**VISVESVARAYA TECHNOLOGICAL UNIVERSITY**

**“JNANA SANGAMA”, BELAGAVI, KARNATAKA-590018**



**“PORTFOLIO WEBSITE”**

## A Report Submitted By

**Ashwini C N 4MN22CS008**

**Bhanu Prakash R 4MN22CS009**

**Chandana M 4MN22CS013**

Submitted in partial fulfillment of the requirement for the award of the degree of

**Bachelor of Engineering in Computer Science and Engineering**

**Under the guidance of**

## Prof Sumaiya

Assistant Professor

Department of Computer Science & Engineering MIT Thandavapura



## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

**MAHARAJA INSTITUTE OF TECHNOLOGY THANDAVAPURA**

Just Off NH 766, Nanjanagudu Taluk, Mysore District – 571302

(Approved by AICTE, Accredited by NBA, New Delhi and Affiliated by VTU, Belagavi)

## 2024-2025

**MAHARAJA INSTITUTE OF TECHNOLOGY THANDAVAPURA**

Just off NH 766, Nanjanagudu Taluk, Mysore District – 571302

(Approved by AICTE, Accredited by NBA, New Delhi and Affiliated by VTU, Belagavi)

## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



**CERTIFICATE**

Certified that the project work entitled “**Portfolio Website**” carried out by **Ashwini C N**(4MN22CS008), **Bhanu Prakash R**(4MN22CS009), **Chandana M**(4MN22CS013) a bonafide students of **Maharaja Institute of Technology Thandavapura** in partial fulfilment for the award of Bachelor of Engineering in **Computer Science & Engineering** of the Visvesvaraya Technological University, Belagavi during the year 2024-25. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the Report deposited in the departmental library.

The project report has been approved as it satisfies the academic requirements in respect of Project work prescribed for the said Degree.

|  |  |  |
| --- | --- | --- |
| Signature of guide |  | Signature of the HOD |
| **Prof Sumaiya** |  | **Dr. Ranjit K N** |
| Assistant Professor |  | Associate Professor |
| Dept. of CS&E, |  | HoD of CS&E |
| MIT Thandavapura |  | MIT Thandavapura |

## Declaration

This is to declare that this report has been written by us. No part of the report is plagiarized from other sources. All information included from other sources have been duly acknowledged. We aver that if any part of the report is found to be plagiarized, we will be solely responsible for it.

Ashwini C N

4MN22CS008

Bhanu Prakash R

4MN22CS009

Chandana M

4MN22CS013

Place: Mysuru Date:27-10-24

## ACKNOWLEDGEMENT

Wholeheartedly, I thank our MET Management for providing the necessary environment, infrastructure and encouragement for carrying out our project work at Maharaja Institute of Technology Thandavapura, Mysuru.

We would like to express our sincere thanks to our beloved Principal Dr. Y.T. Krishne Gowda for giving us moral support and continuous encouragement which has been the key for the successful completion of the mini project.

We are pleased to acknowledge Dr. Ranjit K N, HoD, Department of Computer Science and Engineering, for her encouragement and support throughout the project.

We would like to express our heart-felt gratitude to our Project Co-ordinator Prof Sumaiya Assistant Professor, Department of Computer Science and Engineering for his valuable suggestions and excellent guidance rendered throughout this project.

Further, we extend our thanks to all the faculty members and technical staff of our department for their suggestions, support and providing resources at needed times.

Finally, we express our sincere gratitude to our parents and friends who have been the embodiment of love and affection which helped us to carry out the project in a smooth and successful way.

Ashwini C N [4MN22CS008]

Bhanu Prakash R [4MN22CS009]

Chandana M [4MN22CS013]

# ABSTRACT

This project presents a web application designed to showcase an individual's resume and biodata through a multi-page website. Built using HTML and CSS, the application features a clean, user-friendly interface that allows visitors to easily navigate between pages, including the home page, resume page, biodata page, and contact page. Each section is meticulously crafted to present information in a structured and visually appealing manner, enhancing user experience and accessibility.The resume page highlights key qualifications, work experience, and skills, while the biodata page offers personal details, educational background, and other relevant information. A contact form facilitates user inquiries, enabling seamless communication. The application is fully responsive, ensuring optimal viewing on various devices and screen sizes.This project not only serves as a personal portfolio but also exemplifies best practices in web development, including semantic HTML, CSS styling, and responsive design principles. The end result is a professional online presence that effectively showcases the individual's achievements and facilitates engagement with potential employers or collaborators.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SL.No.** | | | **Index** | **Page No.** |
| **1** |  |  | **INTRODUCTION** | **1-4** |
|  | 1.1 |  | Problem statement | 1 |
|  | 1.2 |  | Objective | 1 |
|  | 1.3 |  | Scope | 2 |
|  | 1.4 |  | Web Application | 3 |
| **2** |  |  | **SYSTEM REQUIREMENT AND SPECIFICATION** | **5-13** |
|  | 2.1 |  | Functional Requirements | 5 |
|  | 2.2 |  | HTML Tags | 5 |
|  | 2.3 |  | CSS | 8 |
|  | 2.4 |  | Cloud to be Hosted | 10 |
|  | 2.5 |  | Non Functional requiremnets | 12 |
| **3** |  |  | **SYSTEM ANALYSIS AND DESIGN** | **14** |
|  | 3.1 |  | Use Case Diagram | 14 |
| **4** |  |  | **IMPLEMENTATION** | **15-22** |
|  | 4.1 |  | Algorithm | 15 |
|  | 4.2 |  | Flowchart | 15 |
|  | 4.3 |  | Source Code | 16 |
|  | 4.4 |  | Snapshots | 19 |
|  |  |  | **CONCLUSION** | 23 |

|  |  |  |
| --- | --- | --- |
| **Figure No.** | **Index** | **Page No.** |
| Fig:4.4.1 | Home Page | 19 |
| Fig:4.4.2 | Portfolio Page | 20 |
| Fig:4.4.3 | Resume Page | 20 |
| Fig:4.4.4 | Biodata Page | 22 |

**Chapter 1**

**INTRODUCTION**

## 1.1 Problem statement

## “Construct a Website (multiple Web pages) containing ‘Resume’ and Bio -data by using relevant HTML elements and appropriate styling for presentation with CSS/jQuery/JavaScript and Host the Website on a cloud platform.”

