

**VISVESVARAYA TECHNOLOGICAL UNIVERSITY  
JNANASANGAMA, BELAGAVI – 590018**



**INTERNSHIP REPORT**

*Submitted in partial fulfilment for the award of degree of*

**BACHELOR OF ENGINEERING**

*In*

**COMPUTER SCIENCE AND ENGINEERING**

*Submitted by*

**PRAJWAL NAIK**

**2VD20CS023**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING  
KLS VISHWANATHRAO DESHPANDE INSTITUTE OF TECHNOLOGY,  
HALIYAL– 581329**

**2023 – 24**

**KLS VISHWANATHRAO DESHPANDE INSTITUTE OF TECHNOLOGY,  
HALIYAL– 581329**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**



**CERTIFICATE**

This is to certify that **PRAJWAL NAIK** bearing **USN: 2VD20CS023**, a bonafied student of **KLS VISHWANATHRAO DESHPANDE INSTITUTE OF TECHNOLOGY** has undergone four weeks of **INTERNSHIP** at **TEQUED LABS**. This Internship report is submitted in partial fulfilment for the award of **Bachelor of Engineering in Computer Science and Engineering** of the **Visvesvaraya Technological University, Belagavi** during the academic year **2023-2024**.

.....  
Signature of Internal Guide  
Dr.. Naveenkumar T Rudrappa

.....  
Signature of HOD  
Dr. Shrinivas S. A.

.....  
Signature of Principal  
Dr. V. A. Kulkarni

Examination:

Name of the Examiners

Signature with date

1) .....

1).....

2) .....

2).....

# INTERNSHIP CERTIFICATE



**TEQUED LABS**  
**RESEARCH AND INNOVATION HUB**

RECOGNIZED BY  
**#startupindia**

## CERTIFICATE OF INTERNSHIP

THIS IS TO CERTIFY THAT

**Prajwal Naik**

from

**KLS Vishwanathrao Deshpande Institute of Technology**

has developed a project titled

**Resume Generator**

and successfully completed 1 Month Internship at Tequed Labs, Bangalore

**Internship ID: 2VD20CS023**

**Internship Domain: Full Stack Web Development**

**Internship Dates: Aug 15th 2023 to Sep 15th 2023**

**Supreeth Y S**  
**CEO**  
**Tequed Labs**

## **ACKNOWLEDGEMENT**

I am indebted to our Principal Dr. V. A. Kulkarni and management of KLS VEDIT for providing an environment with all facilities that helped me in completing the internship.

I am extremely grateful to Dr. Shrinivas Sirdeshpande, HoD of the Computer Science and Engineering Department for his moral support and encouragement.

I wish to express my sincere gratitude to my internal guide Prof. Naveenkumar T Rudrappa from the Computer Science & Engineering Department, for their guidance and suggestions.

I thank all the teaching and non-teaching staff of the Department of Computer Science and Engineering for their kind help.

I also thank the company Tequed Labs for providing me with this golden opportunity to do the internship on the topic Web Technology which helped me in learning new things.

Last but not the least, I would like to add some personal notes. If there is a driving force that keeps me going, and what has not changed, it is the constant support and blessing of my parents, family and friends. There is no doubt, in spite of my strenuous efforts; error might remain in the internship report. Naturally, I take full responsibility for any lack of clarity, occasional erratum or inexactness that may occur.

NAME: Prajwal Naik

USN:2VD20CS023

## Table of Contents

<b>Chapters</b>	<b>Page No.</b>
<b>1. Brief Introduction of the Organization's Business Sector</b>	<b>1</b>
<b>2. Plan of Internship Program</b>	<b>3</b>
<b>3. Training Program</b>	<b>4</b>
<b>4. Project carried out as assignment</b>	<b>17</b>
<b>5. Final project</b>	<b>24</b>
<b>6. Learning Experience</b>	<b>28</b>
<b>7. Conclusion</b>	<b>30</b>

## List of Figures

Figure No.	Figure Name	Page No.
3.3.1.1	Index	18
3.3.1.2	Services	18
3.3.1.3	About us	18
3.3.1.4	Appotiment	19
3.3.1.5	Login	19
3.2.1.1	Home page	24
3.2.1.2	Fotter	24
3.2.1.3	signup	25
3.2.1.4	Login	25
3.2.1.5	Data entry	26
3.2.1.6	Data entry (2)	26
3.2.1.7	Resume Pre-view	25
3.2.1.8	Save as PDF	27
3.2.1.9	Logout and data auto save	27

---

## CHAPTER 1

### Brief Introduction of the Organization's Business Sector

**Table 1.1** Basic Information of the Institution.

Company Name	TEQUED LABS PVT LTD
Management Type	Private
Business Type	Research and Development center
Established	2018
Directors	Mr Aditya S K and Mr Supreeth Y S
Address	Tequed Labs Pvt Ltd No 34/4A, 3rd floor, Dr.Vishnuvardhan Rd, near RNSIT COLLEGE, Channasandra, Uttarahalli Hobli, Bengaluru, Karnataka 56009
Email	info@tequedlabs.com
Phone	9480960424  8660939587
Website	<a href="https://www.tequedlabs.com">https://www.tequedlabs.com</a>

#### VISION:

Tequed Labs aims to empower individuals with the advanced skills and knowledge necessary to excel in the rapidly evolving technological landscape. Our vision is to cultivate a community of forward-thinking professionals and visionaries who are well-equipped to address real-world challenges and contribute to technological innovation. We are dedicated to providing top-quality education training in various cutting-edge technology domains to foster a highly skilled workforce for the future.

**MISSION:**

As for our mission, Tequed Labs is committed to delivering industry-relevant, practical-focused learning experiences through a combination of online and offline programs. Our courses are designed to bridge the gap between academic learning and industry demands, ensuring that our students are not only equipped with the latest skill sets but are also prepared to make significant contributions to their respective fields. We strive to offer globally recognized certifications and collaborate with leading institutions to ensure that our students receive the highest quality of education and professional growth opportunities.

Tequed Labs is a premier tech education company dedicated to providing industry-relevant, practical learning experiences. We offer a wide range of online and offline programs designed to bridge the gap between academic education and industry needs. Our certification programs cover Full Stack Web Development, Internet of Things, React-JS, AI and ML, Cyber Security, and Blockchain. With a strong focus on self-paced learning, we aim to equip our students with the latest skills and expert knowledge. Tequed Labs has been recognized as India's highest-rated Ed-Tech company on Google and has successfully trained over 13,500 students. Additionally, we recommend our trained students to more than 15,000 jobs available at JobSquare and offer premium job recommendations through JobSquare's SuperMatch feature. We collaborate with leading institutions and offer global certifications, empowering individuals to excel in the ever-evolving tech landscape.

