Department of Computer Engineering & Applications

**GLA UNIVERSITY**



# Res-X

A PROJECT REPORT

***Submitted by***

Apoorv Vikram Singh Rathore(201500136)

Suyash Prakash Singh(201500730)

Bhoomi Kamakhya (201500193)

Nishant Tiwari (201500445)

Ritu Kumari (201500580)

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

# BACHELOR OF ENGINEERING

**IN**

Computer Engineering and Application

**GLA University, Mathura**

# BONAFIDE CERTIFICATE

Certified that this project report **“Res-X”** is the bonafide work of

## “APOORV VIKRAM SINGH RATHORE, SUYASH PRATAP SINGH, Bhoomi Kamakhya ,Nishant Tiwari, Ritu Kumari ” who carried out the project work under my/our supervision.

## 

**SIGNATURE SIGNATURE**

Rohit Agrawal Mr. Mandeep Singh

**SUPERVISOR**

**HEAD OF THE DEPARTMENT** Technical Trainer

Computer Engineering and Application Training and Development

Submitted for the project viva-voce examination held on

**INTERNAL EXAMINER EXTERNAL EXAMINER**

# ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the End Term Report of the B.Tech mini project undertaken during B.Tech III Year. This project is going to be an acknowledgement to the inspiration, drive and technical assistance will be contributed to it by many individuals. We owe special debt of gratitude to Mr. Mandeep Singh , Technical Trainer, for providing us with an encouraging platform to develop this project, which thus helped us in shaping our abilities towards a constructive goal and for his constant support and guidance to our work. His sincerity, thoroughness and perseverance has been a constant source of inspiration for us. We believe that he will shower us with all his extensively experienced ideas and insightful comments at different stages of the project & also taught us about the latest industry-oriented technologies. We also do not like miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind guidance and co-operation.

Apoorv Vikram Singh Rathore (201500136)

Suyash Prakash Singh (201500730)

Bhoomi Kamakhya (201500193)

Nishant Tiwari (201500445)

Ritu Kumari (201500580)

TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| S.No | TOPIC |  |
| 1. | Abstract |  |
| 2. | Chapter-1..Introduction |  |
| 3. | 1.1..Identification of Task |  |
| 4. | 1.2..Main Objective of project |  |
| 5. | 1.3..Scope of the project |  |
| 6. | Chapter-2..Design Flow/Process |  |
| 7. | 2.1..Working methodology of the project |  |
| 8. | Chapter-3..Sections and Code Implementation |  |
| 9. | Chapter-4..Conclusion & Future Work |  |
| 10. | 4.1..Conclusion |  |
| 11. | 4.2..Future Work |  |
| 12. | References |  |

# ABSTRACT

A resume builder is a digital tool that helps job seekers create professional resumes that effectively highlight their skills, experience, and qualifications. It typically features an easy-to-use interface that guides users through the process of selecting a template, entering their information, and customizing the design and layout of their resume. A good resume builder will allow users to customize their resumes based on their career goals, and may offer suggestions for improving the content or formatting. Some resume builders also include features such as spell-checking, keyword optimization, and the ability to save and share resumes in various formats. The overall goal of a resume builder is to simplify the job application process and help job seekers stand out from other candidates by presenting their qualifications in the best possible light.

**CHAPTER 1**

**Introduction**

The “Online Resume Builder” has been developed to override the problems prevailing in the

practicing manual system. This software is supported to eliminate and, in some cases, reduce the

hardships faced by this existing system. Moreover, this system is designed for the particular need of

the user to carry out operations in a smooth and effective manner.

**About the Project**

An online resume builder is a software developed to simplify the task of creating a resume for

individuals. The application provides an effective means of designing desired resume in fact a

professional looking resume. The system is flexible to be used and reduces the need of thinking and

designing an appropriate resume according to qualifications. Usually, individuals get confused while

creating a resume especially for a novice person such as graduate students. They don’t get a clear

idea of what things and information must be included in a resume. Hence the system is developed to

provide them an easy way for creating a professional looking resume.

This project is user-friendly and requires minimum human intervention. Individuals just have to fill

up a form that specifies questions from all required fields such as personal questions, educational,

qualities, interest, skills and so on. The answers provided by the users are stored and the system

automatically generates a well-structured resume. Users have option to create resume in any format

and file.

**1.1 Primary Reason to choose This Project: -**

The primary reason to choose a resume builder website is to create a professional-looking resume quickly and easily. A resume builder website typically offers a range of customizable templates that are designed to showcase a job seeker's experience, skills, and achievements in the best possible way. These templates are often created by professional resume writers or graphic designers, so they are visually appealing and easy to read. In addition, a good resume builder website will guide users through the process of creating a resume step-by-step, so they can be sure that they are including all the necessary information and formatting their resume correctly. Using a resume builder website can save job seekers time and effort, and can increase their chances of landing an interview by presenting their qualifications in a clear and professional manner.

**1.2 The Main Objective of the project: -**

The main objective of the Project on Online Resume Builder is to manage the details of Resume,

Job, Qualification, Skill, Jobseeker. It manages all the information about Resume, Individual,

Jobseeker, Resume. The project is totally built at administrative end and thus only the administrator

is guaranteed the access. The purpose of the project is to build an application program to reduce the

manual work for managing the Resume, Job, Individual, Qualification. It tracks all the details about

the Qualification, Skill, Jobseeker.

**1.3 Scope of the project: -**

 Online Resume Builder also manage the Individual details online for Skill details, Jobseeker

details, Resume.

 It tracks all the information of Job, Individual, Skill etc

 Manage the information of Job

 Shows the information and description of the Resume, Qualification

 To increase efficiency of managing the Resume, Job

 It deals with monitoring the information and transactions of Skill

 Manage the information of Resume Builder

**CHAPTER 2**

**Working**

**2.1 Working Methodology of the project: -**

**System Requirements: -**

 Supported Operating system: - Any

 Software Required: - Visual Studio Code

**Hardware Requirements: -**

It can run on any machine having decent specification

with a good internet connection.

As such no specific high-end hardware is required.

**Listing Out testing technology: -**

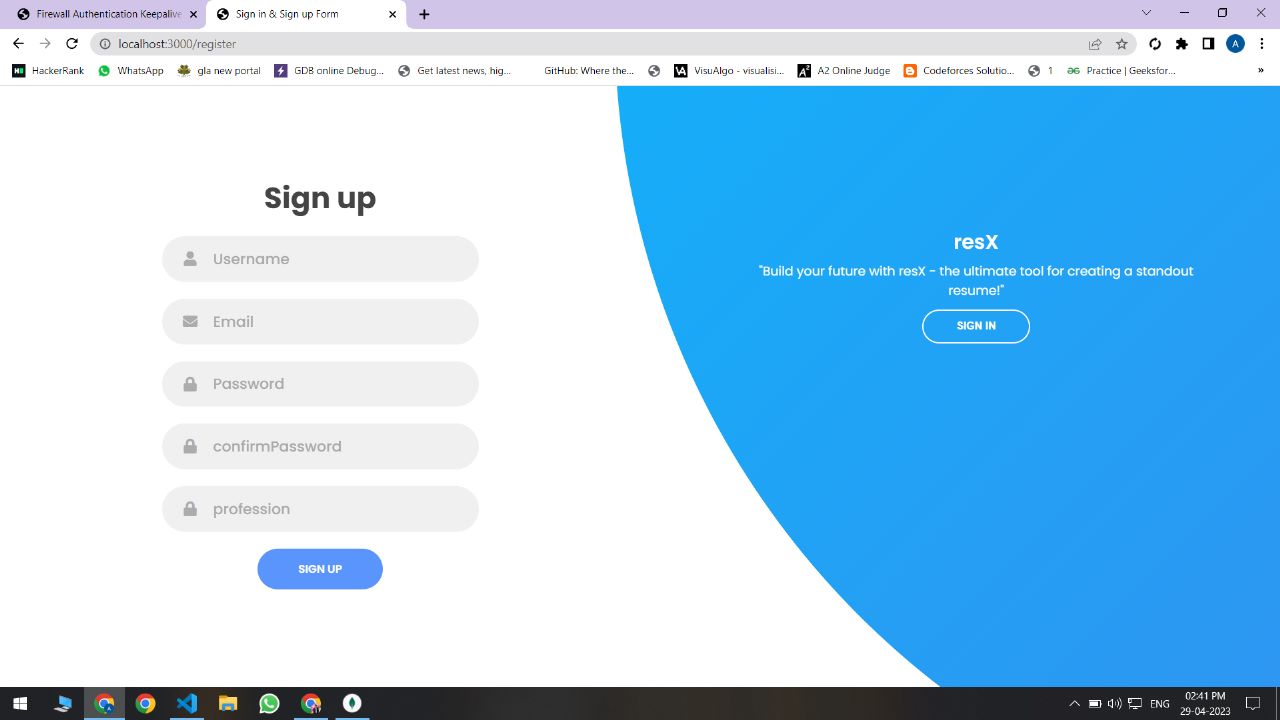
Frontend: - HTML, CSS, Java Script, Bootstrap

Backend: - Mongo DB, Node JS

**CHAPTER 3**

**3. Code Implementation**

**Sign up page**



**Code for sign up page**

<!DOCTYPE html>

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

<head>

<meta charset="utf-8">

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/css/all.min.css"/>

<style>

@import url('https://fonts.googleapis.com/css?family=Poppins:400,500,600,700&display=swap');

\*{

margin: 0;

padding: 0;

/\* user-select: none; \*/

box-sizing: border-box;

font-family: 'Poppins', sans-serif;

}

html,body{

height: 100%;

}

body{

display: grid;

place-items: center;

background: #dde1e7;

text-align: center;

background-image: url(./photo/3d-abstract-background-with-paper-cut-shape\_1217-3284.webp);

}

.content{

width: 500px;

padding: 40px 30px;

background: #dde1e7;

border-radius: 10px;

box-shadow: -3px -3px 7px #ffffff73,

2px 2px 5px rgba(94,104,121,0.288);

}

.content .text{

font-size: 33px;

font-weight: 600;

margin-bottom: 35px;

color: #595959;

}

.field{

height: 50px;

width: 100%;

display: flex;

position: relative;

}

.field:nth-child(2){

margin-top: 20px;

}

.field input{

height: 100%;

width: 100%;

padding-left: 45px;

outline: none;

border: none;

font-size: 18px;

background: #dde1e7;

color: #595959;

border-radius: 25px;

box-shadow: inset 2px 2px 5px #BABECC,

inset -5px -5px 10px #ffffff73;

}

.field input:focus{

box-shadow: inset 1px 1px 2px #BABECC,

inset -1px -1px 2px #ffffff73;

}

.field span{

position: absolute;

color: #595959;

width: 50px;

line-height: 50px;

}

.field label{

position: absolute;

top: 50%;

transform: translateY(-50%);

left: 45px;

pointer-events: none;

color: #666666;

}

.field input:valid ~ label{

opacity: 0;

}

.forgot-pass{

text-align: left;

margin: 10px 0 10px 5px;

}

.forgot-pass a{

font-size: 16px;

color: #3498db;

text-decoration: none;

}

.forgot-pass:hover a{

text-decoration: underline;

}

button{

margin: 15px 0;

width: 100%;

height: 50px;

font-size: 18px;

line-height: 50px;

font-weight: 600;

background: #dde1e7;

border-radius: 25px;

border: none;

outline: none;

cursor: pointer;

color: #595959;

box-shadow: 2px 2px 5px #BABECC,

-5px -5px 10px #ffffff73;

}

button:focus{

color: #3498db;

box-shadow: inset 2px 2px 5px #BABECC,

inset -5px -5px 10px #ffffff73;

}

.sign-up{

margin: 10px 0;

color: #595959;

font-size: 16px;

}

.sign-up a{

color: #3498db;

text-decoration: none;

}

.sign-up a:hover{

text-decoration: underline;

}

</style>

</head>

<body>

<div class="content">

<div class="text">

Login Form

</div>

<form action="/index" method="POST">

<div class="field">

<input type="text" required name="email">

<span class="fas fa-user"></span>

<label>Email</label>

</div>

<div class="field">

<input type="password" required name="password">

<span class="fas fa-lock"></span>

<label>Password</label>

</div>

<button type="submit" name="submit">Log in</button>

<div class="sign-up">

Not a member?