## 1.2 Objective

## Steps to Achieve the Objective

## Plan the Structure

## Decide on the main pages:

## Home

## Resume

## Bio

## Contact

## Design the Layout

## Sketch a layout for each page, including headers, footers, and navigation.

## Create HTML Pages

## Home Page (index.html):

## Introduction and links to other pages.

## Resume Page (resume.html):

## Sections for education, skills, and certifications.

## Page (bio.html):

## Personal background, interests, and contacts.

## Style with CSS

## Use CSS for styling:

## Set a color scheme and fonts.

## Use Flexbox or Grid for layout.

## Style buttons, links, and forms for a cohesive look.

## Choose a Cloud Hosting Platform

## Options include:

## GitHub Pages (free)

## Netlify (free tier)

## Vercel (free tier)

## Technologies Used

## HTML: Structure and content of the website.

## CSS: Presentation and styling.

## Conclusion

## By following these steps, you will create a polished and professional website that effectively presents your resume and bio-data. This project will enhance your web development skills and serve as a valuable online presence.

## 1.3 Scope of the Project

## Project Overview

* Develop a multi-page personal website to showcase your resume and bio-data.
* Utilize HTML, CSS, and JavaScript/jQuery for a dynamic and responsive user experience

## Functional Requirements

## Home Page

## Introduction with a brief overview.

## Navigation links to other sections (Resume, Bio, Contact).

## Resume Page

## Display of professional experience, education, skills, and certifications.

## Use of sections and lists for clear presentation.

## Bio Page

## Personal background information, interests, and achievements.

## Optional: A photo and a brief personal story.

## Non-Functional Requirements

## Responsive Design

## Ensure the website functions well on mobile, tablet, and desktop devices.

## Performance

## Optimize images and scripts for fast loading times.

## Accessibility

## Ensure the website meets basic accessibility standards (e.g., proper alt text for images).

## Design Specifications

## Visual Style

## Choose a cohesive color scheme and typography.

## Consistent button and link styles.

## Layout

## Use Flexbox or Grid for layout structuring.

## Clear hierarchy of content with headings and spacing.

## Technology Stack

 **HTML**: Structure and content.

##  **CSS**: Styling and layout.

## Deliverables

 Fully functional multi-page website.

 Source code hosted on a version control platform (e.g., GitHub).

##  Deployed website accessible via a public URL.

## 1.4 Web Application

## 1. Technology Stack:

## Front-end: HTML, CSS, JavaScript (possibly with a framework like React or Vue.js)

## Back-end: Node.js with Express, or Python with Flask/Django

## Database: MongoDB, MySQL, or any other database of your choice

## Deployment: Platforms like Heroku, Vercel, or GitHub Pages

## 2. Features:

## User Features:

## View resume and biodata

## Download resume in PDF format

## Contact form to send messages

## Admin Features:

## Update resume and biodata

## Manage website content and design

## Structure of the Web Application

## 1. Front-end Development:

## HTML Pages:

## index.html (Home page with an overview)

## resume.html (Displays resume)

## biodata.html (Displays personal information)

## contact.html (Contact form)

## CSS Styles:

## Stylesheets for layout and design

## Responsive design to ensure usability on different devices

## JavaScript:

## Functionality for form validation

## AJAX requests to handle data submission without refreshing the page

## 2. Back-end Development:

## Server Setup:

## Create an Express (or Flask/Django) server to handle requests

## Define routes for:

## Viewing resume and biodata

## Handling contact form submissions

## Admin functionalities

## Database Connection:

## Set up a database to store resume and biodata content

## CRUD (Create, Read, Update, Delete) operations for the admin to manage content

## 3. Database Design:

## Collections/Tables:

## Users: For storing user information (if needed)

## Resume: For storing resume details

## Biodata: For storing personal information

**Chapter 2**

**SYSTEM REQUIREMENT AND SPECIFICATION**

**2.1 Functional Requirements:**

**2.1.1 User Features**

* **View Resume:**
  + Users can access a dedicated page displaying the resume in a clear and structured format.
* **View Biodata:**
  + Users can view a dedicated page for biodata, which includes personal details, education, and work experience.
* **Download Resume:**
  + Users can download the resume in PDF format for offline access.
* **Contact Form:**
  + Users can fill out a contact form to send inquiries, which includes fields for name, email, and message. Upon submission, an email notification will be sent to the admin.

**2.1.2 Admin Features**

* **Login Authentication:**
  + Admin users can securely log in to manage content.
* **Update Resume:**
  + Admin can edit and save changes to the resume content via a dedicated admin interface.
* **Update Biodata:**
  + Admin can modify and save changes to the biodata content.
* **Manage Website Content:**
  + Admin can update overall site design and layout.