The organization provides various online courses in technology-related fields, such as Full Stack Web Development, Internet of Things, React-JS, AI and ML, Cyber Security and Ethical Hacking, and Blockchain Certification Program. These courses offer in-depth learning experiences and are available for self-paced learning with access to expert teachers.

The nature of the organization's customer base encompasses individual learners seeking professional development and career advancement opportunities in the technology sector, as well as collaborations with educational institutions to extend their offerings to students.



---

## CHAPTER 2

### Plan of Internship Program

The internship program was conducted by Tequed Labs. The duration of the internship was from August 14<sup>th</sup> to 20<sup>th</sup> September.

**Table 2.1** Weekly Internship Schedule.

Week	Planned Topics
1	<ul style="list-style-type: none"><li>• Introduction to Full Stack Web Development</li><li>• HTML</li><li>• CSS</li><li>• Javascript</li></ul>
2	<ul style="list-style-type: none"><li>• jQuery</li><li>• BootStrap</li><li>• Portfolio Project</li></ul>
3	<ul style="list-style-type: none"><li>• PHP</li><li>• NodeJS</li><li>• Major Project Using FullSack</li></ul>
4	<ul style="list-style-type: none"><li>• Domain And Hosting</li><li>• INSDUSTRIAL MENTORSHIP SESSION</li></ul>

---

## CHAPTER 3

### Training Program

#### 3.1 Introduction to Full Stack Web Development

In today's digital era, web development plays a pivotal role in shaping the online landscape. As businesses strive to establish a robust online presence, skilled web developers are in high demand to create dynamic and interactive websites and web applications. A Full Stack Web Development program offers a comprehensive curriculum designed to equip individuals with the knowledge and skills needed to excel in this exciting field.

##### **What is Full Stack Web Development?**

Full Stack Web Development refers to the process of building both the front-end and back-end components of web applications. A full stack developer is proficient in working with both the client-side (front-end) and server-side (back-end) technologies, as well as databases and other essential components of web development.

##### **Key Components of a Full Stack Web Development Program:**

1. **Front-end Development:** This component focuses on building the user interface and user experience of web applications using technologies such as HTML, CSS, and JavaScript. Students learn to create visually appealing and responsive websites that provide seamless navigation and interaction.
2. **Back-end Development:** In the back-end development phase, students learn to work with server-side technologies such as Node.js, Python, Ruby on Rails, or Java Spring Boot. They learn to build and manage servers, handle user authentication, manage databases, and implement security measures to ensure the integrity and performance of web applications.
3. **Database Management:** Understanding databases is crucial for storing and retrieving data efficiently in web applications. Students learn SQL and NoSQL database technologies such as MySQL, MongoDB, or PostgreSQL, and gain hands-on experience in designing and querying databases.
4. **Frameworks and Libraries:** Full stack developers leverage frameworks and libraries to streamline the development process and enhance productivity. Students are introduced to popular frameworks and libraries such as React.js, Angular, Vue.js, Express.js, Django, Flask, and more, enabling them to build scalable and maintainable

---

web applications.

5. Version Control and Deployment: Proficiency in version control systems such as Git and knowledge of deployment techniques are essential skills for web developers. Students learn to collaborate on projects, track changes, and deploy web applications to production environments using platforms like Heroku, AWS, or Microsoft Azure.

**Benefits of a Full Stack Web Development Program:**

- Versatility: Graduates of a Full Stack Web Development program possess a diverse skill set that allows them to work on both front-end and back-end aspects of web development, making them versatile professionals in the industry.
- In-Demand Skills: The demand for skilled web developers continues to grow, with ample opportunities available in various industries ranging from tech startups to established corporations.
- Career Advancement: Full stack developers are well-positioned for career advancement, with opportunities to specialize in specific technologies or pursue leadership roles in development teams.

**Conclusion:**

A Full Stack Web Development program provides aspiring developers with the comprehensive knowledge and practical experience needed to thrive in the dynamic field of web development. By mastering both front-end and back-end technologies, graduates are prepared to create innovative and impactful web solutions that drive digital transformation and meet the evolving needs of businesses and users alike.

## 3.2 HTML

### 3.2.1 Introduction to HTML:

HTML serves as the foundation of web development, allowing developers to create structured documents that browsers can interpret and render as webpages. It consists of a series of elements, each enclosed in opening and closing tags, which define the structure and content of a webpage.

### 3.2.2 Basic Structure of an HTML Document:

An HTML document is composed of the following components:

html

```
<!DOCTYPE html>
```

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

```
<head>
```

```
  <meta charset="UTF-8">
```

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

```
  <title>Document</title>
```

```
</head>
```

```
<body>
```

```
  <!-- Content goes here -->
```

```
</body>
```

```
</html>
```

- `<!DOCTYPE html>`: Declares the document type and version of HTML being used.
- `<html>`: Represents the root element of the HTML document.
- `<head>`: Contains meta-information about the document, such as character encoding, viewport settings, and the page title.
- `<meta>`: Provides metadata about the HTML document, such as character encoding and viewport settings.
- `<title>`: Defines the title of the document, which appears in the browser's title bar or tab.
- `<body>`: Contains the content of the webpage, including text, images, links, and other elements.

---

### 3.2.3 HTML Tags:

HTML tags are used to define elements within an HTML document. They consist of an opening tag, content, and a closing tag. Tags are enclosed in angle brackets ``<>``.

Example:

```
<p>This is a paragraph.</p>
```

`<p>`: Opening tag of the paragraph element.

`This is a paragraph.`: Content of the paragraph.

`</p>`: Closing tag of the paragraph element.

Commonly Used HTML Tags:

1. Heading Tags: `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>`, `<h6>`
2. Paragraph Tag: `<p>`
3. Anchor Tag (Link): `<a>`
4. Image Tag: `<img>`
5. List Tags: `<ul>`, `<ol>`, `<li>`
6. Division Tag (Section): `<div>`
7. Span Tag (Inline Section): `<span>`
8. Table Tags: `<table>`, `<tr>`, `<th>`, `<td>`
9. Form Tags: `<form>`, `<input>`, `<button>`, `<select>`, `<textarea>`

### 3.2.4 Sections in HTML:

HTML documents are often organized into different sections to enhance readability and maintainability. Some commonly used sections include:

1. Header Section (`<header>`): Typically contains introductory content, such as a logo, site navigation, or introductory text.
2. Navigation Section (`<nav>`): Contains navigation links to different parts of the website or other webpages.
3. Main Content Section (`<main>`): Houses the primary content of the webpage, such as articles, blog posts, or product listings.
4. Sidebar Section (`<aside>`): Contains supplementary content, such as related links, advertisements, or call-to-action buttons.
5. Footer Section (`<footer>`): Typically includes copyright information, contact details, and links to social media profiles.

