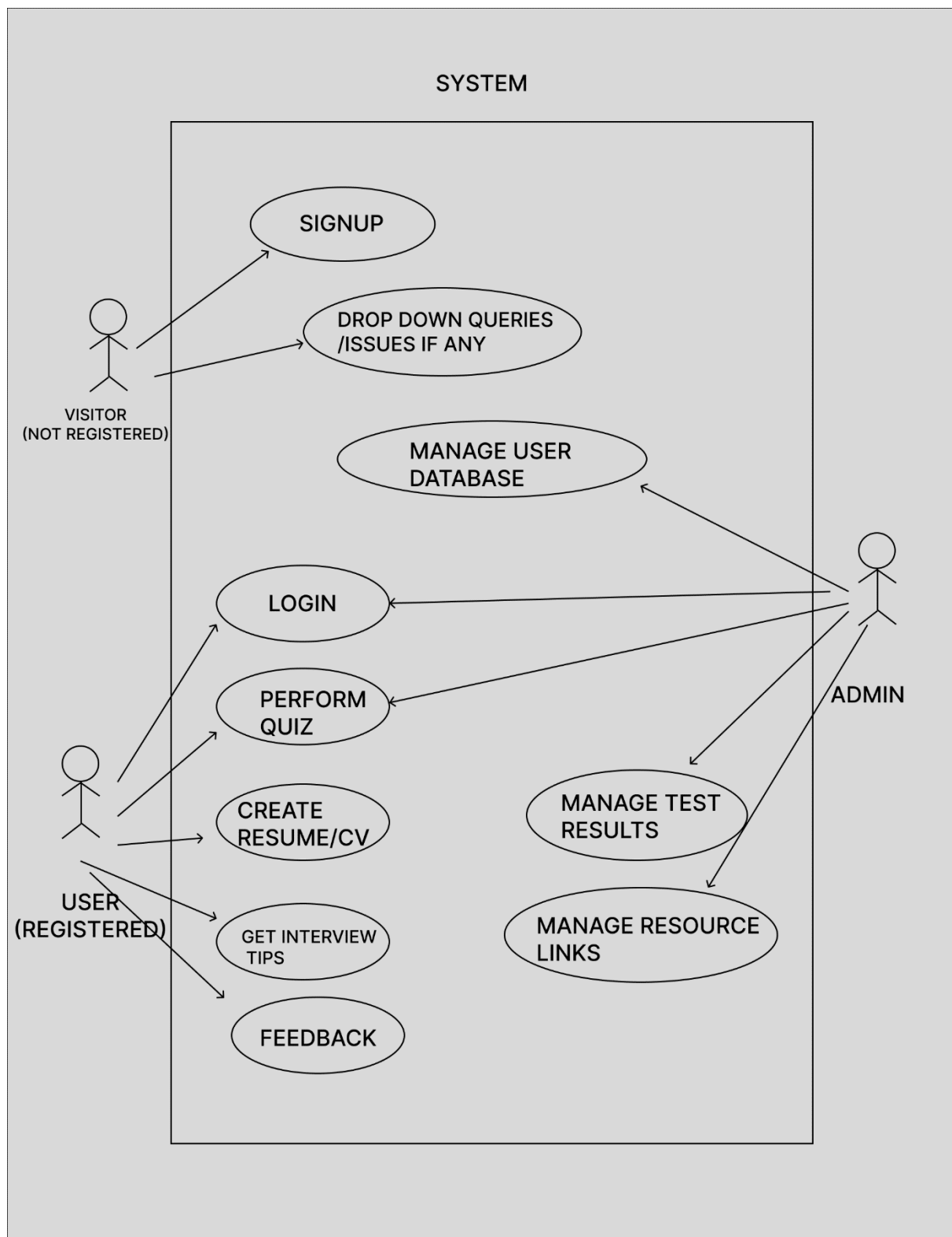


Figure-2.2 Use Case diagram for CMPS Portal.



A use case diagram is a graphical representation of the system's functional requirements from the user's perspective. It illustrates the interactions between the system and the actors (users, external systems, or other stakeholders) in various scenarios.

1. **Actors:** In our case it is users and admin. Actors are the external entities that interact with the system. They can be users, other systems, or external entities that interact with the system. Actors are represented as stick figures in the use case diagram.
2. **Use cases:** Use cases represent the system's functionality or behavior from the user's perspective. They describe a sequence of actions that the user performs to achieve a specific goal or outcome. Use cases are represented as ovals in the use case diagram.  
Ex: login/signup, text, study materials etc.
3. **Relationships:** The relationships between actors and use cases are shown in the use case diagram. An actor can interact with multiple use cases, and a use case can involve multiple actors. The relationships between actors and use cases are represented as lines in the diagram.
4. **System boundary:** The system boundary represents the scope of the system and the boundaries of the use cases. The system boundary is shown as a box that encloses the use cases and actors.

## 2.3 Activity Diagram

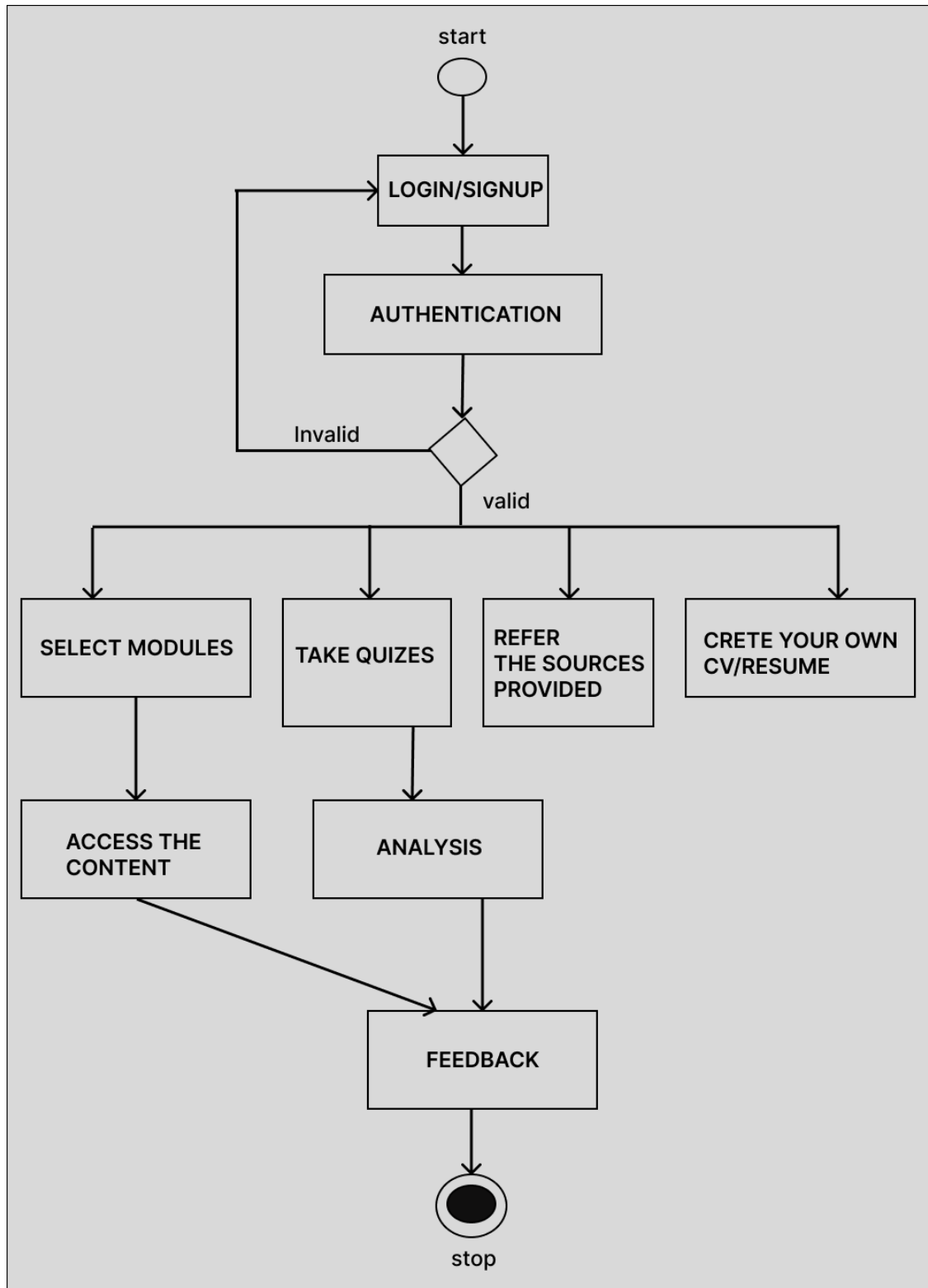


Figure-2.3 Activity diagram for CMPS Portal.

An activity diagram is a graphical representation of the workflow or process of a system, business process, or software application. It shows the sequence of actions, decisions, and branching paths that occur during the execution of the process. Activity diagrams are used to model complex workflows and to illustrate the flow of activities in a system.

Activity diagrams typically include the following elements:

1. **Activities:** Activities are the tasks or actions that are performed as part of the process. They are represented as rounded rectangles in the diagram.
2. **Transitions:** Transitions show the flow of control between activities. They represent the movement from one activity to another and are represented as arrows in the diagram.
3. **Decisions:** Decisions are represented by diamonds in the activity diagram. They represent points in the process where a decision needs to be made, and the process flow can take different paths based on the decision.
4. **Start and End Points:** The start point indicates where the process starts, and the end point represents where the process ends. They are represented as black circles in the diagram.

The activity diagram helps to understand the flow of the website designed in a graphical way. Starting from the login to logout and redirecting to dashboard, the activity diagram with elements like modules, quizzes etc defines the flow.

## CHAPTER 3

### WEB SITE FRONT END IMPLEMENTATION

#### 3.1 Tools used

1. HTML: A markup language used for creating the structure of web pages.
2. CSS: A styling language used for defining the look and feel of web pages.
3. JavaScript: A programming language used for adding interactivity and dynamic behavior to web pages.
4. jQuery: A JavaScript library used for simplifying and enhancing the process of creating web pages.