**2.2 HTML -tags**

**2.2.1 Structure and Markup**

* **HTML5 Compliance:**
  + The application will utilize HTML5 standards to ensure modern functionality and compatibility.
* **Page Structure:**
  + Each page (home, resume, biodata, contact) will have a consistent structure, including:
    - <header> for navigation
    - <main> for content
    - <footer> for contact information and links.
    1. **Basic Structure and Semantics**:
* HTML divides a page into meaningful sections using semantic tags such as <header>, <section>, <article>, <nav>, and <footer>. These tags help organize the content logically and make the page structure clear, both for users and search engines.
  + **DOCTYPE Declaration** (<!DOCTYPE html>) ensures compatibility across browsers by signaling that the document uses HTML5 standards, enabling a consistent experience.
    1. **Core Elements**:
  + **Headers and Navigation**: Tags like <header> and <nav> group titles, logos, and navigational links, providing users with intuitive navigation throughout the multi-page layout.
  + **Sections and Containers**: <section>, <div>, and <article> tags create individual content blocks. This helps to segment the page into distinct areas, making it easier to apply targeted styling or layout changes with CSS.
  + **Lists and Tables**: Ordered (<ol>) and unordered (<ul>) lists structure items in a readable format, useful for listing skills or work experiences. Tables (<table>) are applied sparingly but are effective for structuring specific information like biodata.

**2.2.4 Media Elements**:

* + HTML supports multimedia integration through tags like <img> for images and <video> or <audio> for media content. In the portfolio project, <img> is used to include profile pictures, project images, or icons.
  + **Alt Text** for Images: The alt attribute in <img> tags provides alternative text, ensuring that visual content is accessible to screen readers, which is critical for users with visual impairments.

**2.2.5 Forms and Interactivity**:

* + Forms, created with the <form>, <input>, <button>, and <label> tags, allow for user interaction by submitting feedback or contacting the students. Forms improve engagement and help collect relevant user data.
    1. **Hyperlinks**:
  + HTML anchors (<a>) provide navigation both within and between pages, allowing users to download resumes, visit social profiles, or navigate to additional resources.
  + External links and internal anchors (href attribute) improve navigation, giving users easy access to external resources or specific sections within a page.

**2.2.7** **Accessibility and ARIA Roles**:

* + HTML incorporates **ARIA (Accessible Rich Internet Applications)** attributes like role, aria-label, and aria-hidden to enhance accessibility for users with disabilities. These attributes ensure that screen readers can interpret the structure and purpose of each element, improving the user experience for visually impaired users.
  + Semantic HTML tags also play a role in accessibility, as they define the page's structure in a way that assistive technologies can interpret, making it easier for all users to understand the content.

**2.2.8 Metadata and SEO**:

* + HTML’s <head> section includes metadata such as <title>, <meta name="description">, and <meta name="keywords">, which are essential for search engine optimization (SEO). These elements provide context about the page, improving visibility in search results and enhancing the application’s discoverability.
  + **Charset and Viewport Settings**: <meta charset="UTF-8"> ensures proper character encoding, and <meta name="viewport" content="width=device-width, initial-scale=1.0"> supports responsive design by scaling the page to fit device screens.
    1. **Integration with CSS and JavaScript**:
  + HTML provides structure for CSS and JavaScript, using <link> tags to connect CSS files and <script> tags to add JavaScript or jQuery functionality. This modular setup allows separation of design, behavior, and content, making the application more manageable, scalable, and efficient.
  + By keeping HTML strictly for structure, it becomes easier to modify styles or scripts independently, enabling quick updates without affecting the underlying content.