<a href="/register">signup now</a>

</div>

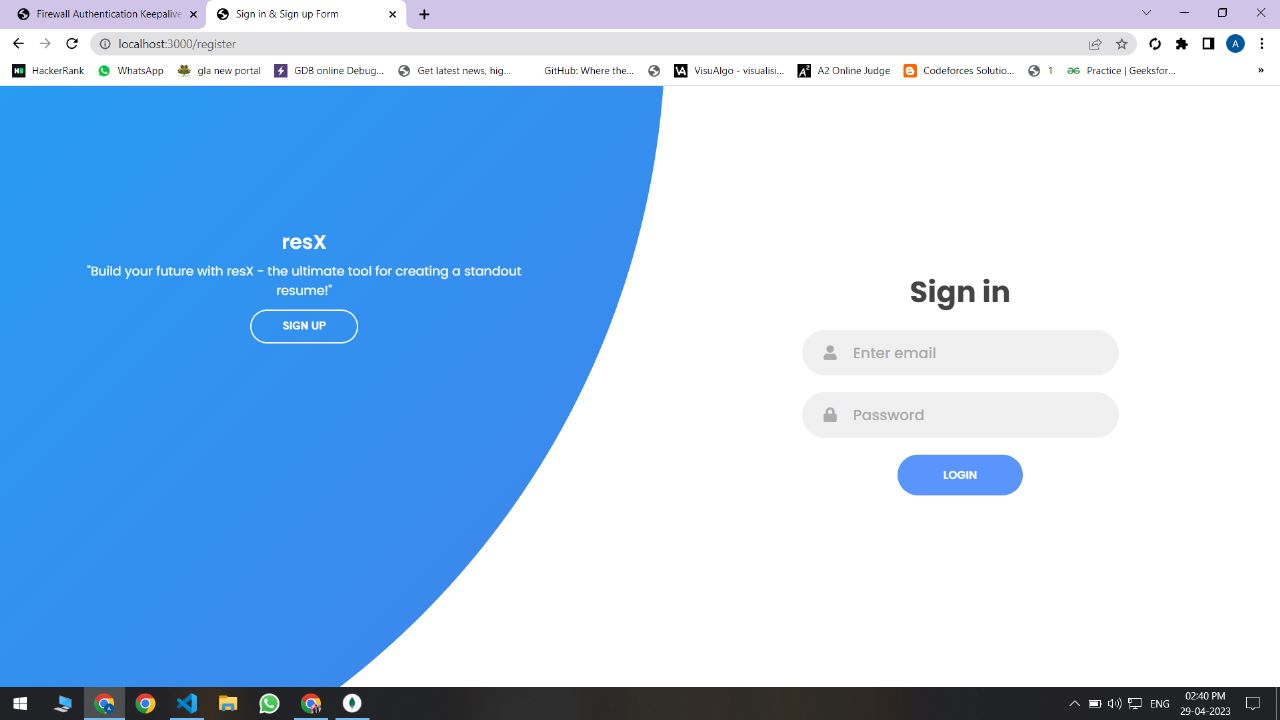
</form>

</div>

</body>

</html>

**Login page**



**Code for login page**

<form method="POST" action="/login">

<h1>Sign in</h1>

<p>to use your account</p>

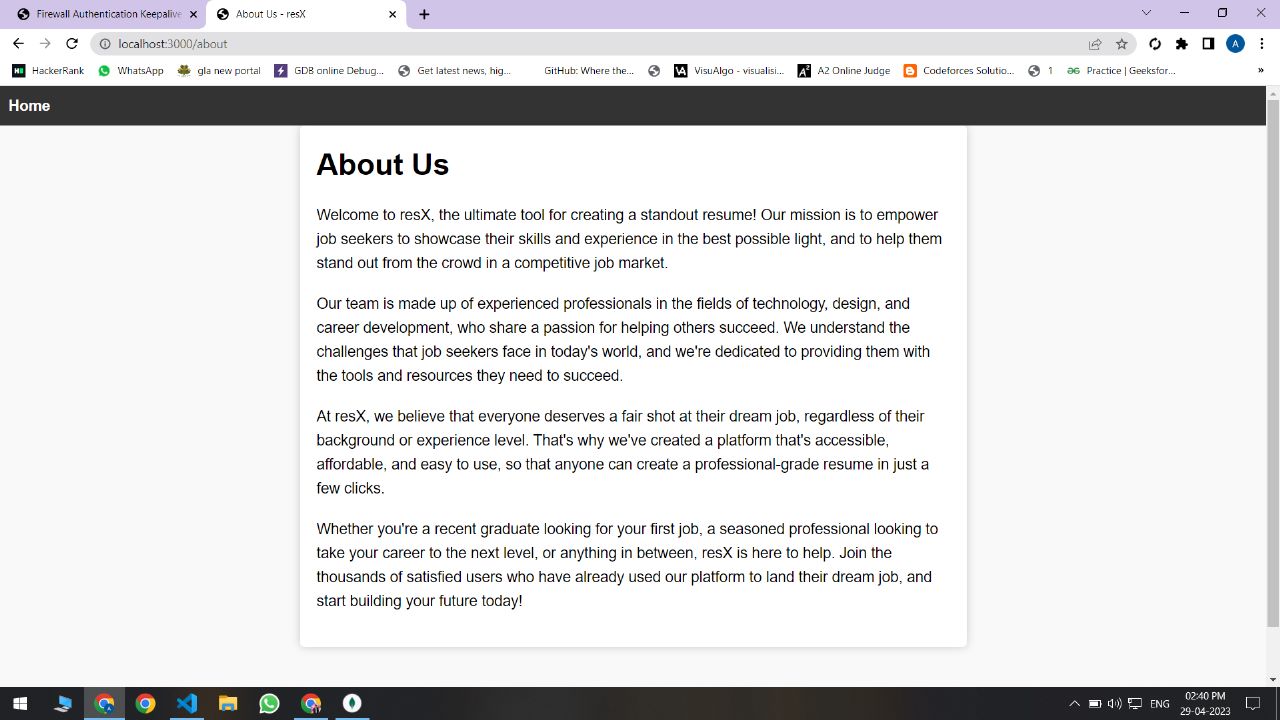
<input type="email" name="email" placeholder="Email" required="">

<input type="password" name="pswd" placeholder="Password" required="">

<button>Sign in</button>

</form>

**About Us Page**



**Code for About us page**

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>About Us - resX</title>

<style>

\* {

box-sizing: border-box;

margin: 0;

padding: 0;

}

body {

font-family: Arial, sans-serif;

font-size: 16px;

line-height: 1.5;

background-color: #f9f9f9;

}

header {

background-color: #333;

color: #fff;

padding: 10px;

display: flex;

justify-content: space-between;

align-items: center;

}

nav a {

color: #fff;

text-decoration: none;

margin-left: 20px;

font-size: 18px;

font-weight: bold;

}

nav a:hover {

color: #f5c400;

}

nav a:first-child {

margin-left: 0;

}

main {

max-width: 800px;

margin: 0 auto;

padding: 20px;

background-color: #fff;

box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.2);

border-radius: 5px;

}

h1 {

font-size: 36px;

margin-bottom: 20px;

}

p {

margin-bottom: 20px;

line-height: 1.6;

font-size: 18px;

}

ul {

list-style: disc;

margin-left: 20px;

margin-bottom: 20px;

}

li {

margin-bottom: 10px;

}

footer {

background-color: #333;

color: #fff;

text-align: center;

padding: 10px;

margin-top: 50px;

}

footer p {

font-size: 14px;

}

@media screen and (max-width: 600px) {

nav {

flex-direction: column;

align-items: flex-start;

}

nav a {

margin: 10px 0;

}

}

\* {

box-sizing: border-box;

margin: 0;

padding: 0;

}

body {

font-family: Arial, sans-serif;

font-size: 16px;

line-height: 1.5;

background-color: #f9f9f9;

}

header {

background-color: #333;

color: #fff;

padding: 10px;

display: flex;

justify-content: space-between;

align-items: center;

}

nav a {

color: #fff;

text-decoration: none;

margin-left: 20px;

font-size: 18px;

font-weight: bold;

}

nav a:hover {

color: #f5c400;

}

nav a:first-child {

margin-left: 0;

}

main {

max-width: 800px;

margin: 0 auto;

padding: 20px;

background-color: #fff;

box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.2);

border-radius: 5px;

}

h1 {

font-size: 36px;

margin-bottom: 20px;

}

p {

margin-bottom: 20px;

line-height: 1.6;

font-size: 18px;

}

ul {

list-style: disc;

margin-left: 20px;

margin-bottom: 20px;

}

li {

margin-bottom: 10px;

}

footer {

background-color: #333;

color: #fff;

text-align: center;

padding: 10px;

margin-top: 50px;

}

footer p {

font-size: 14px;

}

@media screen and (max-width: 600px) {

nav {

flex-direction: column;

align-items: flex-start;

}

nav a {

margin: 10px 0;

}

}

\* {

box-sizing: border-box;

margin: 0;

padding: 0;

}

body {

font-family: Arial, sans-serif;

font-size: 16px;

line-height: 1.5;

background-color: #f9f9f9;

}

header {

background-color: #333;

color: #fff;

padding: 10px;

display: flex;

justify-content: space-between;

align-items: center;

}

nav a {

color: #fff;

text-decoration: none;

margin-left: 20px;

font-size: 18px;

font-weight: bold;

}

nav a:hover {

color: #f5c400;

}

nav a:first-child {

margin-left: 0;

}

main {

max-width: 800px;

margin: 0 auto;

padding: 20px;

background-color: #fff;

box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.2);

border-radius: 5px;

}

h1 {

font-size: 36px;

margin-bottom: 20px;

}

p {

margin-bottom: 20px;

line-height: 1.6;

font-size: 18px;

}

ul {

list-style: disc;

margin-left: 20px;

margin-bottom: 20px;

}

li {

margin-bottom: 10px;

}

footer {

background-color: #333;

color: #fff;

text-align: center;

padding: 10px;

margin-top: 50px;

}

footer p {

font-size: 14px;

}

@media screen and (max-width: 600px) {

nav {

flex-direction: column;

align-items: flex-start;

}

nav a {

margin: 10px 0;

}

}

</style>

</head>

<body>

<header>

<nav>

<a href="/">Home</a>

</nav>

</header>

<main>

<h1>About Us</h1>

<p>Welcome to resX, the ultimate tool for creating a standout resume! Our mission is to empower job seekers to showcase their skills and experience in the best possible light, and to help them stand out from the crowd in a competitive job market.</p>

<p>Our team is made up of experienced professionals in the fields of technology, design, and career development, who share a passion for helping others succeed. We understand the challenges that job seekers face in today's world, and we're dedicated to providing them with the tools and resources they need to succeed.</p>

<p>At resX, we believe that everyone deserves a fair shot at their dream job, regardless of their background or experience level. That's why we've created a platform that's accessible, affordable, and easy to use, so that anyone can create a professional-grade resume in just a few clicks.</p>

<p>Whether you're a recent graduate looking for your first job, a seasoned professional looking to take your career to the next level, or anything in between, resX is here to help. Join the thousands of satisfied users who have already used our platform to land their dream job, and start building your future today!</p>