#### 3.2 Web Page Design

##### 3.2.a Main page of the website

```
<!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>mainpage</title>
<!-- <link rel="stylesheet" href="mainstyle.css"> -->
<style>
  body {
    background-image: url(bg.jpg);
    background-repeat: no-repeat;
    background-attachment: fixed;
    background-size: cover;
  }
  header {
```

```

    background-color: #f2f2f2;
    padding: 20px;
    text-align: center;
}
nav {
    background-color: #333;
    overflow: hidden;
}
nav a {
    float: left;
    color: white;
    text-align: center;
    padding: 14px 16px;
    text-decoration: none;
    font-size: 17px;
}
nav a:hover {
    background-color: #ddd;
    color: black;
}
nav ul {
    list-style-type: none;
    margin: 0;
    padding: 0;
    overflow: hidden;
}
nav li {
    float: left;
}
/* Add this CSS code to style the sections */
section {
    background-color: #f2f2f2;
    padding: 20px;

```

```

    margin-top: 0px;
}
section h2 {
    font-size: 24px;
    margin-bottom: 0px;
}
section p {
    font-size: 18px;
    line-height: 1.5em;
}
/* Add this CSS code to add more colors */
#aboutsection {
    background-color: #F9D5E5; /* light pink */
    color: #4A154B; /* dark purple */
}

#userguidesection {
    background-color: #FCEADE; /* light yellow */
    color: #C84B31; /* dark orange */
}

#contactsection {
    background-color: #C7CEEA; /* light blue */
    color: #3D3D6B; /* dark blue */
}
/* Add this CSS code to change the font style and size */
section h2 {
    font-size: 24px;
    margin-bottom: 0px;
font-family: Arial, sans-serif;
    font-size: 2em;
}

```

```

    section p {
        font-family: Arial, sans-serif;
        font-size: 1em;
        line-height: 1.5em;
    }
    footer p{
        text-align: center;
        text-transform: capitalize;
        text-decoration:solid;
    }
    /*home button styling*/
    .button {
        display:inline-block;
        border-radius: 50px;
        background-color: #f47709;
        border:#1d491a;
        color: #080101;
        text-align: center;
        font-size: 18px;
        padding: 20px;
        width: 150px;
        height: 10px;
        transition: all 0.5s;
        cursor: pointer;
        margin-top: 20px;
    }
    .button a {
        text-decoration: none;
        color: #1d491a;
        margin-top: 20px;
        margin-bottom: 20px;
        text-align: center;
    }

```

```

.button span {
  cursor: pointer;
  display: inline-block;
  position: relative;
  transition: 0.5s;
  margin-bottom: 20px;
}
.button span:after {
  content: '\00bb';
  position: absolute;
  opacity: 0;
  top: 0;
  right: -20px;
  transition: 0.5s;
}
.button:hover span {
  padding-right: 25px;
}
.button:hover span:after {
  opacity: 1;
  right: 0;
}
address h4{
  color: #ddd;
}
</style>
</head>
<body>
<header>
   <br>
</header>
<nav class="navbar" id="navbar">
  <ul>

```



```

<li> <a href="signup.html">SIGN UP</a> </li>
<li> <a href="#aboutsection">ABOUT</a></li>
<li> <a href="#userguidesection">USER GUIDE</a></li>
<li> <a href="#contactsection">CONTACT US</a></li>
<li> <a href="notification.html"> </a></li>
</ul>
</nav>
<br>
<section id="aboutsection">
  <p> About Us: <br> <br>
    A guided resource point for your overall career and placement development.
  <br>
  We ensure to provide you with different modules of assesments, resources, games that
  ensure a smooth path to develop your aras
  of intrest and crack your job interviews.
</p>
</section>
<section id="userguidesection">

  <p> User Guide: <br> <br>
  Thanks for opting CMPS portal. <br>
  You can signup via signup page present in the top menu to access free resources. <br>
  We provide you with study-modules that can be accessed through your emails. <br>
  Under interview preparation module, you will find various quiz assesments and
  interview tips. <br>
  Under resources module, you can access various links and guids related to programming
  and aptitude lessons. <br>
  under career managemant module, you can surf mind games,personality tests and many
  more. <br>
  Feel free to contact us or drop your queries anytime at contactus page via e-mail.
<br>
</p>
</section>

```

```

<section id="contactsection">
  <p> Contact: <br> <br>
    Mail Us At <a href="mailto:CMPS@gmail.com">CMPS portal</a>.<br>
    You can follow us at:
    <ul>
      <li> <a href="https://www.linkedin.com/in/durga-chandana-sree-mandapati-
b74ab9203/"> Linkdin </a></li> <br>
    </ul></p>
</section>

<script>
window.addEventListener("scroll", function () {
  const sections = document.querySelectorAll("section");
  const navLinks = document.querySelectorAll("nav a");
  let current = "";
  sections.forEach(function (section) {
    const sectionTop = section.offsetTop;
    const sectionHeight = section.clientHeight;

    if (pageYOffset >= sectionheader - sectionHeight / 3) {
      current = section.getAttribute("id");
    }
  });
  navLinks.forEach(function (link) {
    link.classList.remove("active");
    if (link.getAttribute("href") === "#" + current) {
      link.classList.add("active");
    }
  });
});
</script>
<FOOTER>

```

```

        <address
            <button class="button" style="vertical-align:middle"><span><a
href="#">Home</a> </span></button>

        <p>
            <a href="linkedin.com">linkedin</a> | <a href="github.com">git</a>
        </p>
        <h4> &copy;Terms And Conditions Of Usage applied </h4>
    </address>
</FOOTER>
</body>
</html>

```

### 3.2.b Signup/login page of the website

```

<!DOCTYPE html>
<html>
<head>
    <title>Login or Signup</title>
    <meta charset="UTF-8">
    <link rel="stylesheet" href="signupstyle.css">
</head>
<body>
    <header>
        <br>
        <B>FREE AND EASY ACCESS TO THE SPECIFIED RESOURCES..! </B> <br> <br>
        SIGN UP NOW!
    </header>
    <div id="login-form">
        <h2>Login</h2>
        <form id="formlogin" method="get" autocomplete="on" action="homepage.html">
            <label for="email">Email:</label>
            <input type="email" id="email" name="email" placeholder="Enter registered email-id"
autofocus required><br>
            <label for="password">Password:</label>

```

```

<input type="password" id="password" name="password" required><br>
<input type="checkbox" id="Remember me" name="Remember me">
<label for="Remember me">Remember me</label> <br>
<input type="submit" value="Submit" id="submit" onclick="validate()">
<input type="reset" value="Reset">
</form>
</div>
<script>
function validate(){

var email = document.getElementById("email").value;
var password = document.getElementById("password").value;
if( email == "Test@gmail.com" && password == "#123"){
    alert("Login successfully");
    window.location = "homepage.html";
    return false;
}
else{
    alert("Login failed - Please enter correct Username and Password")
}
}
</script>
<div id="signup-form">

<h2>Signup</h2>
<form action="/signup" method="post">
    <label for="name">Name:</label>
    <input type="text" id="name" name="name" autofocus required><br>
    <label for="email">Email:</label>
    <input type="email" id="email" name="email" placeholder="Enter valid mail-id"
required><br>
    <label for="tel">Phone Number:</label>

```

```

        <input type="tel" id="tel" name="phone number" pattern="\d{10}"
placeholder="xxxxxxxxxx" required> <br>
        <label for="password">Password:</label>
        <input type="password" id="password" name="password" required><br>
        <label for="password-confirm">Confirm Password:</label>
        <input type="password" id="password-confirm" name="password-confirm"
required><br>
        <input type="submit" value="Submit">
    </form>
</div>
</body> <br>

<footer>
    <address>
        <a href="main.html">Home</a>
    </address>
</footer>

</html>

```

### 3.2.c Home page of the website

```

<!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>homepage</title>
<style>
body {
    background-image: url(bg.jpg);
    background-repeat: no-repeat;

```

```
background-attachment: fixed;
background-size: cover;
}
```

```
header {
    background-color: #f2f2f2;
    padding: 20px;
    text-align: center;
}
```

```
nav {
    background-color: #333;
    overflow: hidden;
}
```

```
nav a {
    float: left;
    color: white;
    text-align: center;
    padding: 14px 16px;
    text-decoration: none;
    font-size: 17px;
}
```

```
nav a:hover {
    background-color: #ddd;
    color: black;
}
```

```
nav ul {
    list-style-type: none;
    margin: 0;
    padding: 0;
    overflow: hidden;
```

```

    }

    nav li {
        float: right;
    }
/* modules */
.modules{
    background: rgb(248, 242, 210);
    padding: 2rem 2rem;
}
.modules .title{
    text-align: center;
}
.title h1 {
    font-size: 2rem;
    text-transform: capitalize;
}
.title span{
    color: rgb(42, 142, 31);
}
.modules_boxes{
    display: flex;
    align-items: center;
    justify-content: center;
    margin-top: 1rem;
}
.box{
    width: 320px;
    background: #dfdeccd0;
    padding: 2rem;
    margin: 1rem 2rem;
    text-align: center;
    border-radius: 10px;

```

```

border: 2px solid transparent;
box-shadow: 1px 30px 20px rgba(0, 0, 0, 0.1);
}
.box i{
font-size: 4rem;
margin-bottom: 2rem;
}
.box p{
color: gray;
margin-top: 10px;
}
.box:hover, .br{
border: 2px solid rgb(213, 228, 8);
}
.box a{
color: black;
text-decoration: none;

</style>
</head>
<body>
<header>
 <br>
</header>
<nav class="navbar" id="navbar">
<ul>
<li> <a href="notification.html"> </a></li>
<li><a href="main.html">LOGOUT</a></li>
</ul>
</nav>
</section>
<section class="modules">
<div class="container">

```



```

<div class="title">
  <h1>Vist our <span>modules..</span></h1>
</div>
<div class="modules_boxes">
  <div class="box">
    <i class="fa-solid fa-hotel"></i>
    <a href="interview.html"><h4>INTERVIEW PREPARATION</h4></a>
  </div>
  <div class="box br">
    <i class="fa-solid fa-plane"></i>
    <a href="resources.html"> <h4>RESOURCES</h4></a>
  </div>
  <div class="box">
    <i class="fa-solid fa-mountain-sun"></i>
    <a href="career.html"><h4>CAREER MANAGEMENT</h4></a>
  </div>
</div>
</div>
</div>
</section>
</body>
</html>