---

### 3.2.5 Conclusion:

HTML is a fundamental aspect of web development, serving as the backbone of webpages and web applications. By mastering HTML, developers gain the ability to create well-structured and visually appealing websites that effectively communicate information to users.

## 3.3 CSS ( Cascading Style Sheet)

### 3.3.1 Introduction to CSS:

CSS is a powerful tool that allows developers to control the visual presentation of web content. By applying styles to HTML elements, CSS enables the creation of visually appealing and consistent designs across webpages.

### 3.3.2 Basic Structure of a CSS Rule:

A CSS rule consists of a selector and one or more declarations enclosed in curly braces `{ }`. Each declaration includes a property and a value, separated by a colon `:`.

Example:

```
css
selector {
    property: value;
}
```

- **Selectors:** Selectors target HTML elements to which styles will be applied. They can target elements based on their tag name, class, ID, attributes, or relationship to other elements.
- **Properties:** Properties define the aspect of an element that will be styled, such as color, font size, margin, padding, etc.
- **Values:** Values specify the desired appearance or behavior for the selected property.

Example of a CSS Rule:

```
h1 {
    color: blue;
    font-size: 24px;
    margin-bottom: 20px;}
```

---

In this example:

- The selector `h1` targets all `<h1>` elements.
- The `color` property sets the text color to blue.
- The `font-size` property sets the font size to 24 pixels.
- The `margin-bottom` property sets the bottom margin to 20 pixels.

### 3.3.3 Common CSS Properties:

1. **Color:** `color`
2. **Font:** `font-family`, `font-size`, `font-weight`, `font-style`
3. **Text:** `text-align`, `text-decoration`, `line-height`, `letter-spacing`
4. **Background:** `background-color`, `background-image`, `background-position`, `background-size`
5. **Box Model:** `margin`, `padding`, `border`, `width`, `height`
6. **Positioning:** `position`, `top`, `right`, `bottom`, `left`
7. **Flexbox:** `display: flex`, `flex-direction`, `justify-content`, `align-items`
8. **Grid:** `display: grid`, `grid-template-columns`, `grid-template-rows`, `grid-gap`

### 3.3.4 Linking CSS to HTML:

CSS can be applied to HTML documents in several ways:

1. **Inline CSS:** Styles are applied directly to HTML elements using the `style` attribute.
2. **Internal CSS:** Styles are defined within the `<style>` element in the `<head>` section of an HTML document.
3. **External CSS:** Styles are stored in a separate CSS file and linked to the HTML document using the `<link>` element.

Example:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <style>
```

---

```
    h1 {  
        color: blue;  
font-size: 24px;  
        margin-bottom: 20px;  
    }  
</style>  
<link rel="stylesheet" href="styles.css">  
</head>  
<body>  
    <h1>Hello, CSS!</h1>  
</body>  
</html>
```

### 3.3.5 Conclusion:

CSS is an essential component of web development, enabling developers to create visually appealing and user-friendly webpages. By mastering CSS, developers can effectively control the presentation and layout of HTML content, resulting in engaging and professional-looking websites.

## 3.4 JavaScript

### 3.4.1 Introduction to JavaScript

JavaScript is a high-level, interpreted programming language that is widely used for client-side scripting in web development. It was originally created to make web pages more dynamic and interactive by allowing developers to manipulate the content and behavior of web pages.

### 3.4.2 Key Features of JavaScript

1. **Client-Side Scripting:** JavaScript is primarily used for client-side scripting, meaning it runs on the user's web browser rather than on a web server. This allows for the creation of dynamic and interactive user interfaces.
2. **Event-Driven Programming:** JavaScript is event-driven, which means it responds to user actions such as clicks, mouse movements, and keyboard inputs. Developers can



---

define event handlers to execute code in response to these events.

3. Object-Oriented Programming: JavaScript is an object-oriented language, allowing developers to create and manipulate objects with properties and methods. Objects can represent real-world entities or abstract concepts, making it easier to organize .

4. Cross-Platform Compatibility: JavaScript is supported by all modern web browsers, making it a cross-platform language. This means that JavaScript code written for one browser will generally work in other browsers as well, ensuring broad compatibility across different platforms and devices.

### 3.4.3 Basic Syntax of JavaScript:

JavaScript code is typically embedded directly into HTML documents using ``<script>`` tags, or it can be included from external JavaScript files.

Example:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>JavaScript Example</title>
  <script>
    // JavaScript code goes here
    alert("Hello, World!");
  </script>
</head>
<body>
  <!-- HTML content goes here -->
</body>
</html>
```

In the above example, the JavaScript code enclosed in ``<script>`` tags displays an alert dialog with the message "Hello, World!" when the web page is loaded.

### 3.4.3 Key Concepts of JavaScript:

1. Variables and Data Types: JavaScript supports various data types such as numbers, strings, booleans, arrays, objects, and more. Variables are used to store and manipulate data within a program.

2. Operators: JavaScript includes arithmetic, comparison, logical, assignment, and

---

other operators for performing operations on data.

3. Control Structures: JavaScript supports conditional statements (if...else), loops (for, while, do...while), and switch statements for controlling the flow of program execution.

4. Functions: Functions in JavaScript allow developers to encapsulate reusable blocks of code. They can accept parameters and return values, making code modular and easier to maintain.

5. DOM Manipulation: The Document Object Model (DOM) represents the structure of an HTML document as a hierarchical tree of objects. JavaScript can interact with the DOM to dynamically update or modify the content and appearance of web pages.

6. Events: JavaScript can respond to various events triggered by user actions or browser actions. Event handlers can be attached to HTML elements to execute JavaScript code in response to these events.

### **3.4.5 Conclusion:**

JavaScript is a powerful and versatile programming language that is essential for modern web development. By mastering JavaScript, developers can create interactive and dynamic web applications that provide rich user experiences and enhance the functionality of websites.