</main>

<footer>

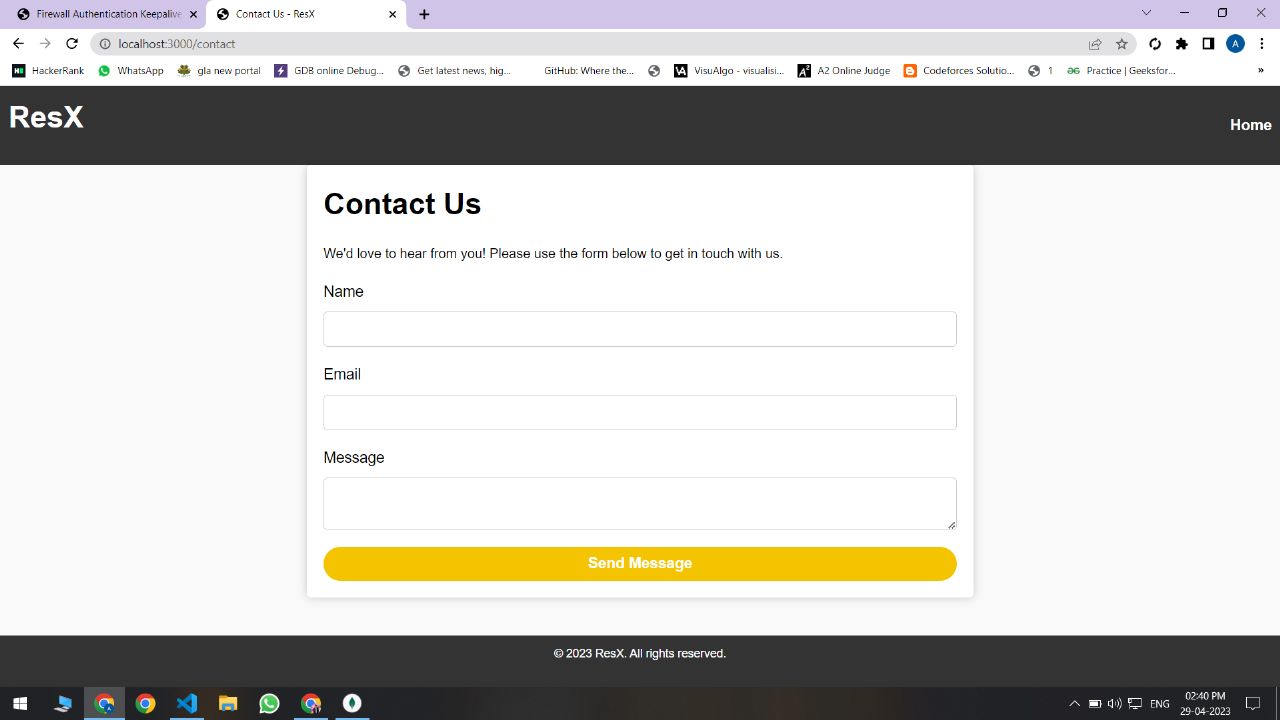
<p>&copy; 2023 resX. All rights reserved.</p>

</footer>

</body>

</html>

**Contact Us Page**



**Code for Contact Us Page**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Contact Us - ResX</title>

<style>

\* {

box-sizing: border-box;

margin: 0;

padding: 0;

}

body {

font-family: Arial, sans-serif;

font-size: 16px;

line-height: 1.5;

background-color: #f9f9f9;

}

header {

background-color: #333;

color: #fff;

padding: 10px;

display: flex;

justify-content: space-between;

align-items: center;

}

nav a {

color: #fff;

text-decoration: none;

margin-left: 20px;

font-size: 18px;

font-weight: bold;

}

nav a:hover {

color: #f5c400;

}

nav a:first-child {

margin-left: 0;

}

main {

max-width: 800px;

margin: 0 auto;

padding: 20px;

background-color: #fff;

box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.2);

border-radius: 5px;

}

h1 {

font-size: 36px;

margin-bottom: 20px;

}

p {

margin-bottom: 20px;

line-height: 1.6;

}

form {

display: flex;

flex-direction: column;

}

label {

font-size: 18px;

margin-bottom: 10px;

}

input, textarea {

padding: 10px;

margin-bottom: 20px;

border-radius: 5px;

border: 1px solid #ccc;

font-size: 18px;

font-family: Arial, sans-serif;

}

button {

padding: 10px 20px;

border-radius: 30px;

background-color: #f5c400;

color: #fff;

border: none;

font-size: 18px;

font-weight: bold;

cursor: pointer;

transition: background-color 0.2s ease-in-out;

}

button:hover {

background-color: #e0b700;

}

button:active {

transform: scale(0.98);

background-color: #d3a800;

}

input:focus, textarea:focus {

outline: none;

border-color: #f5c400;

box-shadow: 0px 0px 5px rgba(245, 196, 0, 0.2);

}

footer {

background-color: #333;

color: #fff;

text-align: center;

padding: 10px;

position: fixed;

bottom: 0;

width: 100%;

}

footer p {

font-size: 14px;

}

</style>

</head>

<body>

<header>

<h1>ResX</h1>

<nav>

<a href="/">Home</a>

</nav>

</header>

<main>

<h1>Contact Us</h1>

<p>We'd love to hear from you! Please use the form below to get in touch with us.</p>

<form>

<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="message">Message</label>

<textarea id="message" name="message" required></textarea>

<button type="submit">Send Message</button>

</form>

</main>

<footer>

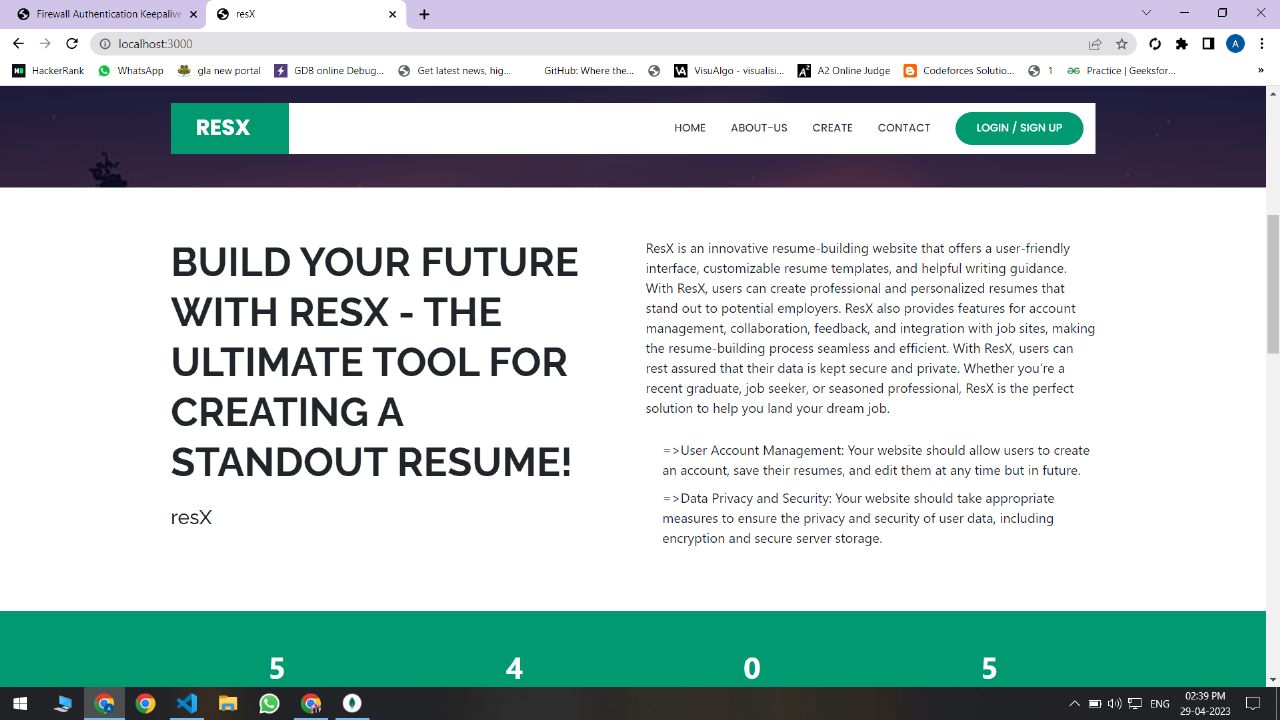
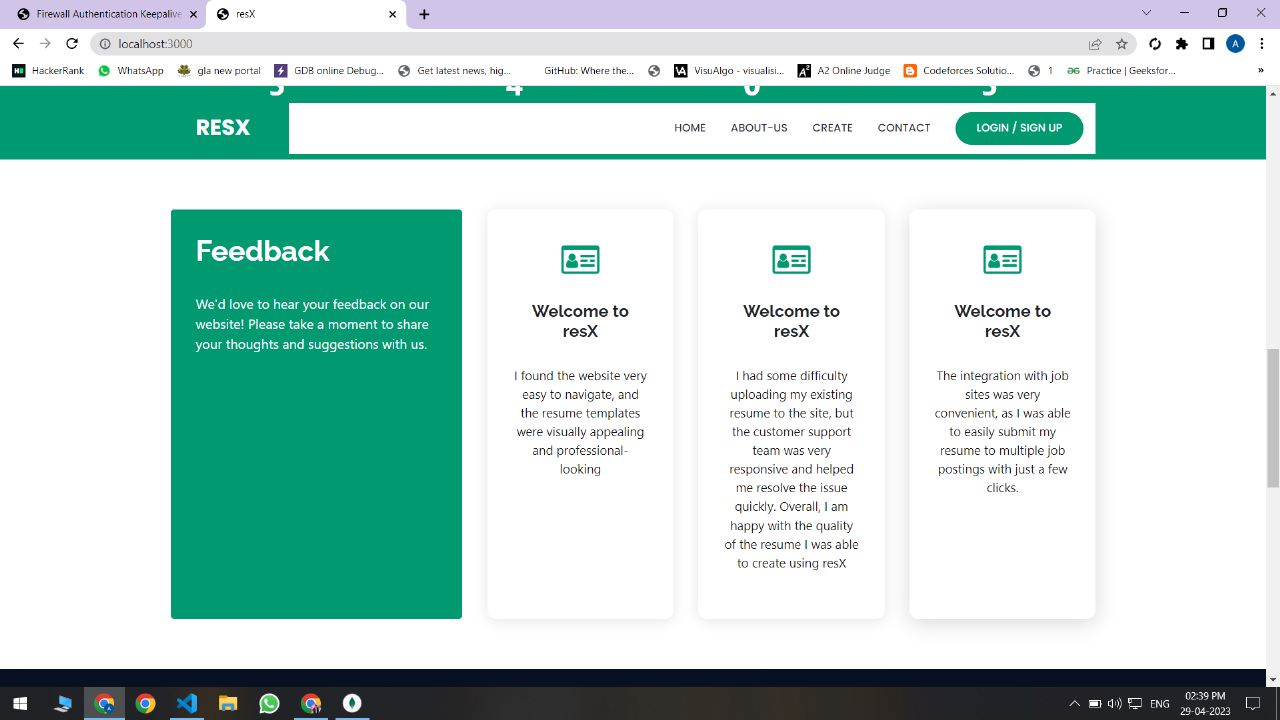
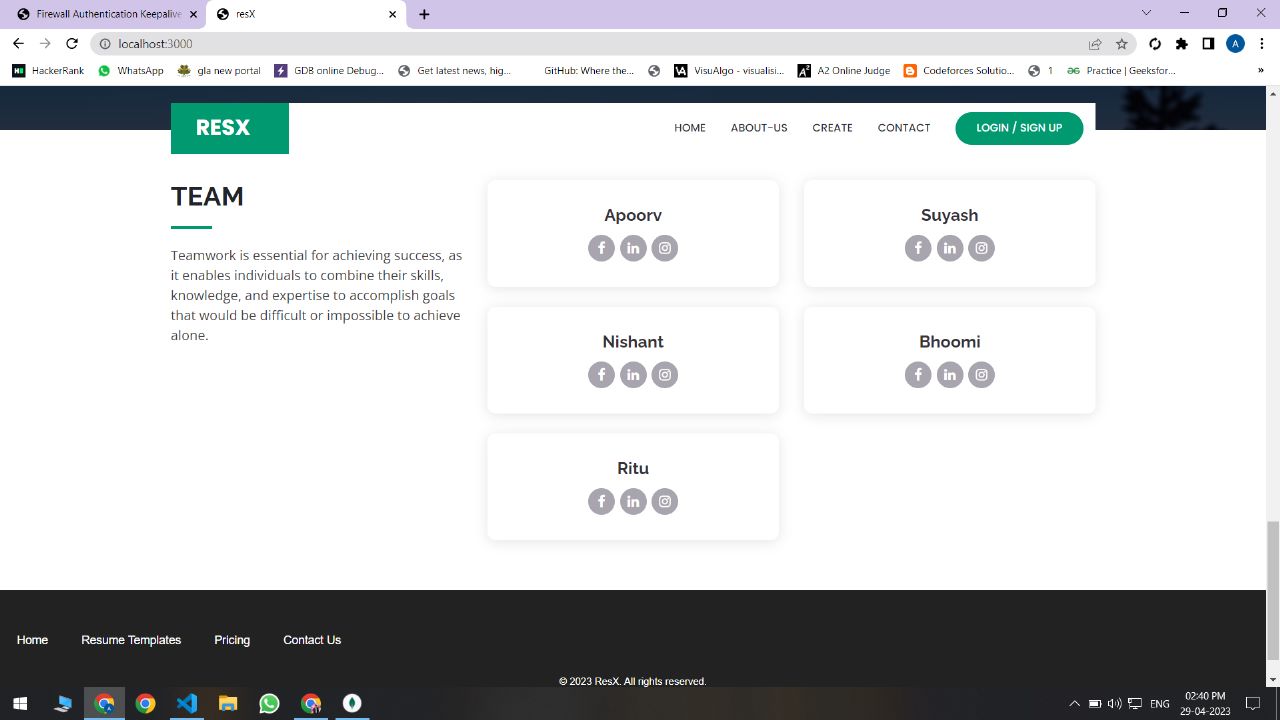
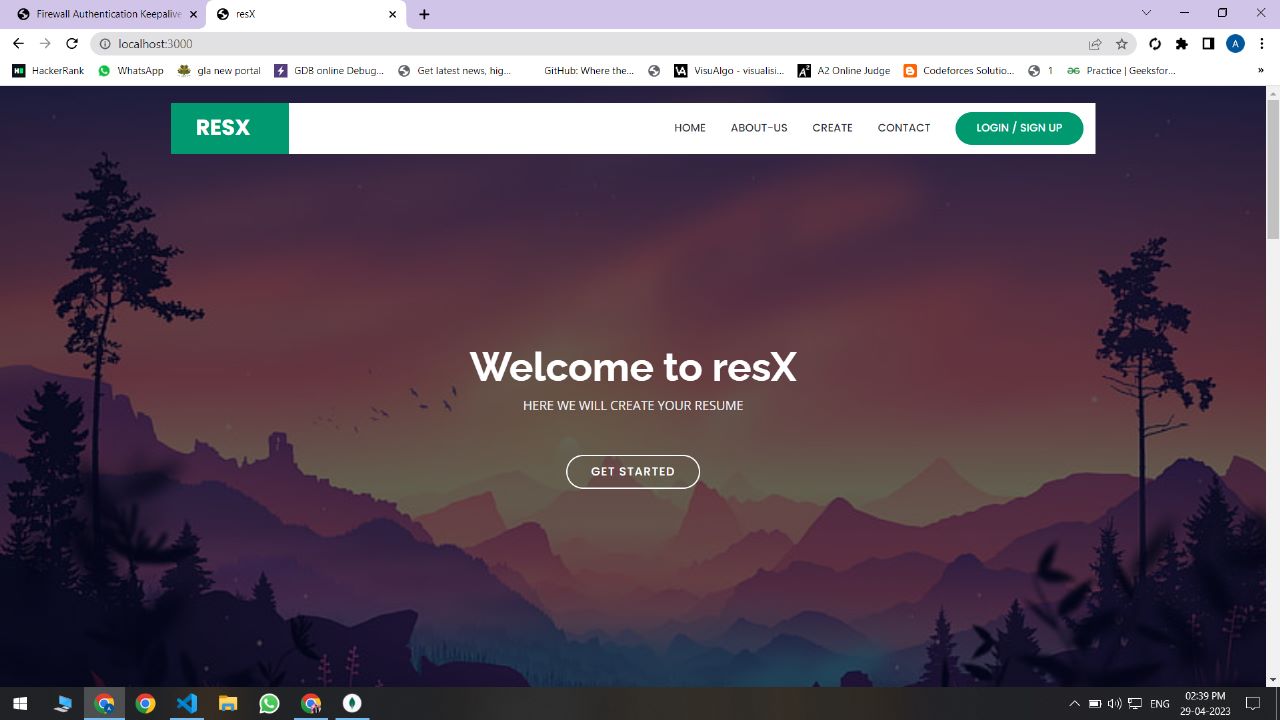
<p>&copy; 2023 ResX. All rights reserved.</p>