**2.3 CSS (Cascading Style Sheets)**

1. **Basic Styling and Layout:**
   1. **Selectors**: CSS targets specific HTML elements using selectors, which define which elements are styled and how. Common selectors include element selectors (e.g., body, header), class selectors (.class-name), and ID selectors (#id-name). These selectors enable precise control over each page component.
   2. **Properties and Values**: Each element's appearance is adjusted using properties like color, font-size, margin, padding, and border. CSS properties and values allow developers to adjust the color scheme, spacing, and typography, creating a visually cohesive experience across pages.
   3. **Box Model**: The CSS box model, which includes margin, border, padding, and content, governs spacing and sizing of elements. This concept is crucial for achieving accurate alignment and a well-organized layout, particularly when designing resumes with sections and borders.
2. **Typography and Font Styles:**
   1. **Font Styling**: CSS enables customization of text fonts, sizes, colors, weights, and styles, allowing each resume or biodata page to have its unique typographic look. Using properties like font-family, font-size, and text-transform, CSS achieves a clear, professional aesthetic that enhances readability.
   2. **Font Import and Customization**: Fonts can be imported from external sources like Google Fonts using @import or <link> in the HTML <head>. This allows for a wide range of fonts and styles, ensuring the text aligns with the overall brand or theme of the portfolio.
3. **Color and Backgrounds:**
   1. **Color Schemes**: CSS properties like background-color, color, and gradients (linear-gradient, radial-gradient) define the application’s color scheme, contributing to a cohesive visual experience. Complementary colors for backgrounds and text improve readability while maintaining an elegant style.
   2. **Background Images and Patterns**: CSS allows for custom backgrounds with images, gradients, and patterns. These backgrounds can cover the whole page (background-size: cover;) or repeat for a patterned look, adding depth and character to each page.
   3. **Dynamic Colors**: CSS variables (--variable-name) enable dynamic color changes across the site. With variables, colors can be adjusted in one location and applied universally, simplifying maintenance and consistency.
4. **Responsive Design**:
   1. **Media Queries**: CSS media queries (@media) are used to create layouts that adapt to different screen sizes, ensuring that pages are responsive and functional across devices. Media queries adjust properties based on screen width, allowing elements to resize or rearrange for optimal viewing on mobile, tablet, and desktop.
   2. **Flexible Units**: CSS units like percentages (%), viewport width/height (vw, vh), and relative font sizes (em, rem) enable elements to adapt fluidly within the layout, contributing to responsiveness.
   3. **Flexbox and Grid Layouts**: CSS layout models like Flexbox (display: flex) and Grid (display: grid) offer advanced, responsive positioning for elements. Flexbox arranges items horizontally or vertically in a flexible container, while Grid allows complex, two-dimensional layouts ideal for organizing content sections in resumes or biodata pages.
5. **Animation and Transition Effects**:
   1. **CSS Animations**: CSS animations (@keyframes) create dynamic effects, like subtle motion or background shifts, to enhance user engagement. CSS animation properties (animation-name, animation-duration) control the timing, iteration, and easing of these effects.
   2. **Transitions**: CSS transitions (transition-property, transition-duration) create smooth changes when users interact with elements. For example, hover effects that enlarge buttons or highlight sections on resume or biodata pages make navigation more intuitive and visually appealing.
   3. **Hover and Active States**: Pseudo-classes like :hover, :focus, and :active change styles when users interact with elements, providing feedback that improves user engagement. Common uses include changing button colors, adding underlines, or enlarging images on hover.
6. **Visual Enhancements and Effects**:
   1. **Borders and Shadows**: CSS properties such as border, border-radius, and box-shadow add depth to elements, creating visual separation and emphasis. Rounded borders and subtle shadows on profile images or content boxes make the interface appear clean and professional.
   2. **Gradients and Opacity**: Gradient backgrounds (linear-gradient, radial-gradient) and opacity adjustments (opacity) add visual layers to the design, giving a modern, multi-dimensional look to sections or backgrounds without the need for images.
7. **Organization and Maintainability**:
   1. **Modular CSS**: CSS files can be modularized by section (e.g., one file for layout, another for colors, another for animations). Using multiple CSS files or sections within a single file (main.css, resume.css, biodata.css) helps maintain organization and makes styling updates manageable.
   2. **CSS Variables**: CSS variables (:root { --primary-color: #333; }) simplify updates by centralizing frequently used colors, fonts, or spacing values. They ensure consistency across pages and reduce redundancy, as changes are made in one place and reflected throughout the project.
8. **Integration with JavaScript/jQuery (if applicable)**:
   1. CSS can work alongside JavaScript or jQuery for advanced interactivity, such as dynamically changing styles based on user actions or page conditions. Although this project primarily relies on HTML and CSS, JavaScript could be used to add interactivity to animations or to dynamically toggle between themes.

**2.4 Cloud to be Hosted**

**2.4.1 Introduction**

This section outlines the requirements and specifications for hosting the web application on a cloud platform. Cloud hosting provides scalability, reliability, and easy management, making it an ideal choice for deploying the resume and biodata portfolio.

* + 1. **Cloud Hosting Features**

**1.Scalability**

* **Dynamic Scaling:** The hosting solution should support auto-scaling to handle varying levels of traffic without performance degradation.
* **Resource Allocation:** Ability to adjust resources (CPU, RAM, storage) based on user demand.

**2.Reliability**

* **Uptime Guarantee:** The cloud provider should offer a minimum uptime guarantee of 99.9%.
* **Backup Solutions:** Regular automated backups to prevent data loss.

**3.Security**

* **Data Encryption:** All data should be encrypted in transit (TLS/SSL) and at rest.
* **Access Control:** Implement user authentication and role-based access controls for managing cloud resources.
* **Firewalls:** Utilize cloud-based firewalls to protect against unauthorized access.

**2.4.3 Cloud Provider Options**

* **Amazon Web Services (AWS):** Offers a wide range of services, including EC2 for computing, S3 for storage, and RDS for databases.
* **Microsoft Azure:** Provides similar services with integrated tools for development and management.
* **Google Cloud Platform (GCP):** Known for its machine learning capabilities and robust computing options.
* **Netlify:** Netlify is a platform that simplifies the process of deploying and hosting web applications, particularly static sites and front-end frameworks.

**2.4.4 Deployment Process**

**1 Setup Requirements**

* **Cloud Account:** Create an account with the chosen cloud provider.
* **Virtual Server Configuration:** Set up a virtual server (e.g., AWS EC2 instance) with the necessary operating system (Linux/Windows).

**2 Software Installation**

* **Web Server:** Install a web server (e.g., Apache, Nginx) to serve the HTML, CSS, and other static files.
* **Database Server:** Set up a database server (e.g., MySQL, PostgreSQL) if dynamic content management is required.

**3 Deployment Steps**

1. **Prepare the Application:**
   * Ensure all files (HTML, CSS, JS) are organized and ready for deployment.
2. **Upload Files:**
   * Use FTP/SFTP or cloud storage services (like AWS S3) to upload files to the server.
3. **Configure Domain:**
   * Point your domain name to the cloud server's IP address.
   * Configure DNS settings as required.
4. **Test Deployment:**
   * Access the application via the domain to verify that it is functioning correctly.

**2.4.5 Monitoring and Maintenance**

* **Monitoring Tools:** Utilize cloud monitoring services (e.g., AWS CloudWatch) to track performance metrics and application health.
* **Regular Updates:** Schedule regular updates for software dependencies and the operating system to maintain security and performance.
* **User Feedback:** Gather user feedback periodically to identify areas for improvement and adjust resources as needed.

**2.4.6 Cost Considerations**

* **Pay-as-You-Go Pricing:** Most cloud providers offer flexible pricing models, allowing you to pay only for the resources you use.
* **Budget Management:** Set budget alerts to monitor spending and avoid unexpected charges

**2.5. Non Functional Requirements**

**2.5.1 Performance**

* The application should load within 3 seconds on standard broadband connections.
* Minimize the use of large images and excessive animations.