```

### 3.2.d Interview.html page of the website

```

<!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>interviewpage</title>
  <style>
    body {
      font-family: Arial, sans-serif;

```

```
background-color: #f5f5dc;
color: #333;
margin: 0;
}
```

```
/* Style the section */
section {
  max-width: 800px;
  margin: 0 auto;
  padding: 20px;
}
```

```
/* Style the quiz module */
.quiz {
  text-decoration: none;
  position: relative;
  top: 10px;
  margin-bottom: 20px;
  margin-top: 25%;
}
```

```
.quiz h1 {
  font-size: 32px;
  margin-top: 0;
}
```

```
.quiz ul {

  margin-left: 30px;
}
```

```
.quiz li {
  font-size: 18px;
```

```

    margin-bottom: 10px;
    text-decoration: none;
}

/* Style the tips section */
.tips {
    text-align: left;
    margin-bottom: 50px;
}

.tips h1 {
    font-size: 32px;
    margin-top: 0;
}

.tips img {
    display: block;
    margin: 0 auto;
    max-width: 50%;
    height: auto;
    margin-bottom: 30px;
}

.tips p {
    font-size: 20px;
    text-align: justify;
    line-height: 1.5;
    margin-bottom: 20px;
}

/* Style the header */
header {
    background-color: #ffd5d5;

```

```
padding: 20px;
text-align: center;
position: fixed;
top: 0;
left: 0;
width: 100%;
z-index: 100;
}
```

```
header img {
  display: inline-block;
  max-width: 100%;
  height: auto;
}
```

```
header p {
  font-size: 20px;
  margin-top: 0;
  float: right;
  margin-right: 20px;
}
```

```
header p a {
  color: #333;
  text-decoration: none;
  font-weight: bold;
  padding: auto;
  margin-right: 10px;
  border: 2px;
}
```

```
header p a:hover {
  color: #666;
```

```

}

/* Style the link */
a {
  color: #007bff;
  text-decoration: none;
}

a:hover {
  text-decoration: underline;
}

</style>
</head>
<body>
  <header>
     <br>
    <p><a href="homepage.html">HOME</a></p>
  </header>
<br><br>
  <section>
    <div class="quiz">
      <h1>Quiz modules</h1>
      <ul>
        <li> <a href="verbalquiz.html">Verbal aptitude</a> </li>
        <li><a href="quantquiz.html">Qantative Analysis</a></li>
        <li><a href="aptitudequiz.html">Programming aptitude</a></li>
        <li><a href="debugquiz.html">Debugging</a></li>

      </ul>
    </div>

    <div class="tips">
      <h1>Here are few interview tips you may follow.</h1>

```

<center>



</center>

<p>

<u> <b>1. Know About Company and Job Role</b></u> <br>

The first and foremost step before going for any interviews is to do research about the organization you will be working for. You must go through the company website, its social media page, and research online about everything, including their year of establishment, work, competitors, challenges, and future goals. If you are asked, “What do you know about our company?”, then you must have the answer on your tips, do not fumble, and tell everything with confidence. Most of the candidates skip this important step and it gives the chance to the employer for cutting out them from the list.

Another important thing that the candidates must do is understand the job role, which is very important, if you have any doubts then it’s better to read about it before going for the interview. Many candidates do not research about the position they applied for and appear for the interview, which can make you look like a fool in front of the employer. If you have come across the job through advertisement or any other medium, then read and carefully and in case of any doubt, consult the recruiter.

<br> <br>

<u><b>2.Documents At One Place</b></u><br>

One of the most important things which should be done before going for an interview is to keep all your documents sorted. You must have several printed copies of your resume and do check them beforehand for any spelling errors or silly mistakes. Apart from this, line up all your academic documents, such as degrees, mark sheets, etc., both original and photocopies in one place for a hassle-free experience. You can also carry your previous internship certificates or any letter of recommendation issued by your previous mentor as it develops a bit more trust.

Furthermore, carrying a small notebook or notepad for writing down important things will also help. Along with this, it will also create a good impression on the employer and show your seriousness regarding the job role.

<br><br>

<u><b>3.Prepare Well For Basic Interview Questions</b></u><br>

Another tip for a successful interview is to prepare for the basic interview questions. There are many repetitive questions, which are asked by the employer, including “Tell us about yourself”, “What are your future goals?” “Why do you want to work with us?” “Talk about your strengths and weaknesses” and many more. Therefore do prepare the answers for these questions and make sure the answers match with your skills.

You can also make your answers interesting by narrating them in the form of stories and adding examples from your previous experience. In addition to this, be confident, if you want you can also rehearse with your family and friends and work on your weaknesses. Hence, all these things will help you to be “smooth as silk” at the time of the interview.

<br><br>

#### <u><b>4.Mock Interview</b></u><br>

Mock interview acts as a complete informational guide for your actual job interview. It provides you with a fair opportunity to practice and improve the points you are weak at. You can prepare with family members, friends, or any of your mentors by creating a completely professional environment. Though it is just for practice, keep all the necessary documents in one place and along with the notepad to write down the feedback. Apart from this, wear formal attire and see whether you are comfortable in that or not. Moreover, avoid sounding like a robot, do not memorize the answers but communicate in a way so that it sounds engaging and makes you look appropriate for the job. <br><br> <u><b>5.Dress Impressively and Be On Time</b></u><br>

These are the two things, which most of the candidates skip, leading to the chances of losing their dream job. Therefore, if you are going for a job interview then you must dress sharply. Wearing a crisp shirt and pants along with neat shoes will make you look formal and interview-ready. Along with this, you can also carry a suit if required as it will help you look more presentable. “Say No” to big watches and jewelry and do not wear very strong perfume as these things can bring unwanted attention.

<br><br>

#### <u><b>6.Show Smart Communication Skills</b></u><br>

Having good communication skills is one of the most important qualities that a candidate can possess. Be it verbal, non-verbal, or written, you should be able to communicate effectively and know how to convey your points. As fresher many candidates hesitate to talk or ask questions from the recruiter, but it should not be the case. If you are about to work with the person or company, you must ask everything from them. While giving answers, structure in a way that shows your confidence and does not sound negatively. Always listen carefully to the interviewer and answer concisely incorporating your skills.

<br><br>

#### <u><b>7.Be Focused and Maintain Positive Body Language</b></u><br>

During an interview, it is very necessary to remain focused and keep a check on the quality of answers. The way you deliver your answers exhibits your preparation. Always think before you speak and move ahead step by step. Do not seem to be stressed as it creates a negative impact on the person sitting in front of you. Your body language also plays an important role and gives an overall peek into your personality. Sit straight and firmly and make eye contact with the interviewer while talking. Also, do not sit idle, use gestures and hand movements while speaking as it will help you earn extra points and will show your connection with the answers. <br><br>

#### <u><b>8.Ask Relevant Questions</b></u><br>

This can be considered as one of the pro tips for freshers. Asking question from your interviewer manifests your research and makes you look curious. On asked “Do you have any doubts or questions” consider it as a golden chance for showcasing your skills and qualities. You can impress the interviewer by asking intelligent questions about their business and industry which you might join soon. This also gives you more clarity about the work which you will be doing soon.

<br><br>

<u><b>9.Think Positively</b></u><br>

Having an optimistic attitude is a life-saving measure. We all know that during and after an interview we all are afraid about many things, especially our future. But maintaining a positive mindset is necessary as it also helps you to showcase your confidence in front of the interviewer. Showing any freaked-out emotion to the person sitting in front of you can grab away the chance of getting the job. Therefore, it is important that you display a good attitude and try thinking from the perspective of the recruiter that what they want. As everyone wants to hire an employee who can work with other teammates and does not panic in any situation.

<br><br>

<u><b>10.Follow Up After Completion</b></u><br>

Thanking the interviewer is considered a common courtesy after the completion of the interview. You should send a thank you mail within 24 hours showing your gratitude towards them and for providing you with such a great opportunity. It also keeps you at the top of their minds and makes you different from other candidates. Once you get the reply you can plan accordingly and make up your mind about joining the company.

Also, it is not necessary that every time you will get a positive response, hence in that case you can ask for feedback upon why you are not offered the job. This will help you improve for your future interviews and you might get considered in the same organization for any different role

Hence, following these above-mentioned tips will help you prepare for the interview and help you succeed. Though being a fresher you might not have any experience, putting all the effort into your interview preparation might help you to crack one and get your desired job!!

<br><br>

</p>

</div>

</body>

</html>



### 3.2.e resources.html page of the website

```
<!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>resources</title>

</style>

  header {

    background-color: #f2f2f2;

    padding: 20px;

    text-align: center;

  }

  nav {

    background-color: #333;

    overflow: hidden;

  }

  nav a {

    float: left;

    color: white;

    text-align: center;

    padding: 14px 16px;

    text-decoration: none;

    font-size: 17px;

  }

  nav a:hover {

    background-color: #ddd;

    color: black;
```

```
}
```

```
nav ul {  
    list-style-type: none;  
    margin: 0;  
    padding: 0;  
    overflow: hidden;  
}
```

```
nav li {  
    float: left;  
}
```

```
/* Add this CSS code to style the sections */
```

```
section {  
    background-color: #f2f2f2;  
    padding: 20px;  
    margin-top: 0px;  
}
```

```
section h2 {  
    font-size: 24px;  
    margin-bottom: 0px;  
}
```

```
section p {  
    font-size: 18px;  
    line-height: 1.5em;  
}
```

```
/* Add this CSS code to add more colors */
```

```
#guidesection {  
    background-color: #F9D5E5; /* light pink */
```

```
color: #4A154B; /* dark purple */  
}
```

```
#pptsection {  
  background-color: #FCEADE; /* light yellow */  
  color: #C84B31; /* dark orange */  
}
```

```
#youtubesection {  
  background-color: #C7CEEA; /* light blue */  
  color: #3D3D6B; /* dark blue */  
}
```

```
/* Add this CSS code to change the font style and size */
```

```
section h2 {  
  font-size: 24px;  
  margin-bottom: 0px;  
font-family: Arial, sans-serif;  
  font-size: 2em;  
}
```

```
section p {  
  font-family: Arial, sans-serif;  
  font-size: 1em;  
  line-height: 1.5em;  
}
```

```
/*styling youtube links*/  
.video-container {  
  align-items: center;  
}
```

```
</style>
```

```

</head>
<body>
  <header>
     <br>
  </header>

  <nav class="navbar" id="navbar">
    <ul>
      <li> <a href="#guidesection">GUIDE</a></li>
      <li> <a href="#pptsection">PPT</a></li>
      <li> <a href="#youtubesection">YOUTUBE REFERENCES</a></li>
      <li> <a href="homepage.html">HOME</a></li>
    </ul>
  </nav>

  <br>
  <p></p>

  <section id="guidesection">

```

```

  <p> <b><u>Interview Guide</u></b> <br>

```

The following guide is a sample document which includes the types of information and materials hiring

managers would want to provide to interviewers when preparing them to conduct behavioral

interviews.