</footer>

</body>

</html>

**Home page**



**Code for home page**

<!doctype html>

<html lang="en">

<head>

<!-- Required meta tags -->

<meta charset="utf-8">

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

<!-- Bootstrap CSS -->

<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/css/bootstrap.min.css"

integrity="sha384-TX8t27EcRE3e/ihU7zmQxVncDAy5uIKz4rEkgIXeMed4M0jlfIDPvg6uqKI2xXr2" crossorigin="anonymous">

<title>Registration Form</title>

<style>

/\* Style the page header \*/

h1 {

text-align: center;

color: #009970;

font-size: 48px;

margin-top: 40px;

}

h2 {

text-align: center;

color: #555;

font-size: 36px;

margin-top: 40px;

}

/\* Style the container \*/

.container {

max-width: 800px;

margin: 0 auto;

padding: 20px;

border: 1px solid #ddd;

border-radius: 10px;

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

}

/\* Style the form \*/

form {

margin-top: 30px;

}

/\* Style the form labels \*/

form label {

display: block;

margin-bottom: 10px;

font-weight: bold;

color: #555;

}

/\* Style the form inputs \*/

form input[type="text"],

form input[type="email"] {

width: 100%;

padding: 10px;

margin-bottom: 20px;

border: 1px solid #ddd;

border-radius: 5px;

}

/\* Style the checkboxes \*/

form input[type="checkbox"] {

display: inline-block;

margin-right: 10px;

}

/\* Style the submit button \*/

form button[type="submit"] {

display: block;

margin: 0 auto;

padding: 10px 30px;

background-color: #009970;

color: #fff;

border: none;

border-radius: 50px;

font-family: 'Open Sans', sans-serif;

font-size: 18px;

cursor: pointer;

transition: background-color 0.3s ease-in-out;

}

form button[type="submit"]:hover {

background-color: #00805d;

}

.kamm {

display: flex;

align-items: center;

margin-bottom: 10px; /\* add some margin to separate the rows \*/

}

.hai {

display: flex;

margin-left: 10px;

}

.hai label {

margin-right: 10px;

}

/\* Add this new rule to align the rows \*/

.kamm:nth-of-type(2) {

margin-top: 10px;

}

</style>

</head>

<body>

<h1>ResX</h1>

<h2>Enter Your Details</h2>

<div class="container">

<form action="/create" method="POST" enctype="multipart/form-data">

<div class="form-row">

<div class="form-group col-md-6">

<label for="inputEmail4">Enter your full Name</label>

<input type="text" class="form-control" id="inputName" placeholder="Name" name="namec" required>

</div>

<div class="form-group col-md-6">

<label for="inputPassword4">Preferd job location</label>

<input type="text" class="form-control" id="inputABC" placeholder="Enter your Preferd joblocation" name="locationc" required>

</div>

</div>

<div class="kamm">

<label for="hobbies">Select your hobbies:</label>

<div class="hai">

<input type="checkbox" id="hobby1" name="hobby1" value="reading">

<label for="hobby1">Reading</label>

<input type="checkbox" id="hobby2" name="hobby1" value="writing">

<label for="hobby2">Writing</label>

<input type="checkbox" id="hobby3" name="hobby1" value="painting">

<label for="hobby3">Painting</label>

<input type="checkbox" id="hobby4" name="hobby1" value="coding">

<label for="hobby3">coding</label>

<input type="checkbox" id="hobby5" name="hobby1" value="playing">

<label for="hobby3">playing</label>

</div>

</div>

<div class="kamm">

<label for="hobbies">Select your SKILLS:</label>

<div class="hai">

<input type="checkbox" id="hobby1" name="skill1" value="Problem-solving">

<label for="hobby1">Problem-solving</label>

<input type="checkbox" id="hobby2" name="skill1" value="Communication">

<label for="hobby2">Communication</label>

<input type="checkbox" id="hobby3" name="skill1" value="Adaptability">

<label for="hobby3">Adaptability</label>

<input type="checkbox" id="hobby4" name="skill1" value="Project-management">

<label for="hobby3">Project-management</label>

<input type="checkbox" id="hobby5" name="skill1" value="Adaptability">

<label for="hobby3">Adaptability</label>

</div>

</div>

<div class="form-row">

<div class="form-group col-md-6">

{{!-- <label for="inputEmail4">Enter your official email</label> --}}

<input type="email" class="form-control" id="inputEmail" placeholder="Email" name="emailc" hidden>

</div>

<div class="form-group col-md-6">

<label for="inputPassword4">Phone Number</label>

<input type="text" class="form-control" id="inputphone" placeholder="enter your valid phone number" name="phonec" required pattern="[0-9]{10}">

</div>

</div>

<div class="form-row">

<div class="form-group col-md-6">

<label for="ABC">College Name</label>

<input type="text" class="form-control" id="abc" placeholder="Enter your College name" name="collegec" minlength="15" required>

</div>

<div class="form-group col-md-6">

<label for="xyz">job Title</label>

<input type="text" class="form-control" id="xyz" placeholder="Enter your stream of graduation" name="streamc" required>

</div>

</div>

<div>

<div class="form-group">

<label for="Projects">About Yourself</label>

<textarea class="form-control" id="exampleFormControlTextarea1" rows="3" placeholder="Describe abotu the projects" name="about" required></textarea>

</div>

<div class="form-group">

<label for="inputAddress">Address</label>

<input type="text" class="form-control" id="inputAddress" placeholder="city,state" name="addressc" required maxlength="17">

</div>

<div class="form-group">

<div id="project-list">

<div class="project-item">

<label for="project\_title1">Project Title:</label>

<input type="text" id="project\_title1" name="projecttitle1" placeholder="Enter project title" required maxlength="10">

<label for="project\_description1">Project Description:</label>

<input id="project\_description1" name="projectdescription1" placeholder="Enter project description" required>

</div>

</div>

<button type="button" onclick="addProject()">Add Project</button>

</div>

<div class="form-group">

<div id="education-list">

<div class="project-item">

<label for="year">EDUCATION Year:</label>

<input type="text" class="form-control" name="year1" placeholder="Enter year" required pattern="[0-9]{4}-[0-9]{4}">

<label for="year">EDUCATION Class:</label>

<input type="text" class="form-control" name="class" placeholder="Enter year" required>

<label for="company">School:</label>

<input class="form-control" name="school" placeholder="about school" required>

<label for="desc">Marks:</label>

<input class="form-control" name="marks" placeholder="marks">

</div>

</div>

<button type="button" onclick="addProo()">Add EDUCATION</button>

</div>

<div class="form-group">

<div id="exprience-list">

<div class="project-item">

<label for="year">EXPERIENCE Year:</label>

<input type="text" class="form-control" name="year" placeholder="Enter year" required pattern="[0-9]{4}-[0-9]{4}">

<label for="company">EXPERIENCE company:</label>

<input class="form-control" name="company" placeholder="about company" required>

<label for="desc">EXPERIENCEt Description:</label>

<input class="form-control" name="desc" placeholder="about describe about them" required>

</div>

</div>

<button type="button" onclick="addPro()">Add EXPERIENCE</button>

</div>

<label for="template">Select a template:</label>

<select name="template" id="template">

<option value="template1">Template 1</option>

<option value="template2">Template 2</option>

<option value="template3">Template 3</option>

<option value="template4">Template 4</option>

</select>

<br>

{{!--

<div class="form-group">

<label for="Addtional">Addtional</label>

<textarea class="form-control" id="exampleFormControlTextarea1" rows="4" placeholder="Addtional information" name="addtionalc"></textarea>

</div> --}}

<div class="form-group">

<label for="Your Image">Your Image :</label>

<input type="file" id="profile" name="imagec" accept="image/\*">

</div>

<div class="text-center">

<button type="submit" name="submitc">Submit</button>

</div>

</form></div>

<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"

integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj"

crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/js/bootstrap.bundle.min.js"

integrity="sha384-ho+j7jyWK8fNQe+A12Hb8AhRq26LrZ/JpcUGGOn+Y7RsweNrtN/tE3MoK7ZeZDyx"

crossorigin="anonymous"></script>

<script>

let projectCount=1;

function addProject() {

const projectList = document.getElementById('project-list');

projectCount++;

const projectDiv = document.createElement('div');

projectDiv.innerHTML = '<label for="project\_title1">Project Title:</label>\

<input type="text" id="project\_title1" name="projecttitle1" placeholder="Enter project title">\

<label for="project\_description1">Project Description:</label>\

<input id="project\_description1" name="projectdescription1" placeholder="Enter project description"></input>';

projectList.appendChild(projectDiv);

}

let EXPERIENCECOunt=1;

function addPro(){

const experincetList = document.getElementById('exprience-list');

EXPERIENCECOunt++;

const exp = document.createElement('div');

exp.innerHTML = ' <label for="year">EXPERIENCE Year:</label>\

<input type="text" class="form-control" name="year1" placeholder="Enter year">\

<label for="company">EXPERIENCE company:</label>\

<input class="form-control" name="company" placeholder="about company">\

<label for="desc">EXPERIENCEt Description:</label>\

<input class="form-control" name="desc" placeholder="about describe about them">';

experincetList.appendChild(exp);