## **3.5 jQuery**

### **3.5.1 Introduction to jQuery:**

jQuery is a powerful and lightweight JavaScript library that simplifies the process of interacting with HTML documents and manipulating the DOM (Document Object Model). It provides a set of methods and functions that make it easier to perform common tasks such as selecting elements, handling events, animating elements, and making AJAX requests.

### **3.5.2 Why Use jQuery?**

1. Cross-browser Compatibility: jQuery handles browser quirks and inconsistencies, ensuring consistent behavior across different browsers.
2. Simplified Syntax: jQuery's concise syntax allows developers to write less code while achieving the same results compared to traditional JavaScript.
3. DOM Manipulation: jQuery simplifies the process of selecting and manipulating HTML elements, making it easier to create dynamic and interactive web pages.

---

4. Event Handling: jQuery provides methods for efficiently handling user interactions and events, such as clicks, mouse movements, and keyboard inputs.

5. AJAX Support: jQuery simplifies the process of making asynchronous HTTP requests, allowing developers to fetch data from servers and update the webpage without reloading the entire page.

### 3.5.3 Getting Started with jQuery

To use jQuery in your web projects, you can include the jQuery library in your HTML document either by downloading it locally or by linking to a hosted version. Here's how you can include jQuery using a CDN (Content Delivery Network):

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>jQuery Basics</title>
  <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
</head>
<body>
  <!-- Your HTML content goes here -->
  <script>
    // jQuery code goes here
  </script>
</body>
</html>
```

### 3.5.4 Selecting Elements with jQuery

jQuery provides powerful selectors that allow you to target HTML elements using CSS-like syntax. Here are some common jQuery selectors:

- ``$( 'element' )``: Selects all elements with the specified tag name.
- ``$( '#id' )``: Selects the element with the specified ID attribute.
- ``$( '.class' )``: Selects all elements with the specified class name.
- ``$( '[attribute]' )``: Selects all elements with the specified attribute.

Example:

```
$(document).ready(function() {  
  // Select all paragraphs and change their text color  
  $('p').css('color', 'red'); });
```

### 3.5.5 Manipulating Elements with jQuery:

jQuery provides methods for manipulating HTML elements, such as changing their content, attributes, and styles.

Example:

```
$(document).ready(function() {  
  // Change the text content of an element  
  $('#myElement').text('Hello, jQuery!');  
  // Change the CSS style of an element  
  $('.box').css({  
    'background-color': 'blue',  
    'color': 'white'  
  });  
  // Add a new class to an element  
  $('button').addClass('btn-primary');  
});
```

### 3.5.6 Handling Events with jQuery

jQuery simplifies event handling by providing methods for attaching event listeners to HTML elements and executing code in response to user interactions.

Example:

```
$(document).ready(function() {  
  // Handle click event  
  $('button').click(function() {  
    alert('Button clicked!');  
  });  
  // Handle mouseover event  
  $('img').mouseover(function() {  
    $(this).css('border', '2px solid red');  
  });  
});
```

---

### 3.5.7 Conclusion

jQuery is a versatile and powerful JavaScript library that simplifies web development by providing a concise and efficient API for common tasks. By mastering jQuery, developers can enhance their productivity, create interactive user interfaces, and build dynamic web applications with ease.

## 3.6 Bootstrap

### 3.6.1 Introduction to Bootstrap

Bootstrap is an open-source front-end framework that enables developers to design and customize responsive web layouts quickly and efficiently. It consists of a set of CSS classes, JavaScript plugins, and pre-designed components that can be easily integrated into web projects to streamline development and ensure consistency across different devices and screen sizes.

### 3.6.2 Key Features of Bootstrap

1. **Responsive Design:** Bootstrap is built with a mobile-first approach, meaning that websites and web applications created with Bootstrap are automatically optimized for various screen sizes, including desktops, tablets, and smartphones.
2. **Grid System:** Bootstrap includes a powerful grid system based on a 12-column layout, allowing developers to create flexible and responsive grid-based layouts. The grid system simplifies the alignment of content and enables the creation of complex, multi-column designs.
3. **Pre-Designed Components:** Bootstrap provides a comprehensive collection of reusable HTML components, such as navigation bars, buttons, forms, cards, modals, and more. These components can be easily customized and styled to match the design requirements of a project.
4. **CSS Flexbox and Grid Support:** Bootstrap 5 introduced full support for CSS Flexbox and Grid, offering more flexibility and control over layout and alignment options.
5. **Customization Options:** Bootstrap offers extensive customization options through SASS variables and mixins, allowing developers to tailor the framework to their specific needs and preferences.

---

6. Browser Compatibility: Bootstrap ensures cross-browser compatibility, meaning that websites and web applications built with Bootstrap are compatible with all modern web browsers, including Chrome, Firefox, Safari, and Edge.

### 3.6.3 Getting Started with Bootstrap

To start using Bootstrap in your web project, you can include the Bootstrap CSS and JavaScript files directly into your HTML document or use a package manager such as npm or yarn to install Bootstrap.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Bootstrap Example</title>
  <!-- Bootstrap CSS -->
  <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0alpha1/dist/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body>
  <!-- Bootstrap components and content go here -->

  <!-- Bootstrap JavaScript -->
  <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-
alpha1/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

### 3.6.4 Commonly Used Bootstrap Components:

1. Navbar: A navigation bar that provides site navigation links and branding.
2. Buttons: Stylish buttons with various styles and sizes.
3. Forms: Pre-styled form components such as input fields, checkboxes, radio buttons, and select dropdowns.
4. Cards: Flexible and customizable containers for displaying content, including

---

images, text, and buttons.

5. Modals: Popup dialogs that overlay the main content to display additional information or user interactions.

6. Carousel: Interactive slideshows for showcasing images or content in a carousel format.

7. Alerts: Dismissible notifications for providing feedback or alerts to users.

### **3.6.5 Conclusion:**

Bootstrap is a powerful and versatile front-end framework that simplifies the process of building responsive and visually appealing websites and web applications. By leveraging Bootstrap's extensive collection of components and utilities, developers can create modern and user-friendly interfaces that adapt seamlessly to different devices and screen sizes.

## **3.3 Project carried out as assignment**

Beautyblender is a focused assignment project, combining a beauty product and booking website with an emphasis on appointment scheduling. The platform allows users to easily create accounts, sign in, and personalize their experience. The user-friendly interface facilitates the booking of various beauty services, making it a convenient one-stop destination for individuals seeking personalized appointments. This project prioritizes the seamless integration of appointment scheduling features while maintaining simplicity and user-centric design principles, offering a comprehensive solution tailored to the diverse needs of beauty enthusiasts.

### 3.3.1 Snapshot

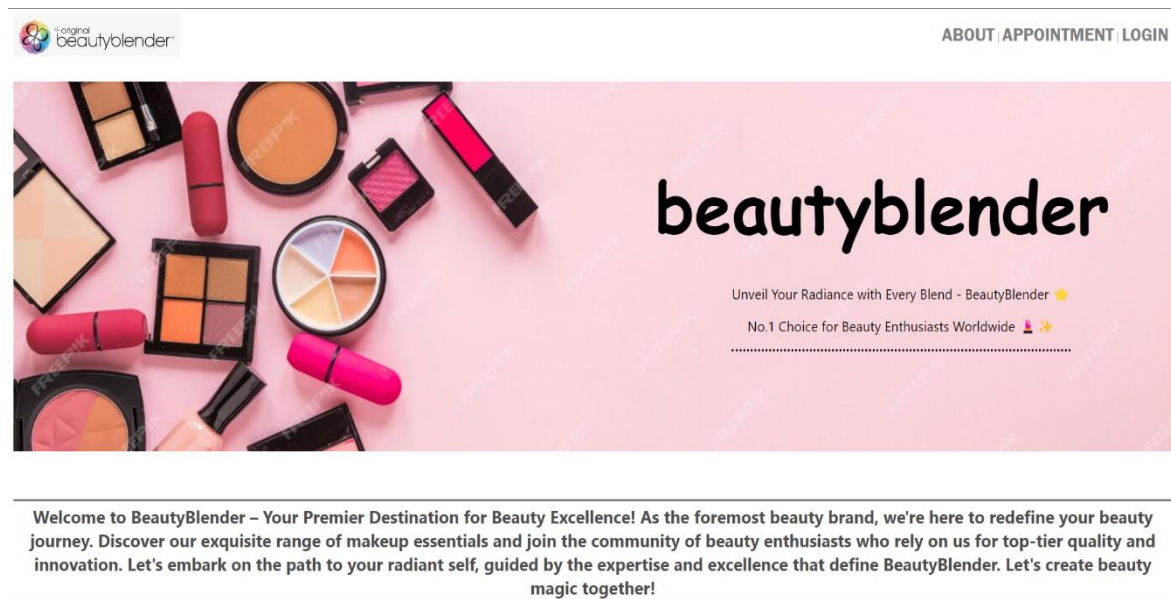


Fig. 3.3.1.1 Index

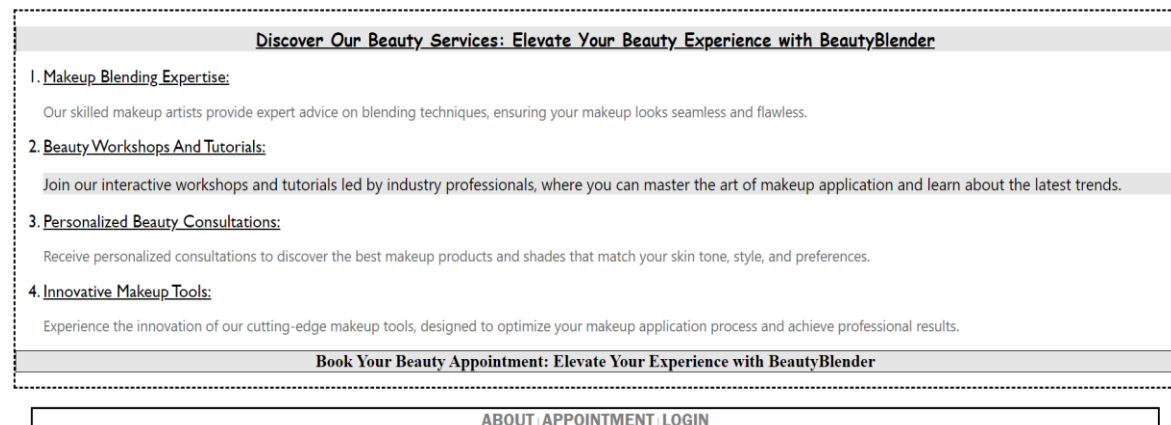


Fig. 3.3.1.2 services

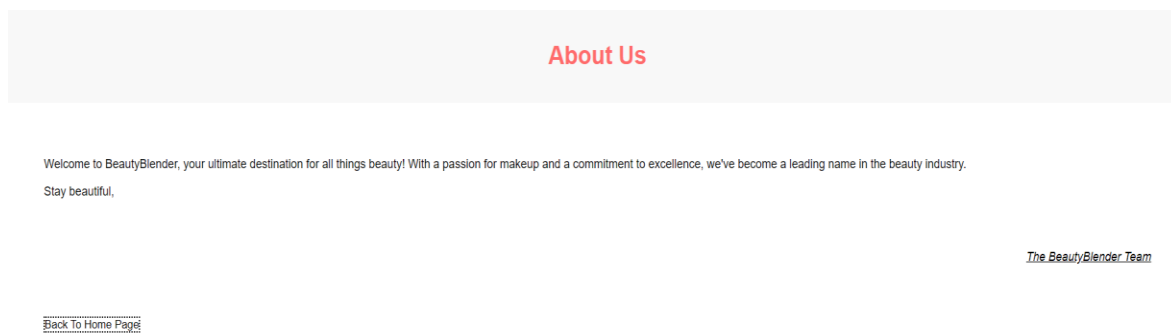


Fig. 3.3.1.3 about\_us



---

***FILL THE FORM To BOOK YOUR APPOINTMENT***

**APPOTIMENT**

i what to book a appotiment!

DEFOOOOOOOO!

[Back To Home Page](#)

Fig. 3.3.1.4 Appotiment

**LOGIN**

Email

Password

[Back To Home Page](#)

Fig. 3.3.1.5 Login

---

## 3.8 PHP (Hypertext Preprocessor)

### 3.8.1 Introduction to PHP

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. It is commonly used to create dynamic web pages, process form data, manage sessions and cookies, interact with databases, and perform various server-side tasks.

### 3.8.2 Key Features of PHP

1. Open Source: PHP is an open-source language, meaning it is freely available for anyone to use, modify, and distribute. It has a large and active community of developers who contribute to its development and maintenance.
2. Server-Side Scripting: PHP is executed on the server, meaning that PHP code is processed on the web server before the resulting HTML is sent to the client's web browser. This allows for dynamic generation of web content based on user inputs or other factors.
3. Cross-Platform Compatibility: PHP is platform-independent, meaning it can run on various operating systems such as Windows, Linux, macOS, and others. It is supported by most web hosting providers and integrates seamlessly with popular web servers like Apache and Nginx.
4. Easy to Learn: PHP has a syntax that is similar to C and other programming languages, making it relatively easy to learn for developers familiar with those languages. It also has extensive documentation and a wealth of online resources available for learning and reference.

### 3.8.3 Basic Syntax of PHP

PHP code is embedded directly into HTML documents using special opening and closing tags `<?php ... ?>`. PHP statements can be placed anywhere within an HTML document.

Example:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

---