**2.5.2 Security**

* Use HTTPS to secure data transmission.
* Validate and sanitize user inputs to prevent XSS and SQL injection attacks.

**2.5.3 Usability**

* The interface should be intuitive and easy to navigate.
* Ensure that interactive elements provide visual feedback on hover and focus.

**2.5.4 Compatibility**

* The application must be compatible with major web browsers (Chrome, Firefox, Safari, Edge).
* CSS should adhere to modern standards (CSS3) for features and properties.

**2.5.5 Testing and Validation**

* Regularly validate HTML and CSS using W3C validators.
* Conduct usability testing to gather user feedback and improve the interface

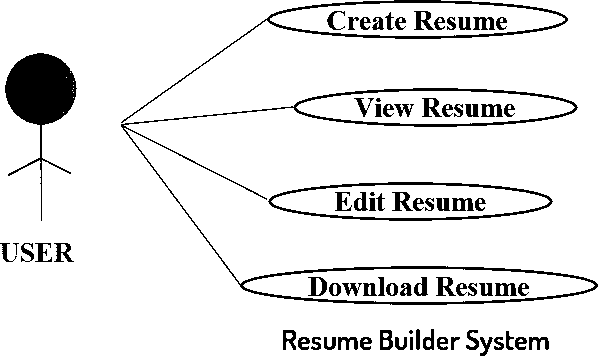
**Chapter 3**

## SYSTEM ANALYSIS AND DESIGN

## 3.1 Use Case

A use case diagram is a type of behavioural diagram in the Unified Modelling Language that represents the functional requirements of a system. It captures the interactions between users (actors) and the system to achieve a goal.

Use case diagram is used to define the scope of a system and its requirements. The primary elements of a use case diagram are actors and use cases. Actors are the users or external systems that interact with the system, while use cases represent the specific functionality or tasks that the system can perform.



**Fig.3.1: Use Case Diagram**

**Chapter 4**

**IMPLEMENTATION**

## 4.1 Algorithm

## Initialize Web Page:

## Load HTML structure and styles.

## Display Images:

## Render images centered on the homepage.

## Handle Button Clicks:

## On clicking "Learn More", redirect to the portfolio.html.

## On clicking team member links, redirect to respective biodata and resume pages.

## On clicking "Download Resume", initiate download of the PDF.

## Display Team Members:

## Render each team member's profile card with respective links.

## End.

## 4.2 Flow Chart

## Start

## 

## Load Homepage

## 

## Display Images

## 

## + User clicks "Learn More" Redirect to Portfolio

## + User views Team Members

## 

## + User clicks member link Redirect to Biodata/Resume

## 

## + User clicks "Download Resume" Initiate Download

End

## 4.3 Source code

* **Home page**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Web Page</title>

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

</head>

<body>

<header>

<h1>Welcome to Student Portfolio</h1>

</header>

<div class="imgs" >

<img src="image.jpg">

</div>

<main>

<a href="portfolio.html" class="button">Learn More</a>

</main>

</body>

</html>

* **Portfolio page**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Team Members</title>

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

</head>

<body>

<header>

<h1>Meet Our Team</h1>

</header>

<main>

<div class="card\_one">

<div class="member">

<h3>Ashwini C N</h3>

<p><a href="biodata.html" class="link" target="\_blank">View Biodata</a></p>

<p><a href="resume.html" class="link" target="\_blank"> View Resume</a></p>

</div>

</div>

<div class="card\_two">

<div class="member">

<h3>Chandana</h3>

<p><a href="biodata02.html" class="link">View Biodata</a></p>

<p><a href="resume02.html" class="link"> View Resume</a></p>

</div>

</div>

<div class="card\_three">

<div class="member">

<h3>Bhanu Prakash</h3>

<p><a href="biodata01.html" class="link">View Biodata</a></p>

<p><a href="resume01.html" class="link"> View Resume</a></p>

</div>

</div>

</main>

</body>

</html>

* **Resume Page**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

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

<title>Resume</title>

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">

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

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">

</head>

<body>

<nav class="navbar">

<marquee> Welcome to My Portfolio</marquee>

</nav>

<div class="resume">

<p>Hello! I am Ashwini CN, a third-year Computer Science student.</p>

<h1>Ashwini CN</h1>

<p>Email: <i class="glyphicon glyphicon-envelope"></i> ashwinicn2004@gamil.com</p>

<p>Phone: <i class="material-icons" style="font-size:15px">call</i> +91 8792762798</p>

<p>LinkedIn: <i class="fa fa-linkedin-square"></i> Ashwini C N</p>

<h2>Objective</h2>

<hr>

<p>Motivated third-year Computer Science student with strong foundation in programming, data structures and algorithms.

Seeking internship opportunities in marketing and IT field.</p>

<h2>Education</h2>

<hr>

<ul>

<li>West Hill Republic School,Channarayaptna &nbsp; &nbsp; &nbsp; &nbsp; 2010-2020<pre>10th standard 92.83%</pre></li>

<li>Times P U College,Channarayaptna &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; 2020-2022<pre>12th standard 95.66%</pre></li>

<li>Maharaja Institute of Technology,Thandavapura &nbsp; &nbsp; &nbsp; &nbsp; (Expected Graduation: 2026)<pre>B.E in Computer Science and Engineering 8.88CGPA</pre></li>

</ul>

<h2>Skills</h2>

<hr>

<ul>

<li>Python,HTML,CSS</li>

<li>Strong communication skills</li>

<li>Adaptability and Flexibility</li>

<li>Team work and Collabration</li>

</ul>

<h2>Extracurricular Activities</h2>

<hr>

<ul>

<li>Writing article</li>

<li>Volunteer for Campus Events</li>

</ul>

<h2>Declaration:</h2>

<hr>

<ul>

<li>I hereby declare that all the details furnished here are true to the best of my knowledge and belief.</li>