}

let EXPERIENCECOuntt=1;

function addProo(){

const educationtList = document.getElementById('education-list');

EXPERIENCECOuntt++;

const edu = document.createElement('div');

edu.innerHTML = ' <label for="year">EDUCATION Year:</label>\

<input type="text" class="form-control" name="year" placeholder="Enter year">\

<label for="year">EDUCATION Class:</label>\

<input type="text" class="form-control" name="class" placeholder="Enter year">\

<label for="company">School:</label>\

<input class="form-control" name="school" placeholder="about school">\

<label for="desc">Marks:</label>\

<input class="form-control" name="marks" placeholder="marks">';

educationtList.appendChild(edu);

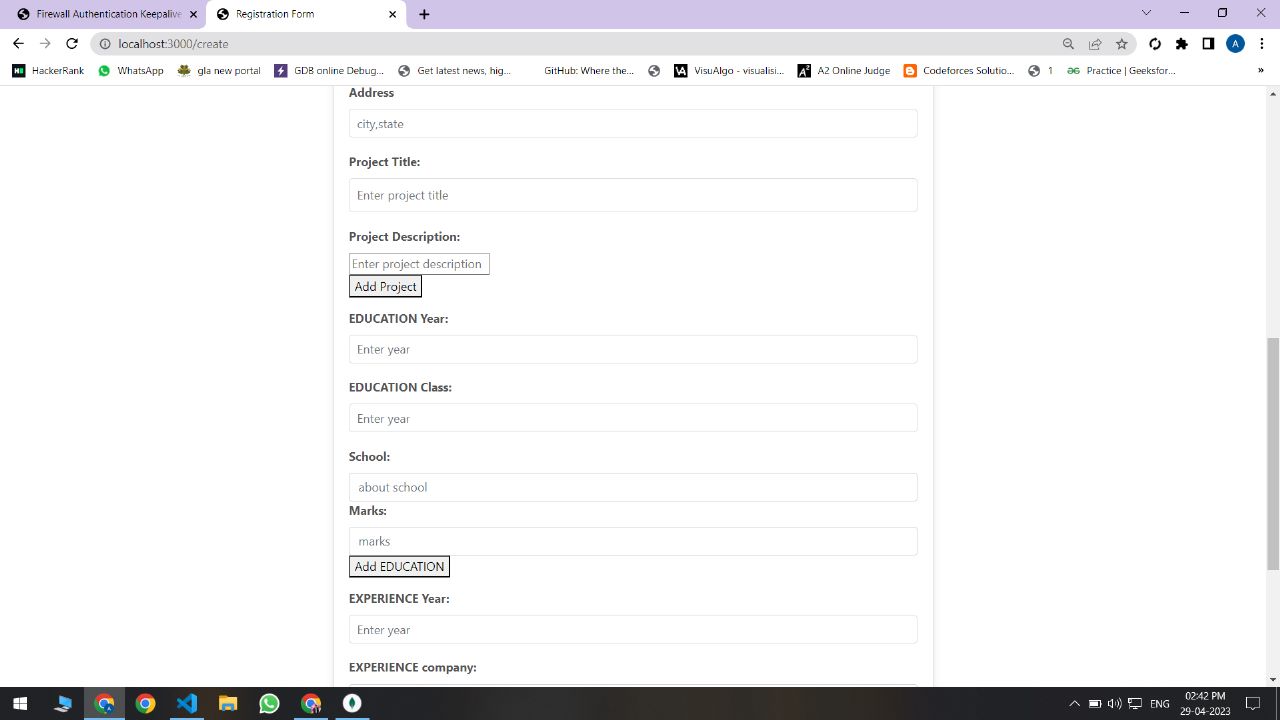
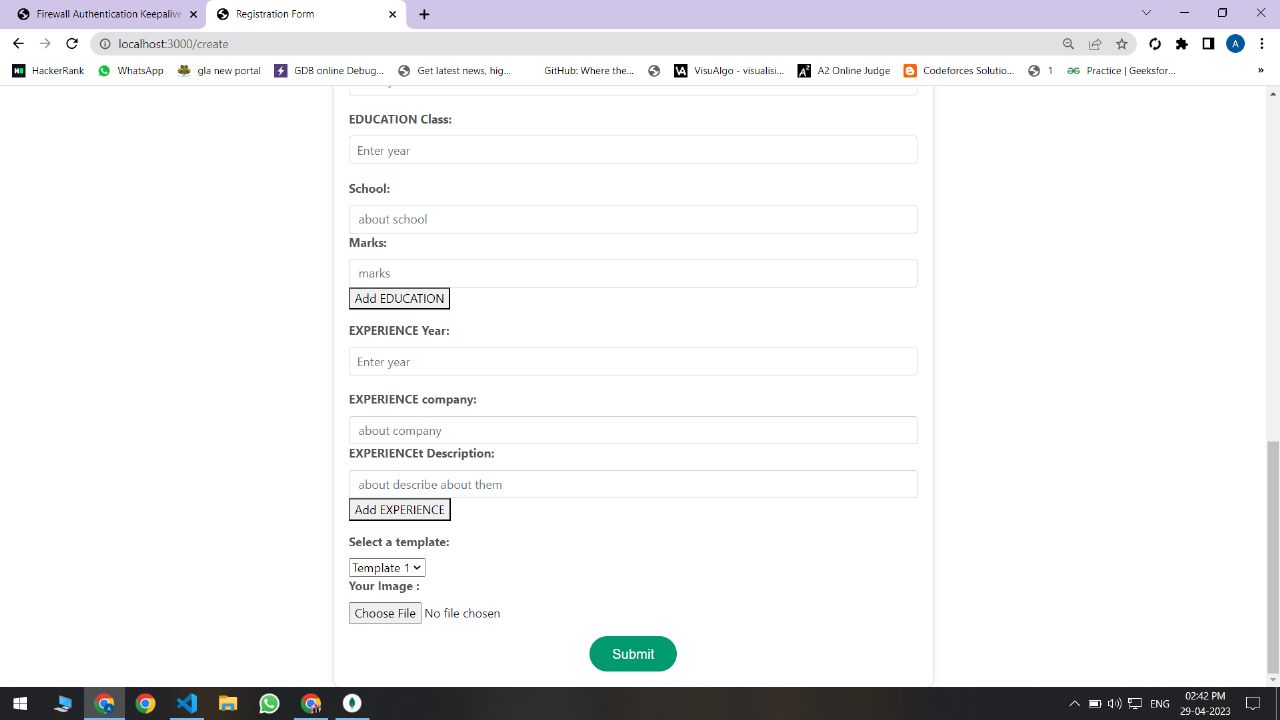
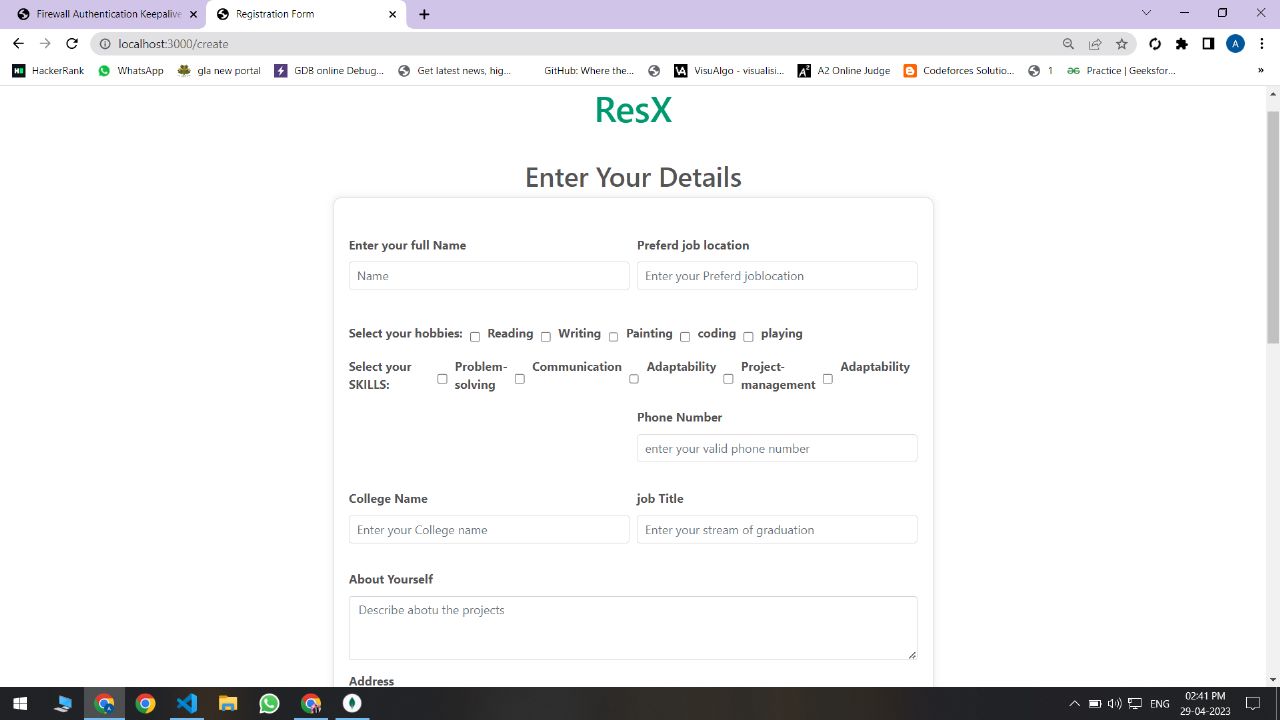
}

</script>

</body>

</html>

**Registration Page**



**Code for Registration Page**

<!DOCTYPE html>

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

<head>

<meta charset="utf-8">

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/css/all.min.css"/>

<style>

@import url('https://fonts.googleapis.com/css?family=Poppins:400,500,600,700&display=swap');

\*{

margin: 0;

padding: 0;

/\* user-select: none; \*/

box-sizing: border-box;

font-family: 'Poppins', sans-serif;

}

html,body{

height: 100%;

}

body{

display: grid;

place-items: center;

background: #dde1e7;

text-align: center;

background-image: url(./photo/3d-abstract-background-with-paper-cut-shape\_1217-3284.webp);

}

.content{

width: 500px;

padding: 40px 30px;

background: #dde1e7;

border-radius: 10px;

box-shadow: -3px -3px 7px #ffffff73,

2px 2px 5px rgba(94,104,121,0.288);

}

.content .text{

font-size: 33px;

font-weight: 600;

margin-bottom: 35px;

color: #595959;

}

.field{

height: 50px;

width: 100%;

display: flex;

position: relative;

}

.field:nth-child(2){

margin-top: 20px;

}

.field input{

height: 100%;

width: 100%;

padding-left: 45px;

outline: none;

border: none;

font-size: 18px;

background: #dde1e7;

color: #595959;

border-radius: 25px;

box-shadow: inset 2px 2px 5px #BABECC,

inset -5px -5px 10px #ffffff73;

}

.field input:focus{

box-shadow: inset 1px 1px 2px #BABECC,

inset -1px -1px 2px #ffffff73;

}

.field span{

position: absolute;

color: #595959;

width: 50px;

line-height: 50px;

}

.field label{

position: absolute;

top: 50%;

transform: translateY(-50%);

left: 45px;

pointer-events: none;

color: #666666;

}

.field input:valid ~ label{

opacity: 0;

}

.forgot-pass{

text-align: left;

margin: 10px 0 10px 5px;

}

.forgot-pass a{

font-size: 16px;

color: #3498db;

text-decoration: none;

}

.forgot-pass:hover a{

text-decoration: underline;

}

button{

margin: 15px 0;

width: 100%;

height: 50px;

font-size: 18px;

line-height: 50px;

font-weight: 600;

background: #dde1e7;

border-radius: 25px;

border: none;

outline: none;

cursor: pointer;

color: #595959;

box-shadow: 2px 2px 5px #BABECC,

-5px -5px 10px #ffffff73;

}

button:focus{

color: #3498db;

box-shadow: inset 2px 2px 5px #BABECC,

inset -5px -5px 10px #ffffff73;

}

.sign-up{

margin: 10px 0;

color: #595959;

font-size: 16px;

}

.sign-up a{

color: #3498db;

text-decoration: none;

}

.sign-up a:hover{

text-decoration: underline;

}

</style>

</head>

<body>

<div class="content">

<div class="text">

Sign up now

</div>

<form action="/register" method="POST">

<div class="field">

<input type="text" required name="name">

<span class="fas fa-user"></span>

<label>Enter Username</label>

</div>

<br>

<div class="field">

<input type="text" required name="email">

<span class="fas fa-user"></span>

<label>Enter Email</label>

</div>

<br>

<div class="field">

<input type="number" required name="age">

<span class="fas fa-lock"></span>

<label>age</label>

</div>

<br>

<div class="field">

<input type="password" required name="password">

<span class="fas fa-lock"></span>

<label>Enter Password</label>

</div>

<br>

<div class="field">

<input type="password" required name="cpassword">

<span class="fas fa-lock"></span>

<label>Confirm Password</label>

</div>

<br>

<div class="field">

<input type="radio" name="gender" value="male">

<span>Male</span><br>

</div>

<div class="field">

<input type="radio" name="gender" value="female">

<span>female</span>

</div>

<br>

<div class="field">

<input type="text" required name="profession">

<span class="fas fa-lock"></span>

<label>Enter your profession</label>

</div>

<button type="submit" name="submit">Create Account</button>

<div class="sign-up">

Already have an account?