The HR team should use this document as a template to help prepare an interview guide that would suit

their own organization's needs. In addition to your organization's interview guide, provide your

interviewers with: <br>

1. Job Description <br>

2. Candidate's Resume<br> <br>

<a href="guides">FOLLOW FOR MORE GUIDES</a>

</section>

<section id="pptsection">

<p> <b><u>Powerpoint presentations</u></b> <br> <br>

<div class="pdf">

<p>Click here to access the pdfs <a href="signup.html">OPEN!</a>.</p>

</div>

</p>

</section>

<section id="youtubesection">

<p> <b><u>Youtube References</u></b> <br> <br>

<div class="video-container">

<iframe width="560" height="315" src="https://www.youtube.com/embed/JZK1MZwUyUU" title="YouTube video player" frameborder="0" allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope; picture-in-picture; web-share" allowfullscreen></iframe>

<iframe width="560" height="315" src="https://www.youtube.com/embed/OhJWg-0qdI0" title="YouTube video player" frameborder="0" allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope; picture-in-picture; web-share" allowfullscreen></iframe>

```

<iframe width="560" height="315"
src="https://www.youtube.com/embed/Wa9NuZ90UbI" title="YouTube video player"
frameborder="0" allow="accelerometer; autoplay; clipboard-write; encrypted-media;
gyroscope; picture-in-picture; web-share" allowfullscreen></iframe>

</div>

</p>

</section>

</body>

</html>

```

### 3.2.d career.html page of the website

```

<!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>career</title>

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

<style>

  header {

    background-color: #ffd5d5;

    padding: 20px;

    text-align: center;}

  header p {

    float:right;

    text-decoration: none;

  }

  .resumeref {

display: flex;

flex-wrap: wrap;

justify-content: center;

```

```

}
.resumeref a,
.resumeref img {
  display: block;
  width: 200px;
  height: auto;
  margin: 10px;
  border: 2px solid #060606;
  border-radius: 5px;
}
.resumeref img:hover {
  opacity: 0.7;
}
.resumeref img:hover {
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.3);
  transform: scale(1.1);
}
body {
  background-color: #f5f5dc;
  color: #333;
}
table, th, td {
  border: 1px solid black;
  background-color: #ffffff;
}
</style>
</head>
<body>
  <header>
     <br>
    <p> <a href="homepage.html">Home</a> </p>
  </header>
  <section>

```

```

<div class="resume">

  <p>
    <h1>Resume references:</h1> <br>
    <!--insert resume ref images here-->
    <div class="resumeref">

      <a href="https://www.google.com"></a>
        <a href="https://www.google.com"> 
</a>

      <a href="https://www.google.com">  </a>
        <a href="https://www.google.com">  </a>

    </div>
  </p>
</div>

<div class="resume">
  <p><h1> Build your resume today!</h1>
<form>
  <h2>Personal Information</h2>
  <label for="name">Name:</label>
  <input type="text" id="name" name="name" required>

  <label for="email">Email:</label>
  <input type="email" id="email" name="email" required>

  <label for="phone">Phone:</label>
  <input type="tel" id="phone" name="phone" required>

  <label for="address">Address:</label>
  <input type="text" id="address" name="address" required>

  <h2>Education</h2>

```



```

<label for="degree">Degree:</label>
<input type="text" id="degree" name="degree" required>

<label for="university">University:</label>
<input type="text" id="university" name="university" required>

<label for="graduation-date">Graduation Date:</label>
<input type="date" id="graduation-date" name="graduation-date" required>

<h2>Work Experience</h2>
<label for="job-title">Job Title:</label>
<input type="text" id="job-title" name="job-title" required>

<label for="employer">Employer:</label>
<input type="text" id="employer" name="employer" required>

<label for="start-date">Start Date:</label>
<input type="date" id="start-date" name="start-date" required>

<label for="end-date">End Date:</label>
<input type="date" id="end-date" name="end-date" required>

<label for="job-description">Job Description:</label>
    <textarea id="job-description" name="job-description" rows="4" cols="50"
required></textarea>

<button type="button" id="generate-cv">Generate CV</button>
</form>
<script>
const generateCVButton = document.getElementById('generate-cv');
generateCVButton.addEventListener('click', generateCV);
function generateCV() {
    const name = document.getElementById('name').value;

```

```

const email = document.getElementById('email').value;
const phone = document.getElementById('phone').value;
const address = document.getElementById('address').value;
const degree = document.getElementById('degree').value;
const university = document.getElementById('university').value;
const graduationDate = document.getElementById('graduation-date').value;
const jobTitle = document.getElementById('job-title').value;
const employer = document.getElementById('employer').value;
const startDate = document.getElementById('start-date').value;
const endDate = document.getElementById('end-date').value;
const jobDescription = document.getElementById('job-description').value
const cvTemplate = `

```

```

<table>
  <tr>
    <td colspan="2">
      <h2>${ name }</h2>
      <p>${ email } | ${ phone } | ${ address }</p>
    </td>
  </tr>
  <tr>
    <td>
      <h3>Education</h3>
    </td>
    <td>
      <h3>Work Experience</h3>
    </td>
  </tr>
  <tr>
    <td>
      <p><strong>${ degree }</strong></p>
      <p>${ university }</p>
      <p>Graduated: ${ graduationDate }</p>
    </td>
  </tr>

```

```

<td>

    <p><strong>${jobTitle}</strong></p>

    <p>${employer}</p>

    <p>${startDate} - ${endDate}</p>

    <p>${jobDescription}</p>

</td>

</tr>

</table>

`;

const cvOutput = document.createElement('div');
cvOutput.innerHTML = cvTemplate;

document.body.appendChild(cvOutput);
}

</script>

</p>    </div>

</section>

</body>

</html>