</ul>

</div>

</body>

</html>

* **Biodata Page**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

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

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">

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

<title>Biodata</title>

</head>

<body>

<nav class="navbar">

<marquee> Welcome to My Portfolio</marquee>

</nav>

<div class="biodata">

<p>Hello! I am Ashwini CN, a third-year Computer Science student.</p>

<h1>Ashwini CN</h1>

<div class="images" >

<img src="pic.jpg">

</div>

<p>Email: <i class="glyphicon glyphicon-envelope"></i> ashwinicn2004@gmail.com</p>

<p>Phone: <i class="material-icons" style="font-size:15px">call</i> +91 8792762798</p>

<p>Date of Birth: April 10, 2004</p>

<h2>Personal Information</h2>

<hr>

<p>Father's Name: Mr. Nagesh CS</p>

<p>Mother's Name: Mrs. Geetha</p>

<p>Age:20</p>

<p>Religion: Hindu</p>

<p>Nationality: Indian</p>

<h2>Languages Known</h2>

<hr>

<li>English</li>

<li>Kannada</li>

<h2>Qualification</h2>

<hr>

<ul>

<li>Bachelor of Engineering in Computer Science and Engineering - Visvesvaraya University (Expected Graduation: 2026)</li>

</ul>

<h2>Hobbies</h2>

<hr>

<li>Reading</li>

<li>Dancing</li>

<li>Traveling</li>

<h2>Address</h2>

<hr>

<li>Channarayaptna(taluk),Hassan(district),Kranataka-573116</li>

</div>

</body>

</html>

**4.4 Snapshots**

Fig:4.4.1

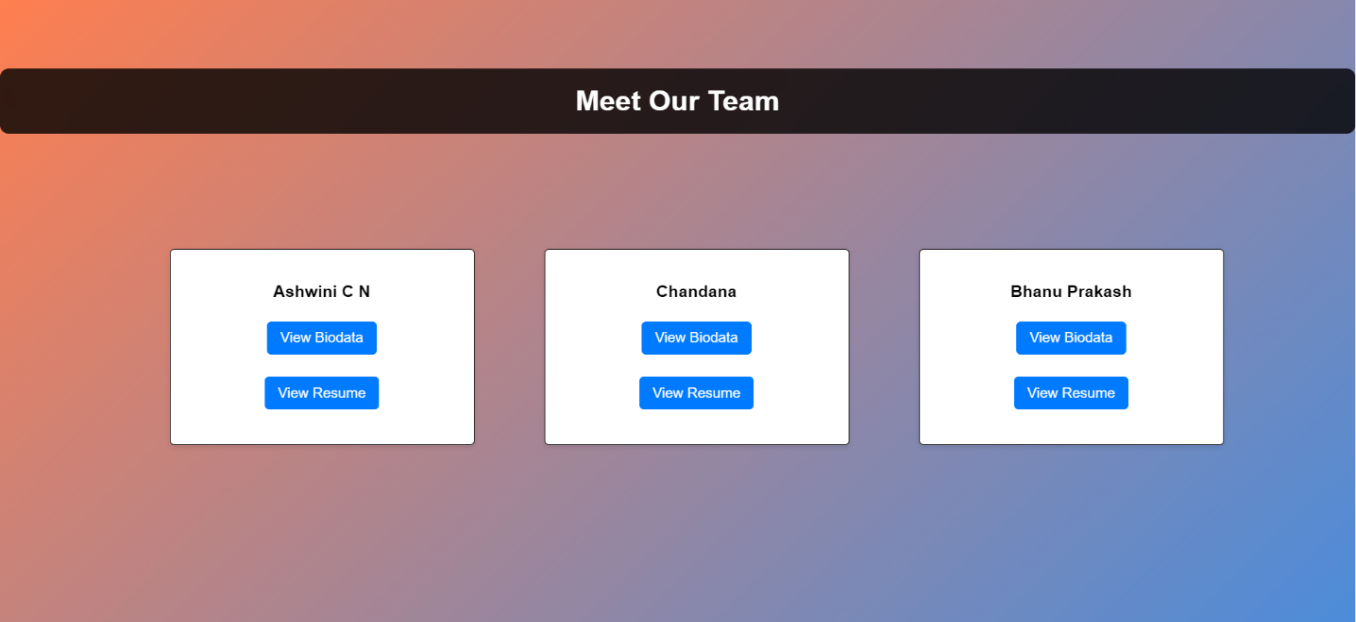
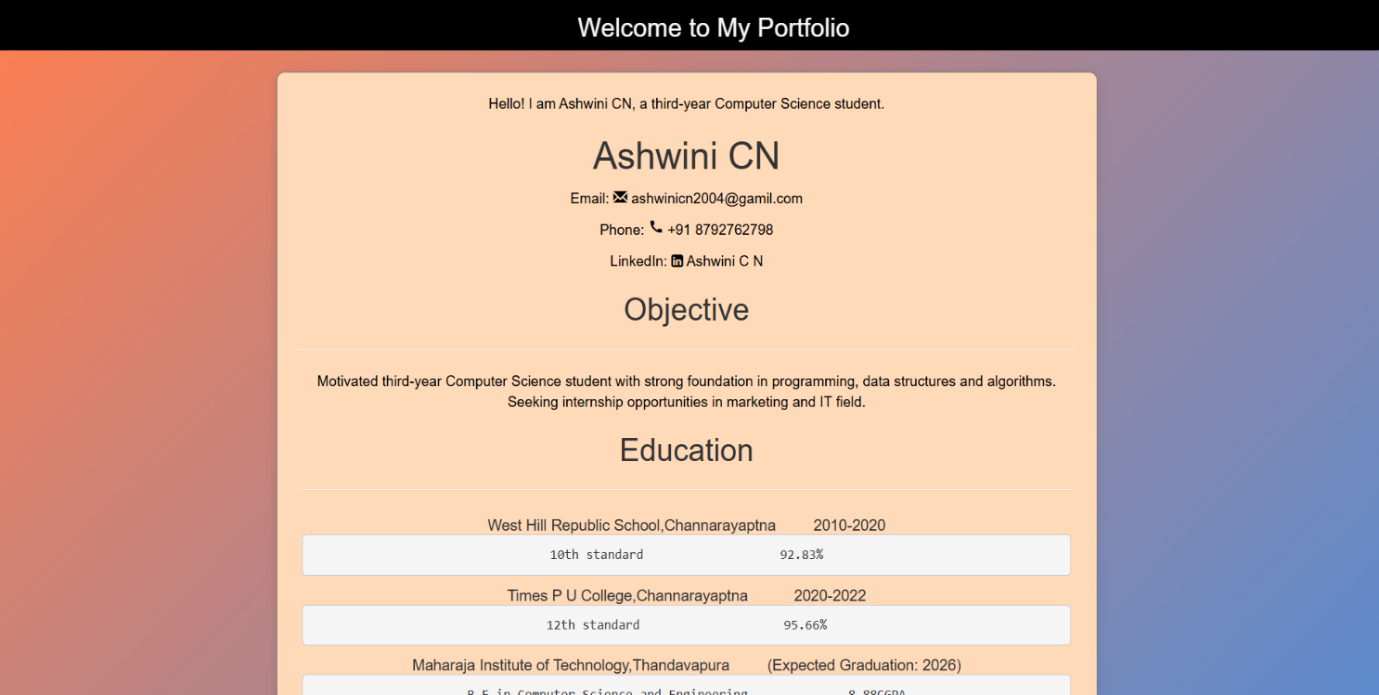
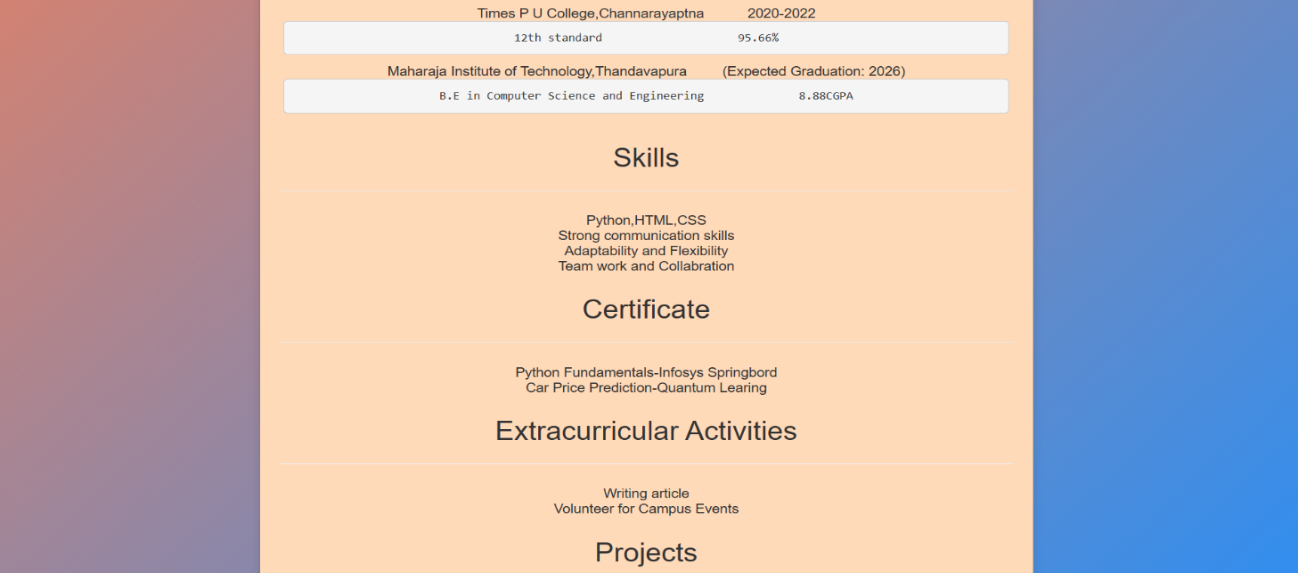