<a href="/">login now</a>

</div>

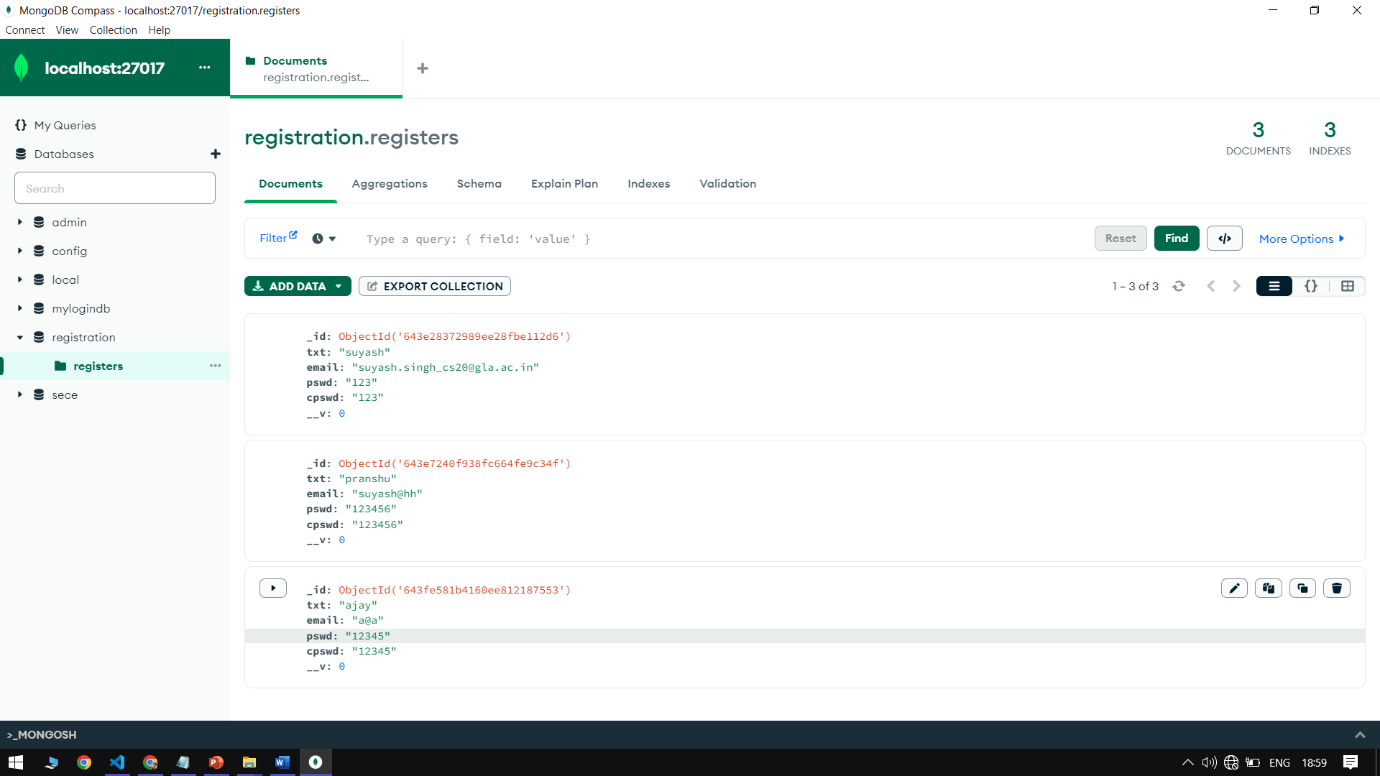
</form>

</div>

</body>

</html>

**Database**



**Connection file**

const mongoose = require("mongoose");

//the connect method gives us the data in the future and store also ..

mongoose.connect("mongodb://127.0.0.1:27017/employe",{

useNewUrlParser:true,

useUnifiedTopology:true,

// useCreateIndex:true

}).then(()=>{

console.log(`connection sussefully done`);

}).catch((e)=>{

console.log(e);

})

**App.js file**

const express= require("express");

const path=require("path");

//in this app we have all the power of express..

const app=express();

const hbs=require("hbs");

require("./db/conn");

const Register=require("./models/registers");

const { error } = require("console");

const port= process.env.PORT || 3000;

const static\_path =path.join(\_\_dirname,"../public");

const template\_path =path.join(\_\_dirname,"../templates/views");

const partials\_path =path.join(\_\_dirname,"../templates/partials");

app.use(express.json());

//mean of this line everything is written inside the form is want to get in from encoding

app.use(express.urlencoded({extended:false}));

console.log(path.join(\_\_dirname));

app.use(express.static(static\_path));

app.set("view engine","hbs");

app.set("views",template\_path);

hbs.registerPartials(partials\_path);

//Basically it is intended for binding the middleware to your application.

app.get("/",(req,res)=>

res.render("index");

});

app.get("/register",(req,res)=>{

res.render("register");

});

app.get("/index",(req,res)=>{

res.render("index");

});

//just demo we write here

// app.post("/register", async(req,res)=>{

// try{

// console.log(req.body.name);

// console.log(req.body.email);

// console.log(req.body.password);

// console.log(req.body.cpassword);

// console.log(req.body.gender);

// console.log(req.body.profession);

// res.send(req.body.name);

// }

// catch{

// res.status(400).send(error);

// }

// })

//this for get the value of index one so we write this prorgam

app.post("/index",async(req,res)=>{

try{

const email = req.body.email;

const password = req.body.password;

// console.log(`${email}and password is${password}`);

const useremail=await Register.findOne({email:email});

//if we want to know the what is password then

//res.send(useremail.password)

if(useremail.password===password)

{

res.status(201).render("index");

}

else

{

res.send("apki dale hui information shi nhi hai kripa shi kr le");

}

}

catch(errore){

res.status(400).send("invalid images");

}

})

// //create a new user in our database

app.post("/register",async(req,res)=>{

try{

const password=req.body.password;

const cpassword=req.body.cpassword;

if(password===cpassword){

const registeremployee=new Register({

name:req.body.name,

email:req.body.email,

age:req.body.age,

password:req.body.password,

cpassword:req.body.cpassword,

gender:req.body.gender,

profession:req.body.profession

})

//we get all data from the database but we want to store all the data then we will use

const registerd= await registeremployee.save();

res.status(201).render("index");

}

else

{

res.send("password are not matching")

}

}

catch(error) {

res.status(400).send(error);

}

})

app.listen(port,()=>{

console.log(`server is runing on port ${port}`);})

**Script.js file**

const sign\_in\_btn = document.querySelector("#sign-in-btn");

const sign\_up\_btn = document.querySelector("#sign-up-btn");

const container = document.querySelector(".container");

sign\_up\_btn.addEventListener("click", () => {

container.classList.add("sign-up-mode");

});

sign\_in\_btn.addEventListener("click", () => {

container.classList.remove("sign-up-mode");

});

$(document).ready(function() {

$("#change").click(function() {

if ($(this).text() === "Login / Sign Up") {

$(this).text("Logout");