```

### 3.3 Screenshots

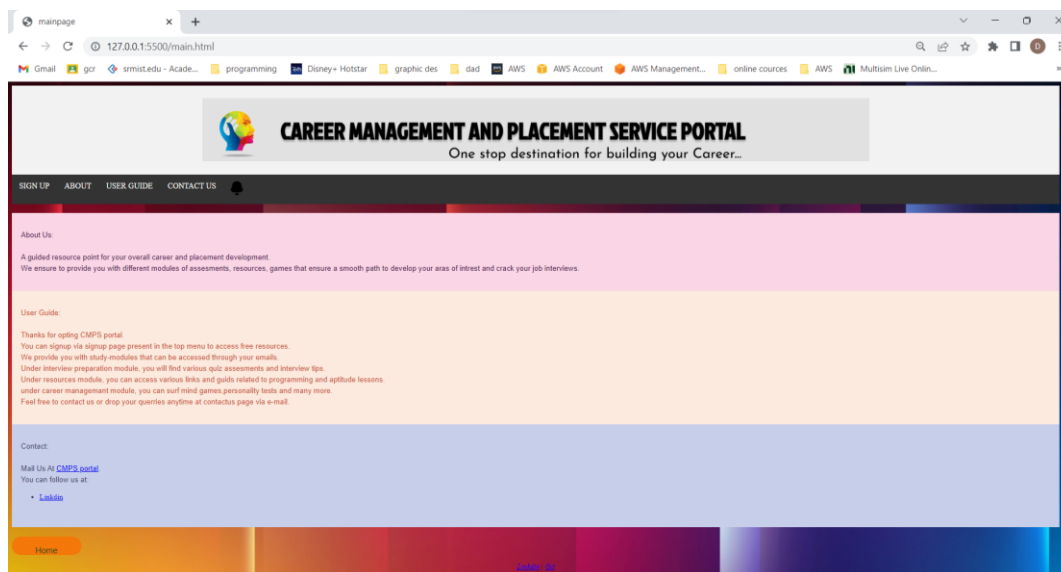


Fig 3.3.1 Main page of the website CMPS

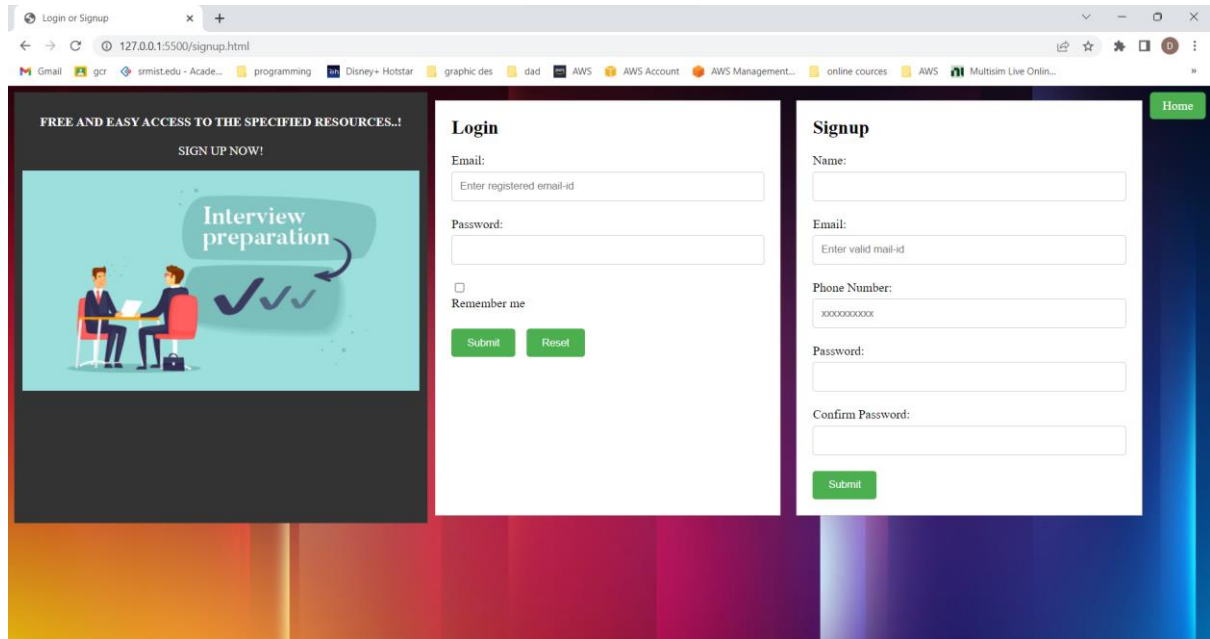


Fig 3.3.2login/signup page of website CMPS

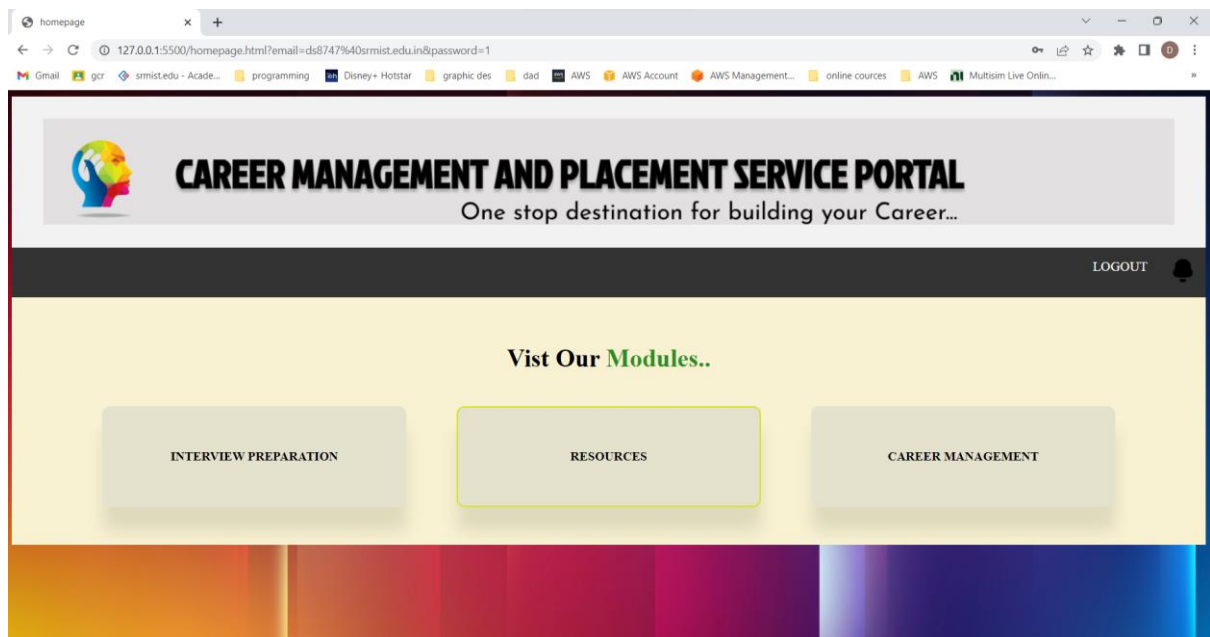


Fig 3.3.3 dashboard page

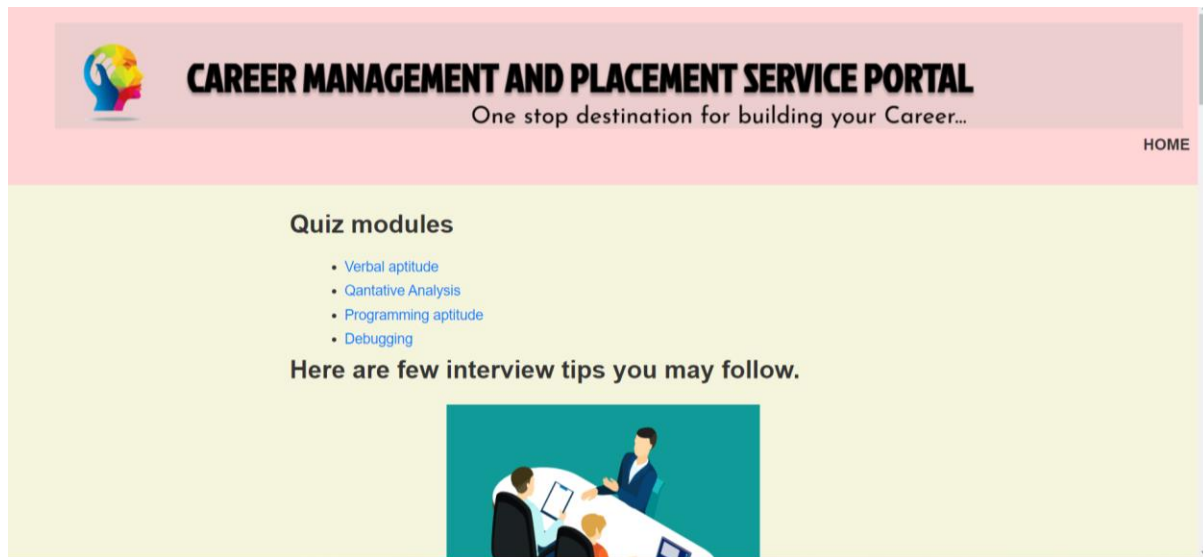
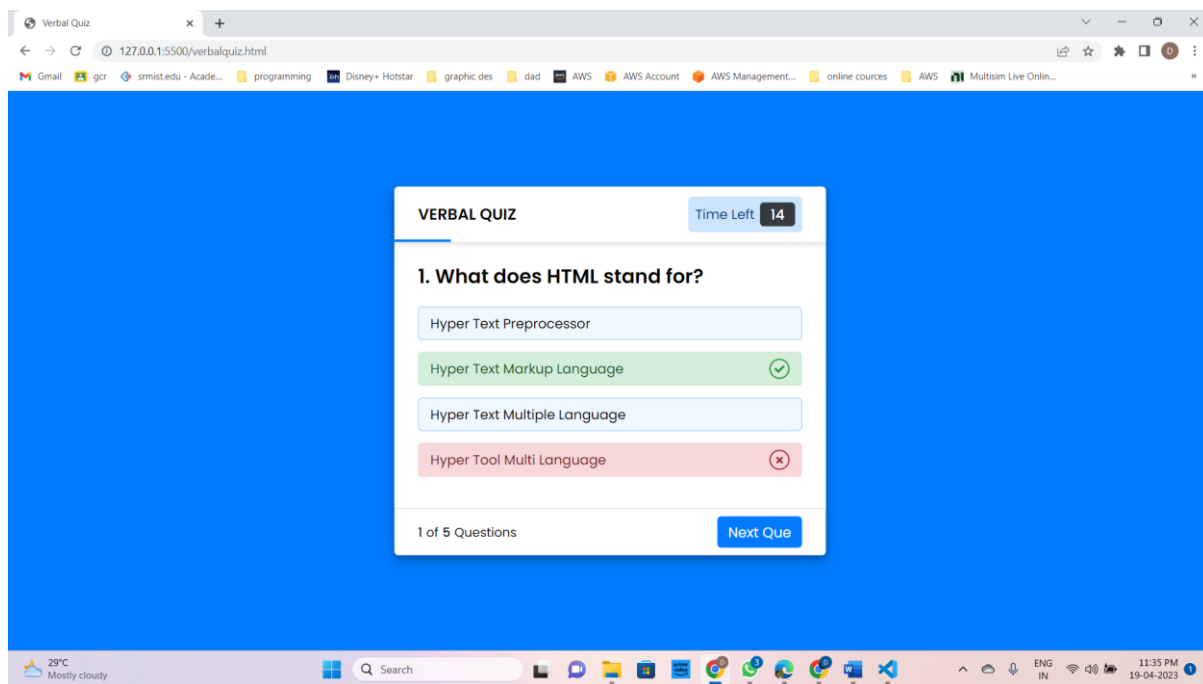