```
<title>PHP Example</title>
</head>
<body>
  <h1> <?php echo "Hello, World!"; ?> </h1>
</body>
</html>
```

In the above example, the PHP code within the ``<?php ... ?>`` tags dynamically generates the content of the ``<h1>`` heading element, displaying the message "Hello, World!" in the web page.

### 3.8.4 Key Concepts of PHP

1. **Variables and Data Types:** PHP supports various data types such as strings, numbers, booleans, arrays, and objects. Variables in PHP are preceded by the `\$` symbol and can hold different types of data.
2. **Control Structures:** PHP includes control structures such as if...else statements, loops (for, while, do...while), switch statements, and foreach loops for controlling the flow of program execution.
3. **Functions:** PHP supports the creation of user-defined functions, which allow developers to encapsulate reusable blocks of code. Functions can accept parameters and return values, making code modular and easier to maintain.
4. **Arrays:** PHP provides robust support for arrays, which are used to store collections of data. Arrays can be indexed or associative, and PHP offers a wide range of array manipulation functions.
5. **File Handling:** PHP allows for reading from and writing to files on the server using file handling functions such as fopen(), fread(), fwrite(), fclose(), and others.
6. **Database Connectivity:** PHP has built-in support for connecting to and interacting with databases such as MySQL, PostgreSQL, SQLite, and others. It provides functions and extensions for executing SQL queries, fetching results, and managing database connections.

### 3.8.5 Conclusion

PHP is a powerful and versatile server-side scripting language that is widely used for web development. By mastering PHP, developers can create dynamic and interactive web applications that provide rich user experiences and functionality.

## 3.9 NodeJS

### 3.9.1 Introduction to Node.js

Node.js is an open-source, cross-platform runtime environment that enables developers to run JavaScript code on the server-side. It utilizes the V8 JavaScript engine, developed by Google for use in Chrome, to execute JavaScript code with high performance and efficiency.

### 3.9.2 Key Features of Node.js

1. **Asynchronous and Event-Driven:** Node.js uses an event-driven, non-blocking I/O model, which allows for handling multiple concurrent operations without blocking the execution of other code. This asynchronous nature makes Node.js highly scalable and efficient for building real-time applications.
2. **JavaScript on the Server-Side:** With Node.js, developers can use JavaScript not only for client-side scripting but also for server-side programming. This enables full-stack JavaScript development, where developers can use the same language and codebase for both client-side and server-side components of an application.
3. **Built-in Modules:** Node.js provides a rich set of built-in modules for performing various tasks such as file system operations, networking, HTTP handling, and more.

These modules, along with the npm (Node Package Manager) ecosystem, make it easy to extend Node.js functionality by installing and using third-party packages.

4. **Scalability:** Node.js is well-suited for building highly scalable applications, thanks to its non-blocking I/O model and event-driven architecture. It can handle large numbers of concurrent connections efficiently, making it ideal for building real-time web applications, APIs, and microservices.
5. **Community and Ecosystem:** Node.js has a vibrant and active community of developers contributing to its growth and evolution. The npm registry hosts over a million packages, providing a vast ecosystem of libraries and tools for building a wide range of applications.

### 3.9.3 Getting Started with Node.js

To start using Node.js, you'll need to install it on your machine. You can download and install Node.js from the official website: [Node.js Downloads](<https://nodejs.org/en/download/>).

---

Once Node.js is installed, you can create a new Node.js project by creating a directory for your project and running ``npm init`` to initialize a new ``package.json`` file. This file will contain metadata about your project and dependencies.

### 3.9.4 Creating a Simple Node.js Application

Here's an example of a simple Node.js application that creates a basic HTTP server:

```
// Import the 'http' module
const http = require('http');

// Define the port number
const PORT = 3000;

// Create a HTTP server
const server = http.createServer((req, res) => {
  // Set the response header
  res.writeHead(200, {'Content-Type': 'text/plain'});
  // Send the response
  res.end('Hello, Node.js!');
});