} else {

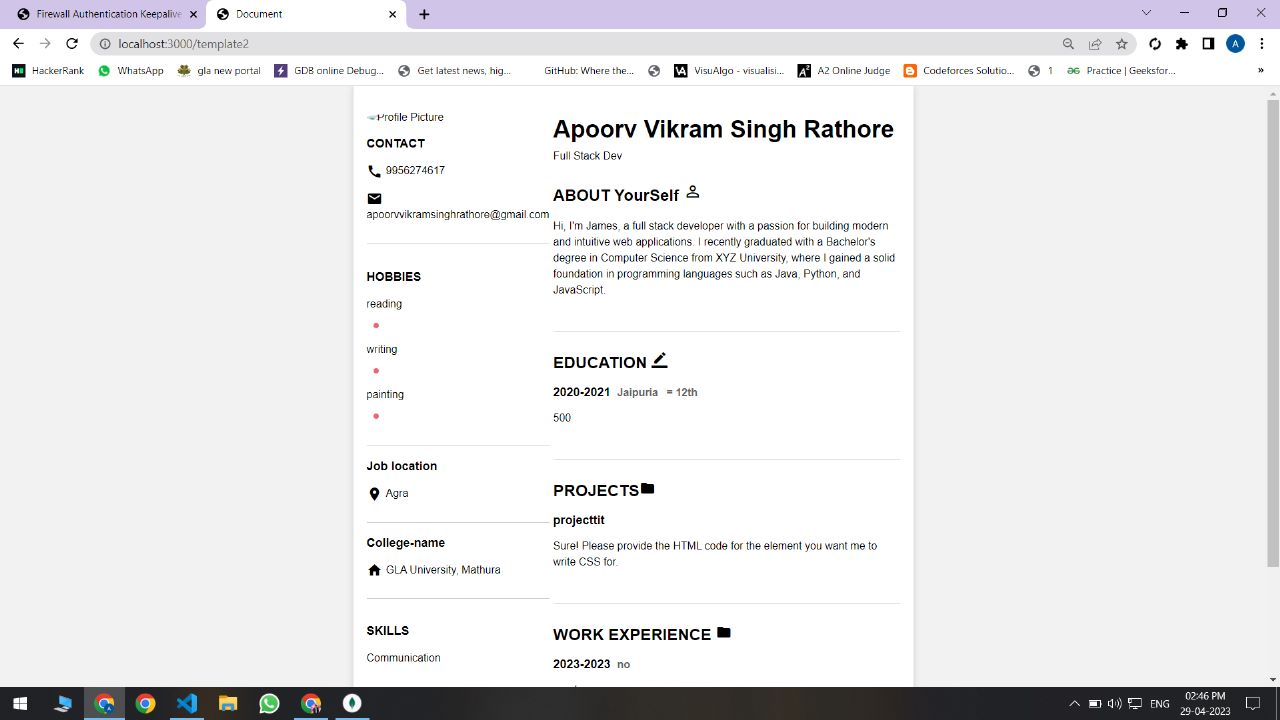
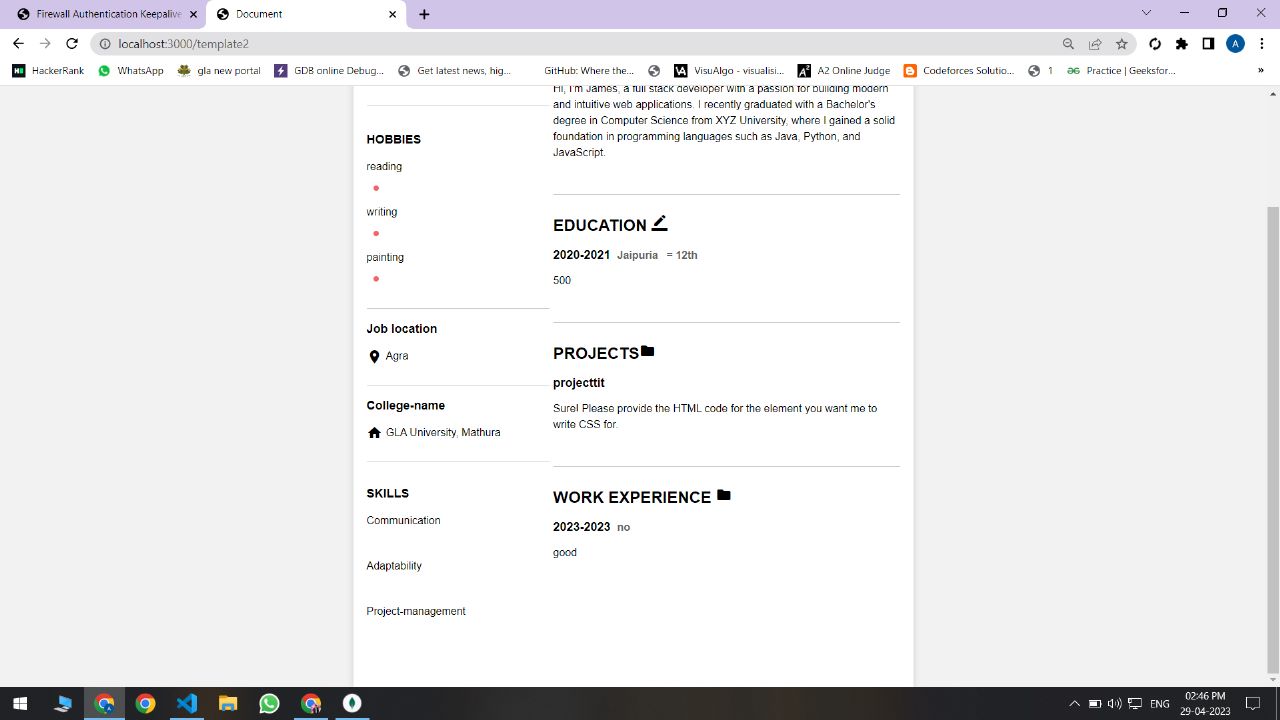
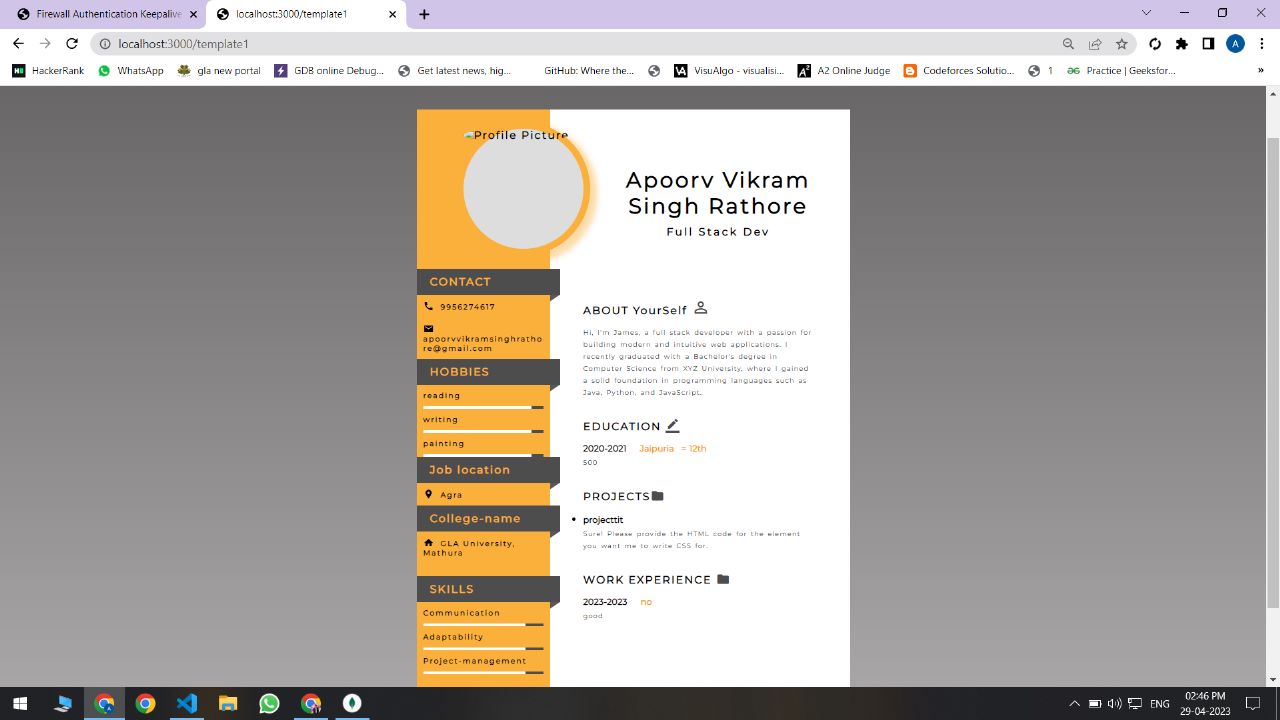
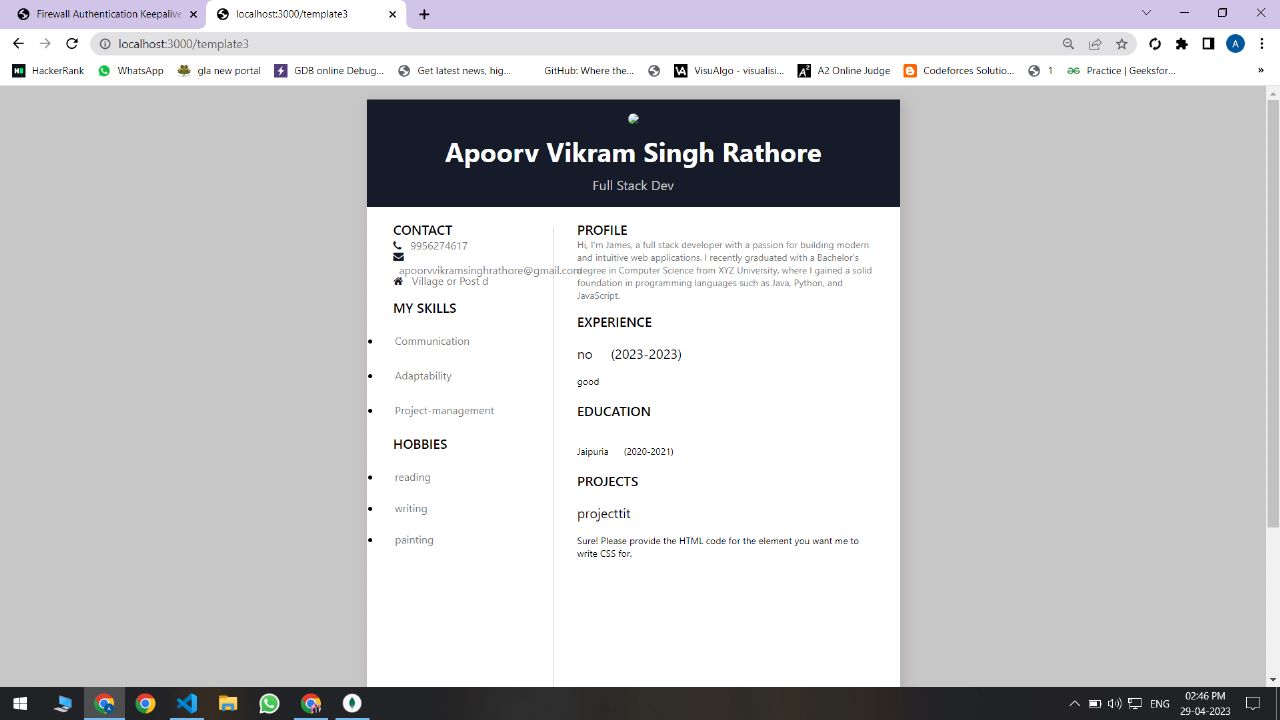
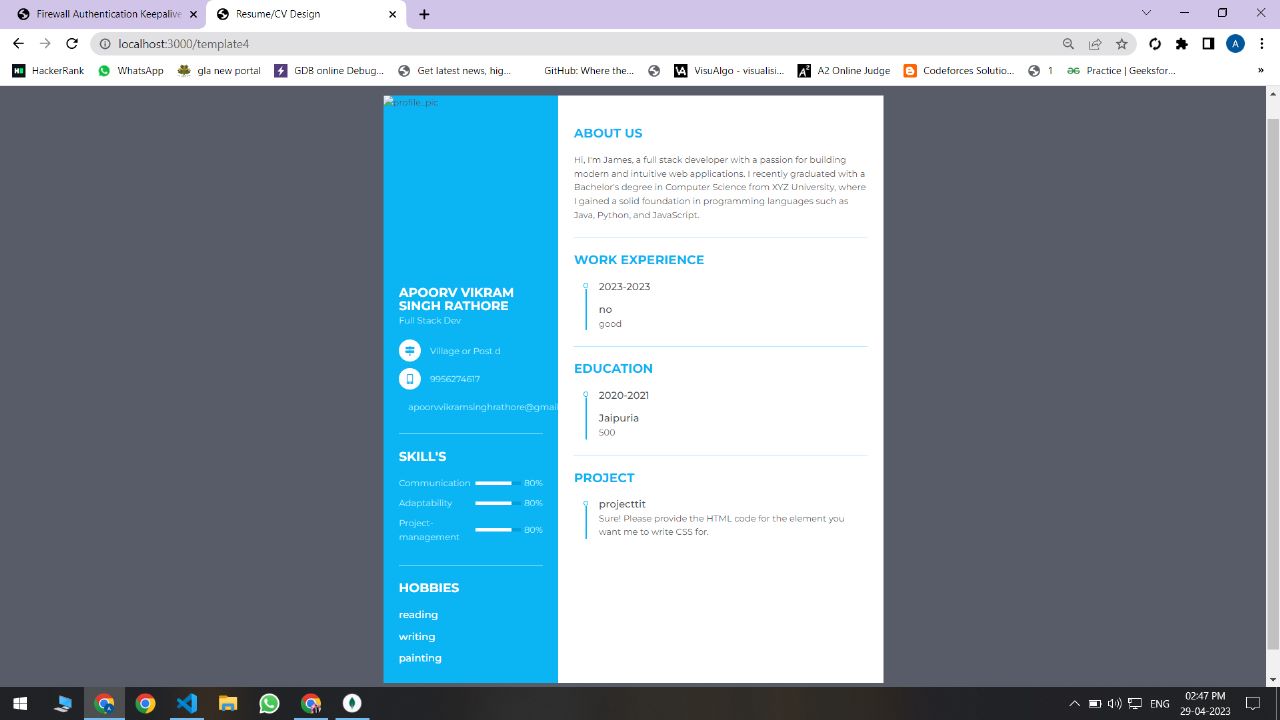
$(this).text("Login / Sign Up");

}

});

});

**Templates**



**Code For templates**

**Template1:**

<!DOCTYPE html>

<html>

<head>

<title></title>

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

<style>

@import url('https://fonts.googleapis.com/css2?family=Montserrat&display=swap');

body{

background-image: linear-gradient(#636162, #b1afaf);

height: 122vh;

font-weight: bold;

letter-spacing: 2px;

}

\*{

margin: 0px; padding: 0px; font-family: 'Montserrat', sans-serif;

}

.main{

background-color: white;

height: auto;

width: 650px;

position: absolute;

transform: translate(-50%, -50%);

left: 50%;

top: 60%;

overflow-wrap: break-word;

}

.left{

background-color: #fbb03b;

width: 200px;

height: 100%;

float: left;

}

.right{

width: 450px;

float: left;

}

.profile-img{

height: 180px;

width: 180px;

background-color: #ddd;

border-radius: 50%;

margin-bottom: 20px;

margin-left: 60px;

border: 10px solid #fbb03b;

box-shadow: 6px 7px 9px 5px #fbb03b73;

}

.profile-img img{

width: 100%;

border-radius: 50%;

}

.box-1{

position: relative;

}

.heading{

background-color: #4d4d4d;

padding: 10px;

color: #fbb03b;

text-align: left;

padding-left: 20px;

margin-right: -15px;

}

.heading::after{

content: '';

position: absolute;

width: 21px;

height: 18px;

background-color: #5a5959;

left: 191px;

top: 26px;

transform:rotate(56deg);

z-index: -1;

}

.p1{

padding: 9px 10px;

font-size: 11px;

}

.icons1{

font-size: 16px!important;

padding-right: 10px!important;

vertical-align: sub;

}

.skill-container{

width: 90%;

background-color: #4d4d4d;

margin: 0px 10px;

}

.skill{

text-align: right;

padding-top: 2px;

padding-bottom: 2px;

color: white;

}

.html{

width: 90%;

background-color: white;

}

.css{

width: 80%;

background-color: white;

}

.js{

width: 85%;

background-color: white;

}

.jquery{

width: 80%;

background-color: white;

}

.web {

width: 85%;

background-color: white;