Fig:4.4.2





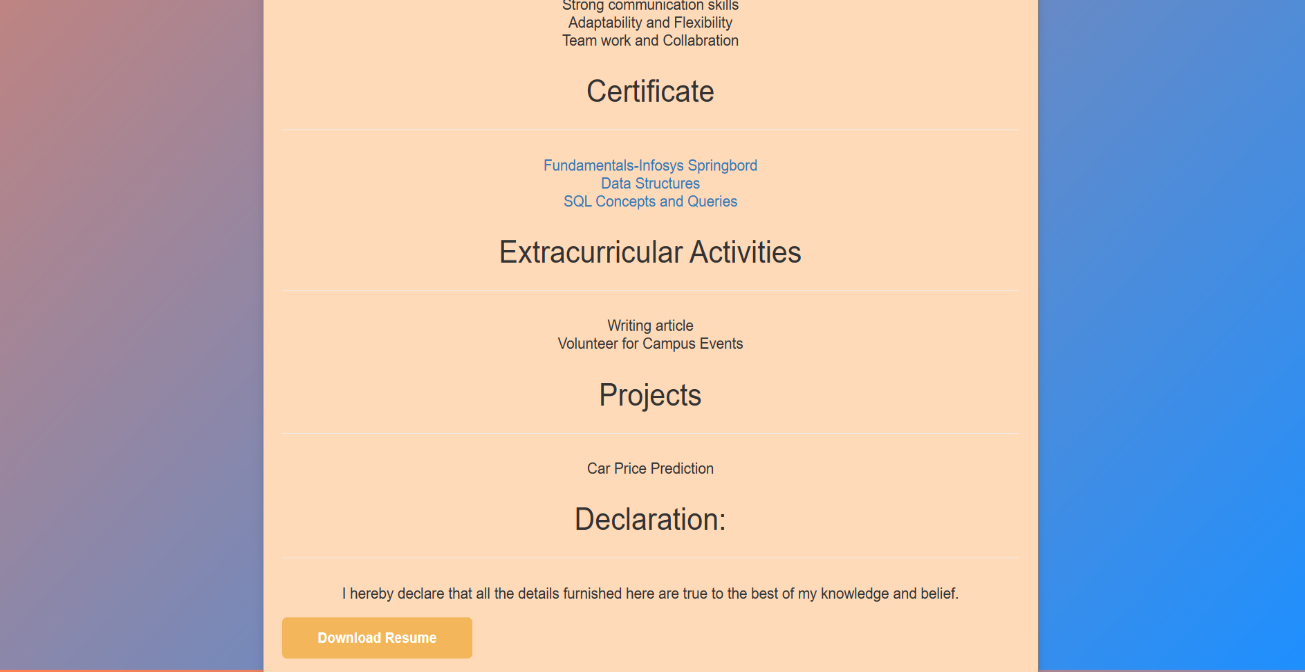


Fig:4.4.3



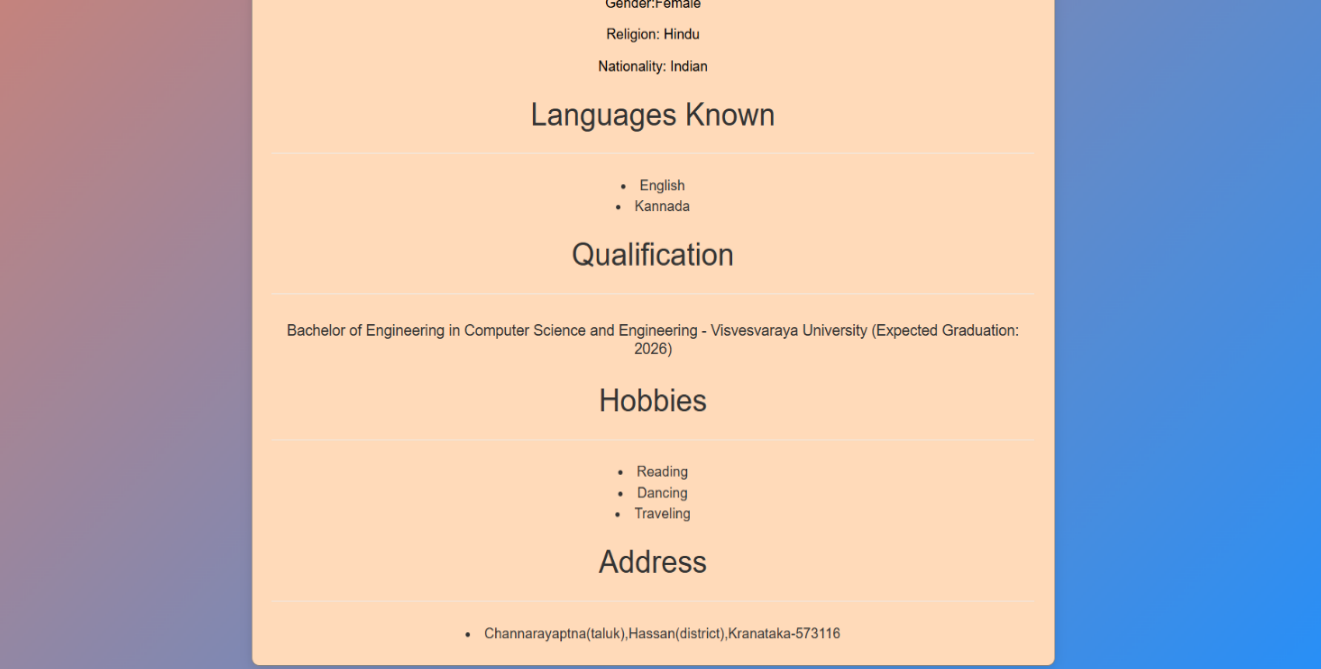


Fig:4.4.4

**CONCLUSION**

In conclusion, this website serves as a dynamic and comprehensive showcase of your professional identity, integrating your resume and biodata into a cohesive online portfolio. Through the use of HTML and CSS, we have crafted an aesthetically pleasing and user-friendly interface that highlights your skills, experiences, and achievements effectively.

The multi-page structure allows for organized and intuitive navigation, making it easy for potential employers, colleagues, or networking contacts to explore your qualifications in depth. Each section is designed to convey essential information clearly, ensuring that your unique value proposition stands out.

Additionally, the responsive design ensures that your content looks great on any device, from desktops to smartphones, providing accessibility and convenience for all visitors. As your career progresses, this platform can be easily updated to reflect new accomplishments, skills, or projects, keeping it relevant and current.

We encourage you to share this website with your professional network, and we hope it opens new opportunities and connections in your career journey. Thank you for visiting, and we look forward to seeing how you continue to grow and thrive!