Fig 3.3.4 interview.html page



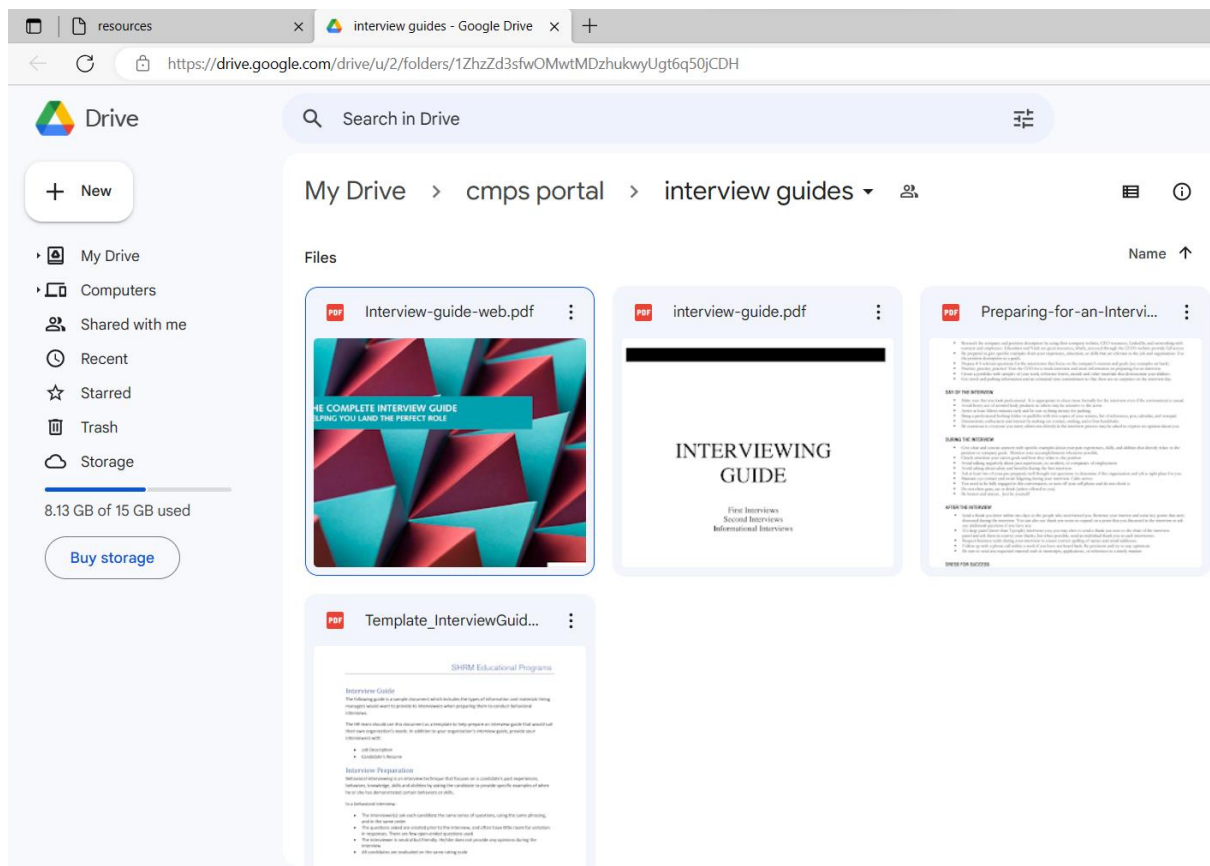
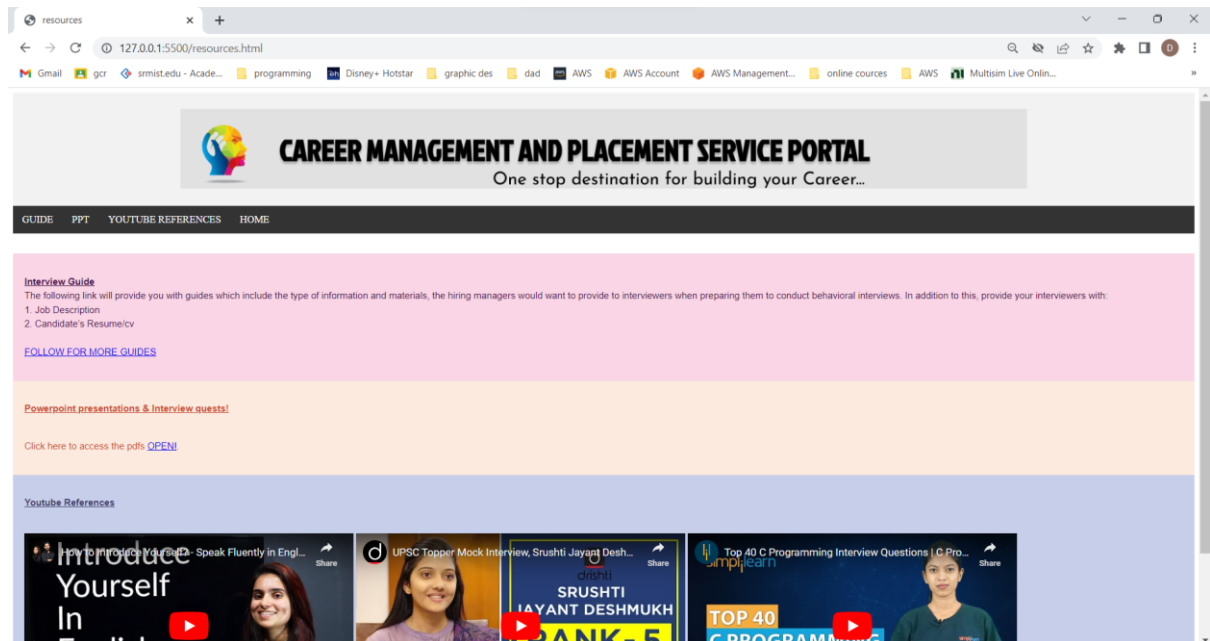


Fig 3.3.5 resources.html page

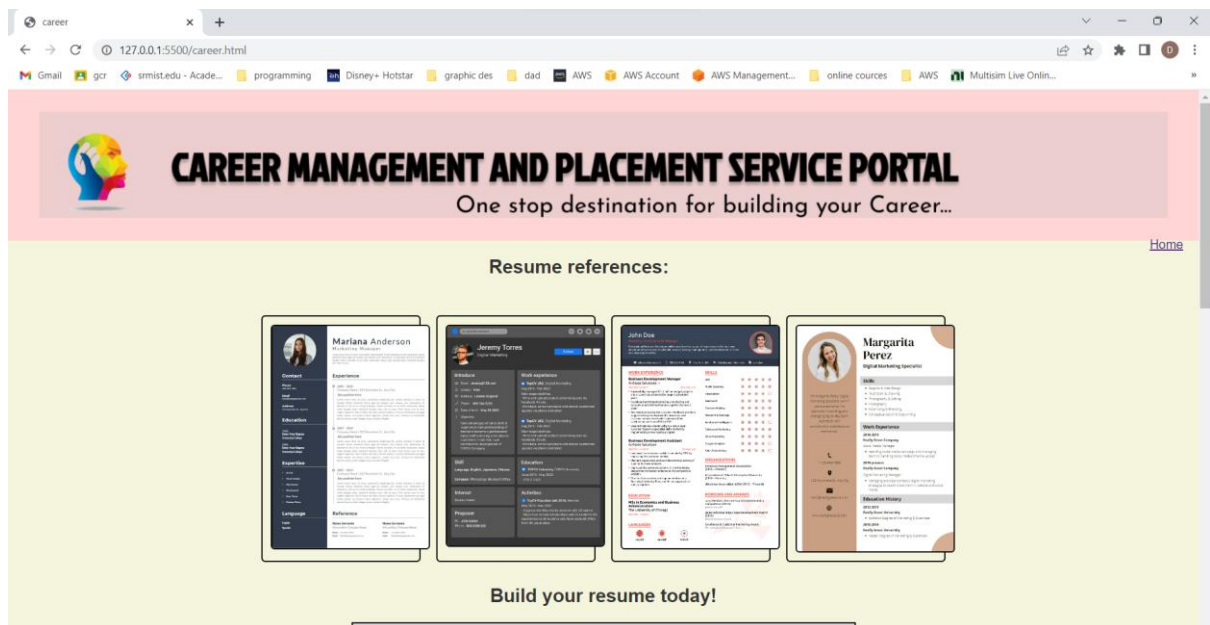
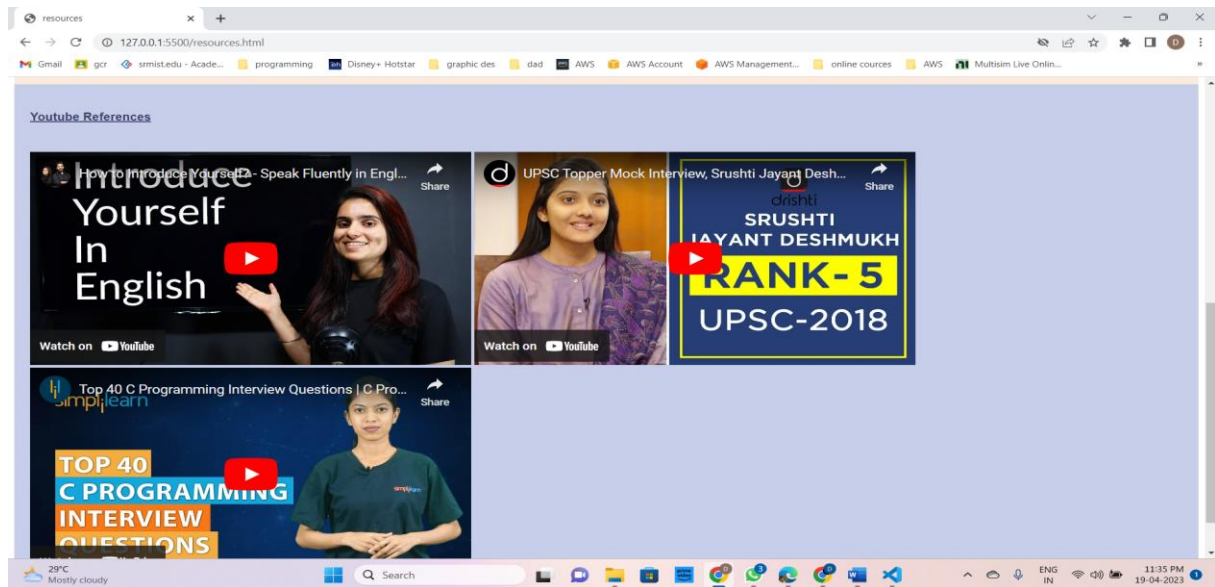


Fig 3.3.6 career.html page

Build your resume today!

**Personal Information**

Name:

Email:

Phone:

Address:

**Education**

Degree:

University:

Graduation Date:

**Work Experience**

STUDENT

Start Date:

End Date:

Job Description:

**Generate CV**

**DURGA CHANDANA SREE MANDAPATI**

ds747@srmist.edu.in | 9866449444 | 12-456/5/1 OPP SRI USHODAYA TOWERS,NH16

Education	Work Experience
btech srm Graduated: 2024-05-24	student STUDENT 2023-04-12 - 2023-04-29 na



## **CHAPTER 4**

### **WEB SITE BACK-END IMPLEMENTATION**

#### **4.1 Tools used**

**Servlet:** A servlet is a Java programming language class that dynamically generates web content, such as HTML, XML, or other types of documents. Servlets are typically used in web applications to handle HTTP requests and to generate responses to those requests.