// Start the server and listen for incoming requests
server.listen(PORT, () => {
  console.log(`Server running at http://localhost:${PORT}/`);
});
```

Save this code in a file named ``app.js``. You can then run the application by executing the command ``node app.js`` in your terminal. This will start a HTTP server listening on port 3000, and you can access it in your web browser at ``http://localhost:3000``.

### Conclusion

Node.js is a versatile and powerful runtime environment for building server-side applications with JavaScript. By mastering Node.js, developers gain the ability to create scalable, efficient, and real-time applications that can handle large numbers of concurrent connections.

### 3.2.1 Snapshot

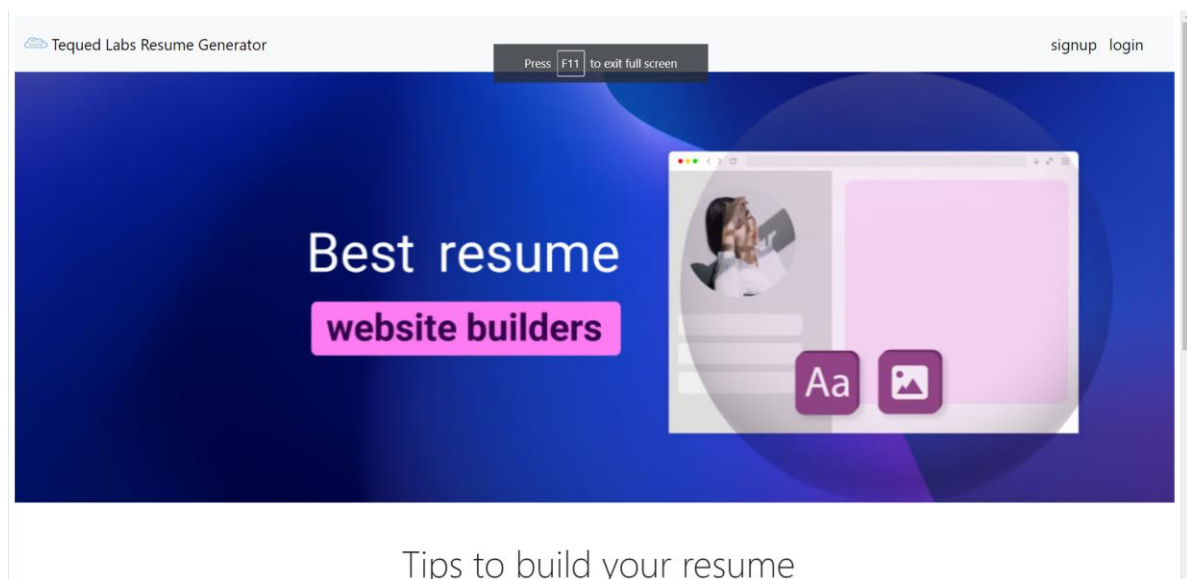


Fig. 3.2.1.1 Home page

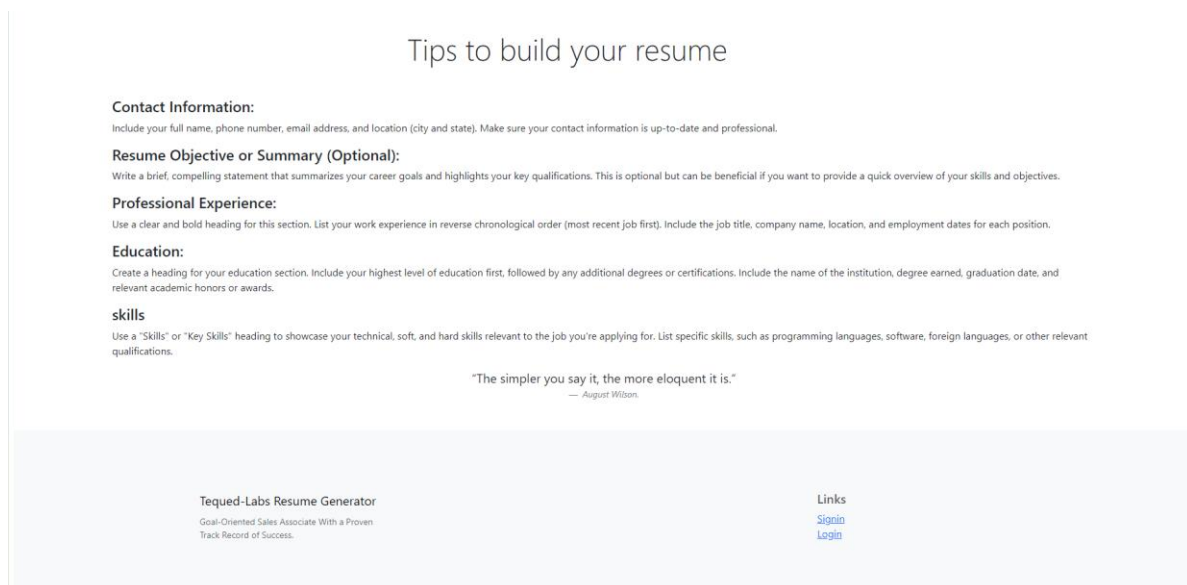


Fig. 3.2.1.2 Footer

The screenshot shows the 'tequedlabs resume builder' header with 'Home' and 'login' links. The main form area contains three input fields labeled 'Email', 'Phno', and 'Password'. Below these is a blue 'signup' button. At the bottom, there is a footer section with the text 'Tequed-Labs Resume Generator' and 'Goal-Oriented Sales Associate With a Proven Track Record of Success.' on the left, and 'Links' with a 'login' link on the right.

Fig. 3.2.1.3 Signup

The screenshot shows the 'tequedlabs resume builder' header with 'Home' and 'signup' links. The main form area contains two input fields labeled 'Email' and 'Password'. Below these is a blue 'LOGIN' button. At the bottom, there is a footer section with the text 'Tequed-Labs Resume Generator' and 'Goal-Oriented Sales Associate With a Proven Track Record of Success.' on the left, and 'Links' with a 'Signin' link on the right.

Fig. 3.2.1.4 Login

The screenshot shows the 'ACHIEVEMENTS' section with a form containing two input fields: 'Title' (with 'test' entered) and 'Description' (with 'test' entered). Below this is the 'EXPERIENCE' section with a form containing six input fields: 'Title' (with 'test'), 'Company / Organization' (with 'test'), 'Location' (with 'test'), 'Start Date' (with '24-02-2024' and a calendar icon), 'End Date' (with '08-02-2024' and a calendar icon), and 'Description' (with 'test').

Fig. 3.2.1.5 Data entry

**EDUCATION**

School

test

Degree

test

City

test

Start Date

04-02-2024

End Date

24-02-2024

Description

test

**PROJECTS**

Project Name

test


Project link

test

Description

test

Fig. 3.2.1.6 Data entry (2)



test test test

TEST

ABOUT

test

test

test

test

SKILLS

test,test,test,test

**ACHIEVEMENTS**

test

test

**EDUCATIONS**

test

test test 2024-02-04 2024-02-24

test

**EXPERIENCES**

test

test test 2024-02-24 2024-02-08

test

Fig. 3.2.1.7 Resume pre-view



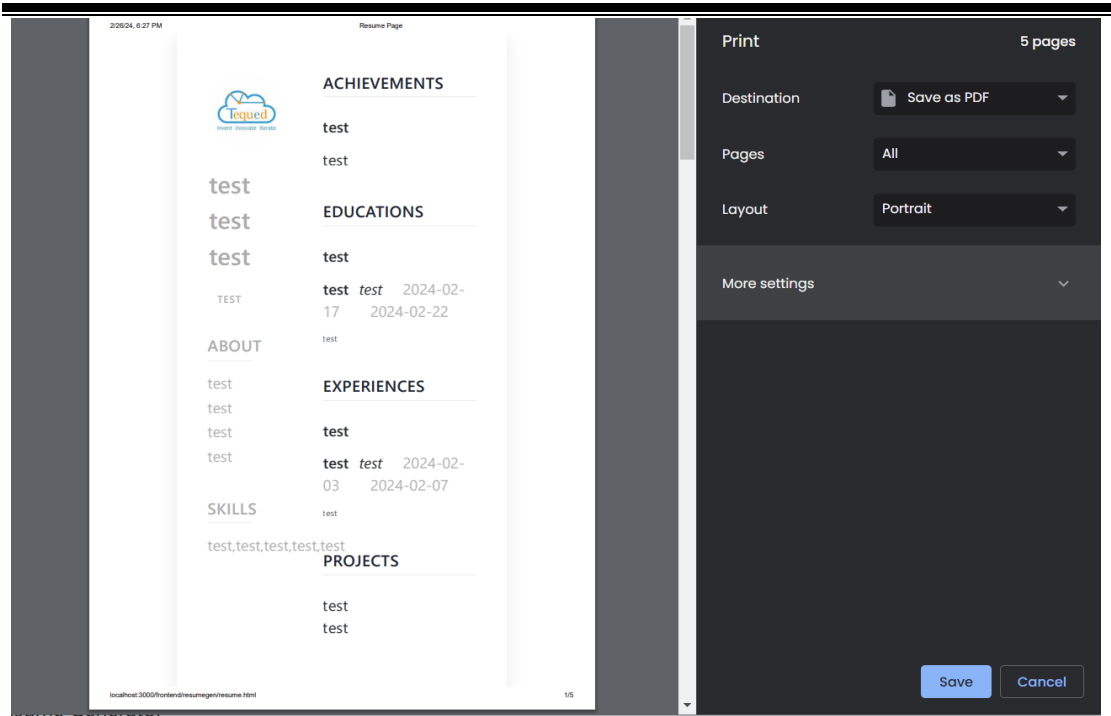


Fig. 3.2.1.8 Save as PDF

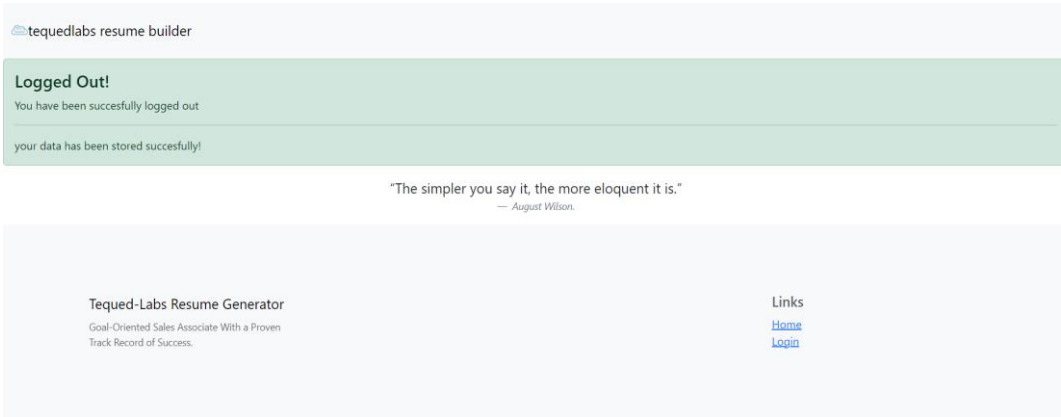


Fig. 3.2.1.9 Logout & data auto save

---

## CHAPTER 4

### Learning Experience

As an intern at TEQUED LABS PVT LTD, I've had a valuable learning experience that has allowed me to acquire new knowledge, gain skills, and tackle challenging tasks. Here's an overview of my internship experience:

#### 4.1 Knowledge Acquired

During my internship at TEQUED LABS, I've gained valuable knowledge in various areas of software development, including:

- Front-end development: I've learned about HTML, CSS, and JavaScript, which are essential for building the user interface and interactivity of web applications.
- Back-end development: I've been introduced to server-side programming languages such as Node.js and PHP, as well as databases like MySQL and MongoDB, enabling me to understand how data is processed and managed on the server.