}

.gra{

width: 70%;

background-color: white;

}

.icons2{

padding: 18px 10px;

}

.name-div{

padding: 86px 0px 60px 55px;

text-align: center;

letter-spacing: 3px;

}

.name-div h1{

margin-bottom: 10px;

}

.box-2{

padding: 0px 50px;

margin-top: 30px;

}

.p2{

font-size: 10px;

font-weight: 300;

letter-spacing: 1px;

word-spacing: 2px;

line-height: 18px;

margin-top: 5px;

}

h2{

font-size: 16px;

margin-bottom: 15px;

}

.icons3{

vertical-align: bottom;

font-size: 21px!important;

color: #4d4d4d;

}

.p3{

font-size: 13px;

word-spacing: 1px;

letter-spacing: 0px;

margin-top: 10px;

}

.p3 span{

color: #ff9b00; padding-left: 20px;

}

</style>

<link href="https://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet">

</head>

<body>

<div class="main">

<div class="left">

<br>

<div class="profile-img">

<img src="/uploads/{{imagec.name}}" alt="Profile Picture" />

</div>

{{!-- <div class="profile-img"><img src="/images/{{data.imagec.contentType}}.jpg"></div> --}}

<div class="box-1">

<div class="heading">

<p>CONTACT</p>

</div>

<p class="p1"><i class="material-icons icons1">call</i>{{data.phonec}}</p>

<p class="p1"><i class="material-icons icons1">email</i>{{data.emailc}}</p>

</div>

<div class="box-1">

<div class="heading">

<p>HOBBIES</p>

</div>

{{#each data.hobby1}}

<p class="p1">{{this}}

<div class="skill-container">

<div class="skill html"></div>

</div>

</p>

{{/each}}

</div>

<div class="box-1">

<div class="heading">

<p>Job location</p>

</div>

<p class="p1"><i class="material-icons icons1">location\_on</i>{{data.locationc}}

</p>

</div>

<div class="box-1">

<div class="heading">

<p>College-name</p>

</div>

<p class="p1"><i class="material-icons icons1">home</i>{{data.collegec}}

</p>

</div>

<br>

<div class="box-1">

<div class="heading">

<p>SKILLS</p>

</div>

{{#each data.skill1}}

<p class="p1">{{this}}

<div class="skill-container">

<div class="skill web"></div>

</div>

</p>

{{/each}}

</div>

<br>

</div>

<div class="right">

<div class="name-div">

<h1>{{ data.namec}}</h1>

<p>{{data.streamc}}</p>

</div>

<div class="box-2">

<h2>ABOUT YourSelf <i class="material-icons icons3" style="font-size: 28px!important; ">perm\_identity</i></h2>

<p class="p2">

{{data.about}}

</p>

</div>

<div class="box-2">

<h2>EDUCATION <i class="material-icons icons3" >border\_color</i></h2>

{{#each data.education as |education index|}}

{{#each education.year1 as |kamm kam|}}

<p class="p3">{{lookup education.year1 kam}}<span>{{lookup education.school kam}}&ensp; = {{lookup education.class kam}}</span></p>

<p class="p2">

{{lookup education.marks kam}}

</p>

{{/each}}

{{/each}}

</div>

<div class="box-2">

<h2>PROJECTS<i class="material-icons icons3">folder</i></h2>

{{#each data.project as |project index|}}

{{#each project.projecttitle1 as |kamm kam|}}

<ul>

<li>

<p class="p3">{{lookup project.projecttitle1 kam}}</p>

<p class="p2">{{lookup project.projectdescription1 kam}}</p>

</li>

</ul>

{{/each}}

{{/each}}

</div>

<div class="box-2">

<h2>WORK EXPERIENCE <i class="material-icons icons3" >folder</i></h2>

{{#each data.experience as |experience index|}}

{{#each experience.year as |kamm kam|}}

<p class="p3">{{lookup experience.year kam}}<span>{{lookup experience.company kam}}</span></p>

<p class="p2">

{{lookup experience.desc kam}}

</p>

{{/each}}

{{/each}}

</div>

</div>

</div>

</body>

</html>

**Template2:**

<!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>

<style>

/\* body styles \*/

body {

font-family: Arial, Helvetica, sans-serif;

background-color: #f2f2f2;

margin: 0;

padding: 0;

}

/\* container styles \*/

.main {

max-width: 800px;

margin: 0 auto;

display: flex;

flex-wrap: wrap;

justify-content: space-between;

background-color: #fff;

box-shadow: 0px 0px 10px #ccc;

padding: 20px;

}

/\* left side styles \*/

.left {

flex-basis: 30%;

}

.profile-img img {

display: block;

width: 100%;

height: auto;

border-radius: 50%;

}

.box-1 {

margin-top: 20px;

border-bottom: 1px solid #ccc;

padding-bottom: 20px;

}

.box-1:last-of-type {

border: none;

}

.heading {

font-size: 18px;

font-weight: bold;

margin-bottom: 10px;

}

.p1 {

font-size: 16px;

line-height: 1.5;

margin-bottom: 10px;

}

.icons1 {

font-size: 20px;

margin-right: 5px;

vertical-align: middle;

}

.skill-container {

margin-left: 10px;

display: inline-block;

}

.skill {

display: inline-block;

height: 8px;

width: 8px;

border-radius: 50%;

margin-right: 2px;

}

.html {

background-color: #f1646c;

}

.box-1:nth-child(2) {

margin-top: 40px;

}

.box-1:nth-child(4) {

margin-top: 40px;

}

/\* right side styles \*/

.right {

flex-basis: 65%;

}

.name-div {

margin-top: 0;

}

h1 {

font-size: 36px;

margin-bottom: 0;

}

.name-div p {

font-size: 16px;

margin-top: 10px;

}

.box-2 {

margin-top: 30px;

border-bottom: 1px solid #ccc;

padding-bottom: 30px;

}

.box-2:last-of-type {

border: none;

}

.icons3 {

font-size: 20px;

margin-right: 5px;

}

h2 {

font-size: 24px;

font-weight: bold;

margin-bottom: 20px;

}

.p2 {

font-size: 16px;

line-height: 1.5;

margin-bottom: 20px;

}

.p3 {

font-size: 18px;

font-weight: bold;

margin-bottom: 5px;

}

.p3 span {

font-size: 16px;

margin-left: 10px;

color: #666;

}

ul {

list-style: none;

margin: 0;

padding: 0;

}

ul li {

margin-bottom: 10px;

}

@media screen and (max-width: 768px) {

.main {

max-width: 100%;

flex-direction: column;

box-shadow: none;

padding: 10px;

}

.left {

flex-basis: 100%;

margin-bottom: 20px;

}

.right {

flex-basis: 100%;

}

.box-1,

.box-2 {

margin: 0;

border: none;

padding: 0;

}

.profile-img img {

margin-bottom: 20px;

}

h1 {

font-size: 28px;

}

h2 {

font-size: 20px;

}

}

/\* additional styles for print \*/

@media print {

body {

background-color: #fff;

}

.main {

box-shadow: none;

}

.left {

margin-bottom: 0;

}

.name-div {

page-break-inside: avoid;

}

}

</style>

<link href="https://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet">

</head>

<body>

<div class="main">

<div class="left">

<br>

<div class="profile-img">

<img src="../../uploads/{{data.\_id}}" alt="Profile Picture" />

</div>

{{!-- <div class="profile-img"><img src="/images/{{data.imagec.contentType}}.jpg"></div> --}}

<div class="box-1">

<div class="heading">

<p>CONTACT</p>

</div>

<p class="p1"><i class="material-icons icons1">call</i>{{data.phonec}}</p>

<p class="p1"><i class="material-icons icons1">email</i>{{data.emailc}}</p>

</div>

<div class="box-1">

<div class="heading">

<p>HOBBIES</p>

</div>

{{#each data.hobby1}}

<p class="p1">{{this}}

<div class="skill-container">

<div class="skill html"></div>

</div>

</p>

{{/each}}

</div>

<div class="box-1">

<div class="heading">

<p>Job location</p>

</div>

<p class="p1"><i class="material-icons icons1">location\_on</i>{{data.locationc}}

</p>

</div>

<div class="box-1">

<div class="heading">

<p>College-name</p>

</div>

<p class="p1"><i class="material-icons icons1">home</i>{{data.collegec}}

</p>

</div>

<br>

<div class="box-1">

<div class="heading">

<p>SKILLS</p>

</div>

{{#each data.skill1}}

<p class="p1">{{this}}

<div class="skill-container">

<div class="skill web"></div>

</div>

</p>

{{/each}}

</div>

<br>

</div>

<div class="right">

<div class="name-div">

<h1>{{ data.namec}}</h1>

<p>{{data.streamc}}</p>

</div>

<div class="box-2">

<h2>ABOUT YourSelf <i class="material-icons icons3" style="font-size: 28px!important; ">perm\_identity</i></h2>

<p class="p2">

{{data.about}}

</p>

</div>

<div class="box-2">

<h2>EDUCATION <i class="material-icons icons3" >border\_color</i></h2>

{{#each data.education as |education index|}}

{{#each education.year1 as |kamm kam|}}

<p class="p3">{{lookup education.year1 kam}}<span>{{lookup education.school kam}}&ensp; = {{lookup education.class kam}}</span></p>

<p class="p2">

{{lookup education.marks kam}}

</p>

{{/each}}

{{/each}}

</div>

<div class="box-2">

<h2>PROJECTS<i class="material-icons icons3">folder</i></h2>

{{#each data.project as |project index|}}

{{#each project.projecttitle1 as |kamm kam|}}

<ul>

<li>

<p class="p3">{{lookup project.projecttitle1 kam}}</p>

<p class="p2">{{lookup project.projectdescription1 kam}}</p>

</li>

</ul>

{{/each}}

{{/each}}

</div>

<div class="box-2">

<h2>WORK EXPERIENCE <i class="material-icons icons3" >folder</i></h2>

{{#each data.experience as |experience index|}}

{{#each experience.year as |kamm kam|}}

<p class="p3">{{lookup experience.year kam}}<span>{{lookup experience.company kam}}</span></p>

<p class="p2">

{{lookup experience.desc kam}}

</p>

{{/each}}

{{/each}}

</div>

</div>

</div>

</body>

</html>

**CHAPTER 4**

**4.1 Conclusion**

We have completed our project within time limit with the coordination of our team members under the supervision of our mentor Mr. Mandeep Singh.

**4.2 Future Work**

As the job market continues to evolve, resume builder websites are likely to remain an essential tool for job seekers. Here are some potential future uses of resume builder websites:

1. Personalized Resume Creation: With advances in artificial intelligence and machine learning, resume builder websites can provide even more personalized resume creation. They could analyze a job seeker's skills, experience, and goals, and create a tailored resume that highlights their strengths and matches the requirements of the job.

2. Easy Resume Updating: As job seekers gain new skills and experiences, they need to update their resumes accordingly. A resume builder website could make this process even easier by allowing users to import their latest information from other sources, such as social media platforms or online portfolios.

3. Increased Accessibility: As remote work becomes more prevalent, job seekers may need to apply for jobs across different countries and time zones. A resume builder website could increase accessibility by supporting multiple languages and providing resources on how to create resumes that meet the requirements of specific countries.

4. Integration with Hiring Platforms: As more companies move their recruitment processes online, resume builder websites could integrate with hiring platforms to make it easier for job seekers to apply for jobs. For example, they could allow users to submit their resumes directly to job postings on LinkedIn or other job search platforms.

5. Career Development Resources: A resume builder website could expand its offerings to include career development resources, such as networking tips, interview preparation guides, and career coaching services. This would provide job seekers with a comprehensive suite of tools and resources to help them succeed in their career goals.

6. Enhanced User Experience: Finally, resume builder websites could continue to improve the user experience by providing more user-friendly interfaces, intuitive navigation, and simplified processes for creating and editing resumes. This would make it easier and faster for users to create effective resumes and ultimately land their dream jobs.

Overall, the future use of resume builder websites is likely to involve even more personalized and streamlined resume creation, increased accessibility and integration with hiring platforms, and expanded resources for career development.

**REFERENCES**

* FCC: https://[www.freecodecamp.org/](http://www.freecodecamp.org/)

* MDN: https://developer.mozilla.org/en-US/

* Mongo DB: https://[www.mongodb.com/docs/](http://www.mongodb.com/docs/)

* [www.google.com](http://www.google.com/)

* W3 School: https://[www.w3schools.com/](http://www.w3schools.com/)