When a client sends an HTTP request to a web server, the web server passes the request to the servlet container. The servlet container then looks for the appropriate servlet to handle the request based on the URL and passes the request to the servlet. The servlet processes the request, generates a response, and sends it back to the servlet container, which then sends the response back to the client.

**Servlet architecture:**

The servlet architecture consists of several components that work together to process HTTP requests and generate responses. The main components of the servlet architecture are:

1. **Servlet container:** The web server or application server that provides the environment for running servlets.
2. **Servlet:** A Java class that handles HTTP requests and generates responses.
3. **Request and response objects:** The objects that the servlet container creates for each HTTP request and response.
4. **Web application:** A collection of servlets, JSPs, HTML pages, and other resources that are organized into a directory structure and deployed to a servlet container.
5. **Deployment descriptor:** An XML file that provides configuration information for the web application.

The servlet architecture provides a robust and flexible framework for building web applications. Servlets can be used to handle a wide range of HTTP requests, generate dynamic content, and interact with databases and other resources. They are portable across different platforms and operating systems and can be easily integrated with other web technologies, such as JSP and JSF.

**Eclipse:** An IDE for running, and debugging programming languages and for deploying web applications for server-side functionalities. Here I have used Java class and servlet for running my login/signup page.

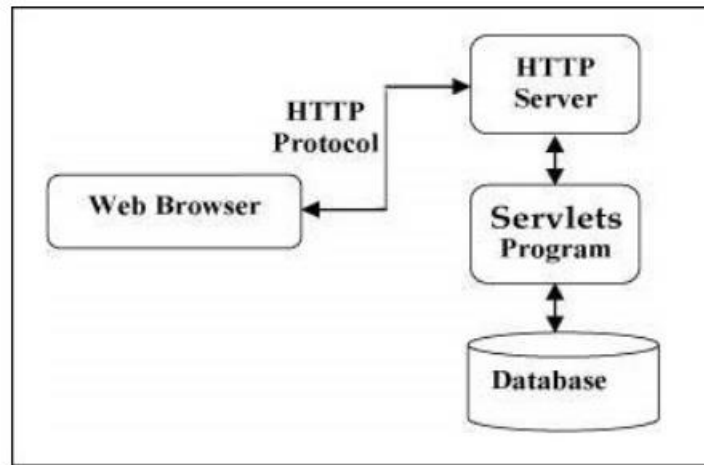


Fig 4.1.1 Servlet Architecture that has been used as a backend for CMPS portal.

## 4.2 Code for Back-end

### 4.2.1 loginservlet.java

```

package com.servlet;
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class loginservlet extends HttpServlet {

    protected void doGet(HttpServletRequest req, HttpServletResponse res) throws
    ServletException, IOException{

        PrintWriter pw =res.getWriter();
  
```

```

res.setContentType("text/html");

String name= req.getParameter("email");
String pwd = req.getParameter("password");
boolean status=false;

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

    Connection connection =
DriverManager.getConnection("jdbc:mysql://localhost:3306/login?useSSL=false","root","Dc
@04072002");

    java.sql.PreparedStatement ps = connection.prepareStatement("select * from user
where email = ? and password = ?");

    ps.setString(1, name);
    ps.setString(2, pwd);
    System.out.println(ps);
    ResultSet rs=ps.executeQuery();
    status=rs.next();
}
catch(SQLException | ClassNotFoundException e) {
    System.out.println(e);
}

if(status) {
    // Redirect to homepage.html upon successful login
    res.sendRedirect("homepage.html");
}
else {

    pw.print("<p style=color:red>Username and Password invalid</p>");
    pw.print("<form                                action=\"signup.html\"><button
type=\"submit\">Signup</button></form>");
}

```

```

    }

    protected void doPost(HttpServletRequest req, HttpServletResponse res) throws
ServletException, IOException{
        doGet(req,res);
    }
}

```

#### 4.2.2 signupervlet.java

```

package com.servlet;

import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 * Servlet implementation class signin
 */
@WebServlet("/signin")
public class signupervlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

```

```

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

    PrintWriter pw =response.getWriter();
    response.setContentType("text/html");
    String name=request.getParameter("name");
    String email=request.getParameter("email");
    String phoneno=request.getParameter("phone number");
    String password=request.getParameter("password");
    // RequestDispatcher dispatcher=null;

    try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection connection =
DriverManager.getConnection("jdbc:mysql://localhost:3306/login?useSSL=false","root","Dc
@04072002");

        java.sql.PreparedStatement ps =
connection.prepareStatement("INSERT INTO `login` .`user`
(email`,`password`,`name`,`phone number`) VALUES (?,?,,?)");

        ps.setString(1, email);
        ps.setString(2, password);
        ps.setString(3, name);
        ps.setString(4, phoneno);

        int row = ps.executeUpdate();
        // dispatcher = request.getRequestDispatcher("signin.jsp");
        if (row > 0) {
            response.sendRedirect("signup.html");
            // request.setAttribute("status","success");

        }
        else {
            // request.setAttribute("status" , "failed");

```

```

        response.sendRedirect("https://www.youtube.com/watch?v=YvPGeUkafHs");
    }
    // dispatcher.forward(request, response);
        connection.close();
    }
    catch(SQLException | ClassNotFoundException e) {
        System.out.println(e);
    }
}

protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
    // TODO Auto-generated method stub
    doGet(request, response);
}
}

```

#### 4.2.3 web.xml

```

<?xml version="1.0" encoding="UTF-8"?>

<web-app
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns="http://xmlns.jcp.org/xml/ns/javaee"
    xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee
http://xmlns.jcp.org/xml/ns/javaee/web-app_4_0.xsd" id="WebApp_ID" version="4.0">
    <display-name>ServletRequest</display-name>
    <welcome-file-list>

        <welcome-file>main.html</welcome-file>

    </welcome-file-list>
    <servlet>
        <servlet-name>backend</servlet-name>

```

```

    <servlet-class>com.servlet.loginservlet</servlet-class>

</servlet>

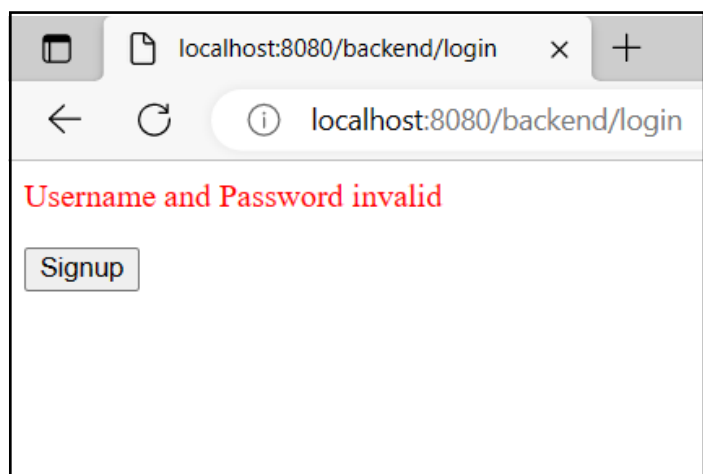
<servlet-mapping>
    <servlet-name>backend</servlet-name>

    <url-pattern>/login</url-pattern>
</servlet-mapping>
</web-app>

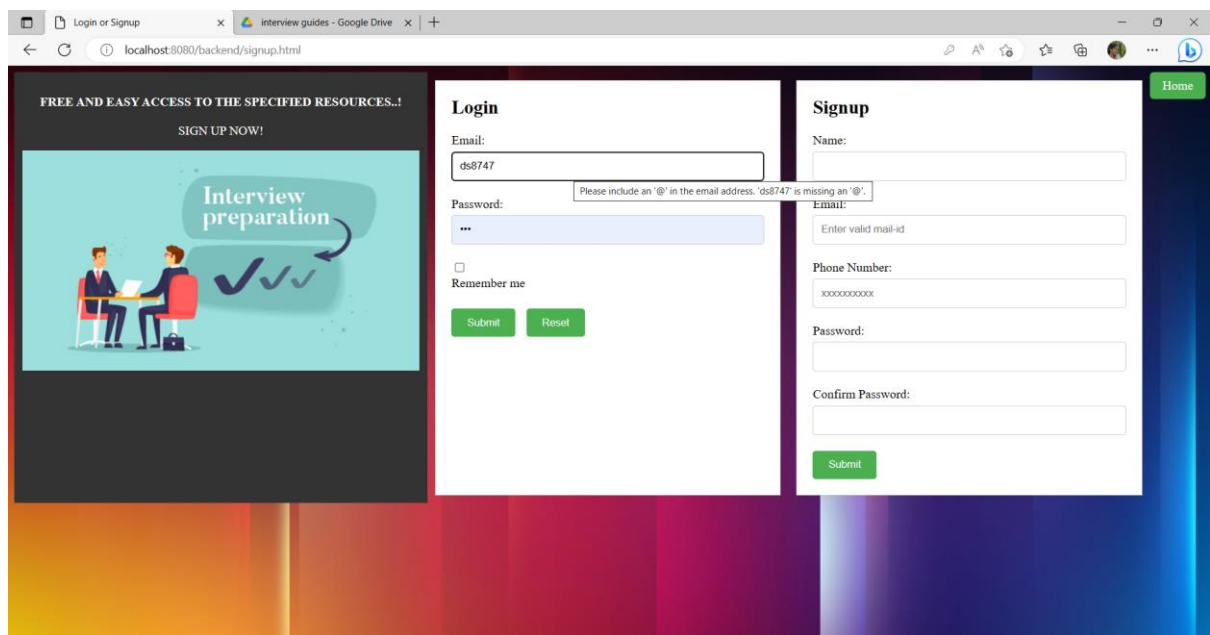
```

## Screenshots :

On entering wrong credentials, it directs to error page as follows.



Checks for the valid input from the user.



Giving credentials in the signup page that store in the user table of login schema.