- Version control: I've learned how to use Git for version control, allowing me to collaborate with team members and track changes in code repositories.
- Agile methodologies: I've been exposed to Agile development practices such as Scrum and Kanban, gaining an understanding of how teams organize and manage software development projects in an iterative and incremental manner.

#### 4.2 Skills Gained

My internship at TEQUED LABS has equipped me with a range of technical and soft skills, including:

- Proficiency in front-end technologies: I've become proficient in building responsive and visually appealing web interfaces using HTML, CSS, and JavaScript, and I've gained experience with frameworks like Bootstrap and libraries like jQuery.
- Back-end development skills: I've learned how to create server-side applications using Node.js and PHP, and I've acquired knowledge of RESTful API design principles for building scalable and maintainable web services.
- Problem-solving abilities: Through various coding challenges and projects, I've honed my problem-solving skills and learned how to approach complex technical problems systematically.

- Communication and teamwork: Working closely with my colleagues and mentors, I've improved my communication skills and learned how to collaborate effectively in a team environment, contributing ideas and sharing insights to achieve common goals.
- Time management and organization: Managing multiple tasks and deadlines during my internship has helped me develop strong time management and organizational skills, enabling me to prioritize tasks and meet project milestones efficiently.

### **4.3 Most Challenging Task**

One of the most challenging tasks during my internship was developing a full-stack web application from scratch. This project required me to apply my knowledge of both front-end and back-end technologies to design and implement a feature-rich application that met the client's requirements. From defining the project scope and architecture to implementing user authentication, database integration, and user interface design, this task challenged me to think critically and creatively, solve technical problems, and work collaboratively with my team members to deliver a high-quality product within the specified timeline.

Overall, my internship experience at TEQUED LABS has been incredibly rewarding, providing me with hands-on learning opportunities, valuable mentorship, and the chance to apply my skills in a real-world setting. I'm grateful for the support and guidance I've received from my colleagues and mentors, and I'm excited to continue my journey in the field of software development with the knowledge and experience gained during my internship.

## CHAPTER 5

### Conclusion

In conclusion, my internship at Tequed Labs has been a transformative experience that has significantly shaped my understanding of web development. The hands-on learning environment allowed me to explore and implement HTML, CSS, JavaScript, and PHP in real-world projects, providing a practical foundation for my technical skills. Working alongside a seasoned team at Tequed Labs not only accelerated my learning curve but also instilled in me the importance of collaboration and innovative problem-solving in the field. This internship has not only deepened my knowledge of web development but has also cultivated a sense of confidence in applying these skills in a professional setting. I am grateful for the valuable lessons learned and the mentorship received during my time at Tequed Labs, making it a pivotal experience in my journey toward becoming a proficient web developer.