The screenshot shows a web browser window with two tabs: 'Login or Signup' and 'Interview guides - Google Drive'. The URL is 'localhost:8080/backend/signup.html'. The page has a dark header with the text 'FREE AND EASY ACCESS TO THE SPECIFIED RESOURCES..! SIGN UP NOW!' and an illustration of two people in an interview. The main content area has two forms: 'Login' and 'Signup'. The 'Signup' form is filled with the following data:

Field	Value
Name	DURGA SREE
Email	durga@gmail.com
Phone Number	09876543210
Password	..
Confirm Password	

There is a 'Submit' button at the bottom of the Signup form and a 'Home' button in the top right corner.

Registered details will be stored in the login schema in MySQL Workbench.

The screenshot shows the MySQL Workbench interface. The 'Schemas' pane on the left shows the 'login' schema selected. The 'Query' pane shows the query 'SELECT \* FROM login.user;'. The 'Result Grid' pane shows the following data:

	id	email	password	name	phone number
1	1	ds@srmist.edu.in	123		
2	2	durgachandanasree4@gmail.com	12345		
3	3	cs6761@srmist.edu.in	123	satya	9866449444
4	4	arun@srmist.edu.in	12	arun	9866449444
5	5	durga@gmail.com	Dc	DURGA SREE	9876543210
*	NULL	NULL	NULL	NULL	NULL



## CHAPTER 5

### DATABASE DESIGN

#### 5.1 Database used

MySQL is a relational database management system (RDBMS) that uses SQL (Structured Query Language) to manage and manipulate data. It is known for its reliability, scalability, and ease of use. MySQL is often used for web applications because it integrates well with PHP and other web programming languages.

Here are some of the key features of MySQL Workbench:

1. Database design: MySQL Workbench provides a visual interface for designing database schemas, including tables, columns, and relationships.
2. SQL editing: You can use MySQL Workbench to write and execute SQL queries, including SELECT, INSERT, UPDATE, and DELETE statements.
3. Data modeling: MySQL Workbench includes tools for creating and managing data models, including entity-relationship diagrams (ERDs) and data flow diagrams (DFDs).
4. Database administration: MySQL Workbench includes tools for managing database administration tasks, such as creating users and roles, managing permissions, and performing backups and restores.
5. Migration: MySQL Workbench provides tools for migrating databases from other database management systems to MySQL.

#### 5.2 Database Schema and Tables

My database has a table called 'users' under the schema 'login'.

The user table further contains columns- id, email, and password.

Id is auto-incremented every time a new row is added serving as the primary key.

For signup / registered user's data, we have name and phone number too.

Once the user signs up the portal, his/her details will be reflected in this user table of login schema. In this way we save the details of the registered users in our database with the help of MySql workbench.

### 5.3 Database Connectivity

Database -MySQL is connected with the login/signup page of the website using login servlet and signup servlet. We have mapped the url in we.xml files in the eclipse ide.

#### **signupservlet.java**

```
package com.servlet;

import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 * Servlet implementation class signin
 */
@WebServlet("/signin")
public class signupservlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
        PrintWriter pw =response.getWriter();
        response.setContentType("text/html");
        String name=request.getParameter("name");
```

```

String email=request.getParameter("email");
String phoneno=request.getParameter("phone number");
String password=request.getParameter("password");
// RequestDispatcher dispatcher=null;

try{
    Class.forName("com.mysql.jdbc.Driver");
    Connection connection =
DriverManager.getConnection("jdbc:mysql://localhost:3306/login?useSSL=false","root","Dc
@04072002");

    java.sql.PreparedStatement ps =
connection.prepareStatement("INSERT INTO `login` .`user`
(`email`,`password`,`name`,`phone number`) VALUES (?,?,,?);");

    ps.setString(1, email);
    ps.setString(2, password);
    ps.setString(3, name);
    ps.setString(4, phoneno);

    int row = ps.executeUpdate();
    // dispatcher = request.getRequestDispatcher("signin.jsp");
    if (row > 0) {
        response.sendRedirect("signup.html");
        // request.setAttribute("status","success");

    }
    else {
        // request.setAttribute("status", "failed");
        response.sendRedirect("https://www.youtube.com/watch?v=YvPGeUkafHs");
    }
    // dispatcher.forward(request, response);
    connection.close();
}

```

```

        catch(SQLException | ClassNotFoundException e) {
            System.out.println(e);
        }
    }

    protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
        // TODO Auto-generated method stub
        doGet(request, response);
    }
}

```

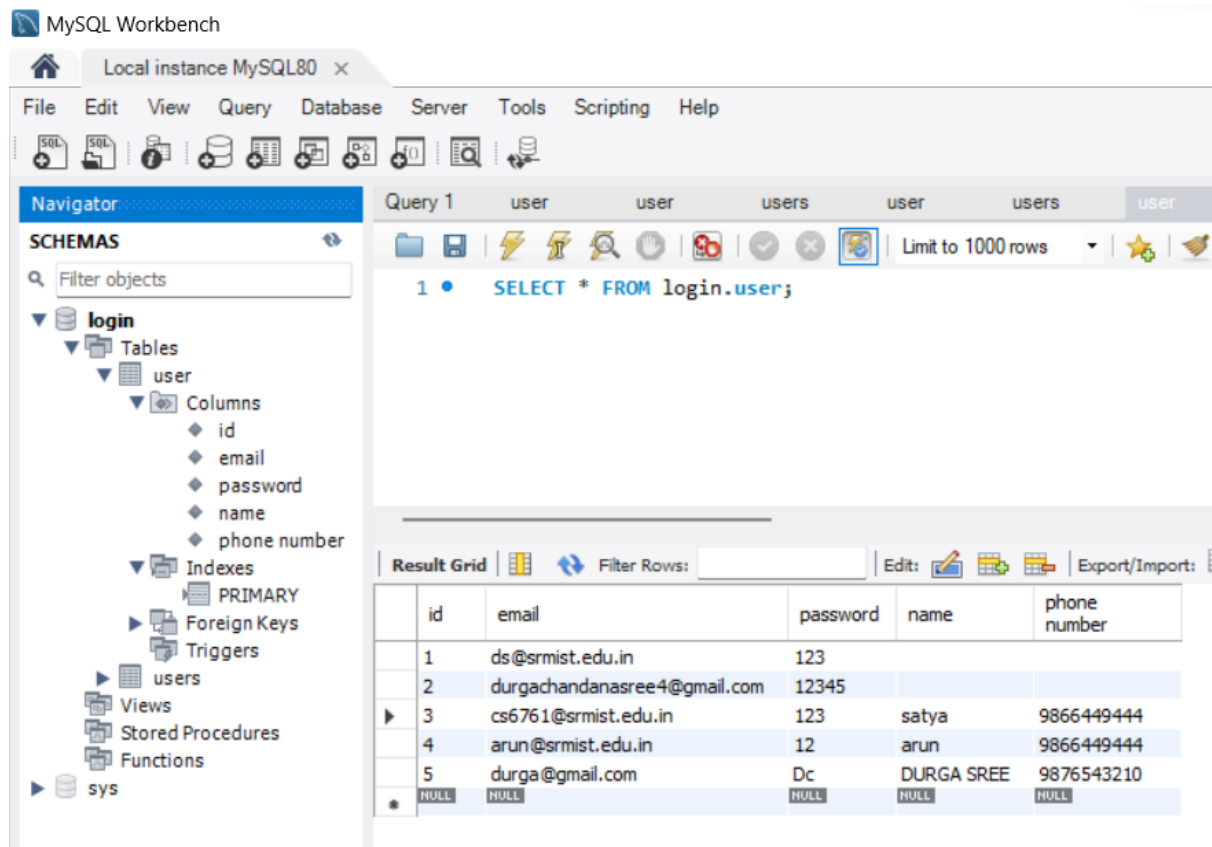


Fig 5.1 user table in mysql workbench

## **CHAPTER 6**

### **CONCLUSION AND FUTURE WORK**

#### **Conclusion:**

In conclusion, the Career Management and Service Portal is a comprehensive platform designed to assist job seekers in advancing their careers. The platform offers job seekers a range of services, including job listings, resume writing services, interview preparation tips, and resources on aptitude and verbal reasoning. The project aims to address some of the limitations of existing job search platforms by providing job seekers with a centralized platform that offers a wide range of services. The platform aims to be user-friendly, continually updated, and tailored to meet the specific needs of each user.

#### **Future Works:**

In the future, the Career Management and Service Portal can be enhanced by adding additional features such as networking opportunities, career coaching services, and access to mentorship programs. The platform can also incorporate artificial intelligence (AI) and machine learning (ML) technologies to provide personalized recommendations to job seekers, based on their skills, experience, and career goals. Additionally, the platform can be expanded to cover a wider range of industries and professions to cater to the needs of a broader audience. Finally, partnerships with employers and recruitment agencies can be established to provide job seekers with access to exclusive job listings and career opportunities.

**Github link: <https://github.com/dc0407/WDD-PROJECT-CMPS-PORTAL>**

## References

- I. Anderson, C. (2019). The ultimate guide to job interview answers. Kindle Edition.
- II. Barron's Educational Series. (2017). Barron's Mechanical Aptitude and Spatial Relations Test, 4th Edition.
- III. O'Brien, J. (2018). Resumes for Dummies, 8th Edition.
- IV. Stone, J. (2019). Cover Letters for Dummies, 4th Edition.
- V. Weide, M. (2019). Mastering the Job Interview: The Definitive Guide to Successful Job Interviewing. Kindle Edition.
- VI. The Balance Careers. (2021). Job Interview Questions and Answers. Retrieved from <https://www.thebalancecareers.com/job-interview-questions-and-answers-2061204>
- VII. The Muse. (2021). 31 Attention-Grabbing Cover Letter Examples. Retrieved from <https://www.themuse.com/advice/31-attentiongrabbing-cover-letter-examples>
- VIII. TopResume. (2021). Resume Examples by Industry. Retrieved from <https://www.topresume.com/resume-examples>
- IX. Wall Street Journal. (2021). How to Ace a Job Interview: What to Wear, What to Bring, and Other Tips. Retrieved from <https://www.wsj.com/articles/how-to-ace-a-job-interview-what-to-wear-what-to-bring-and-other-tips-11617677400>
- X. Workopolis. (2021). How to Write a Cover Letter: 10+ Cover Letter Tips. Retrieved from <https://careers.workopolis.com/advice/how-to-write-a-cover-letter-10-cover-letter-tips/>