



# Web Development Internship

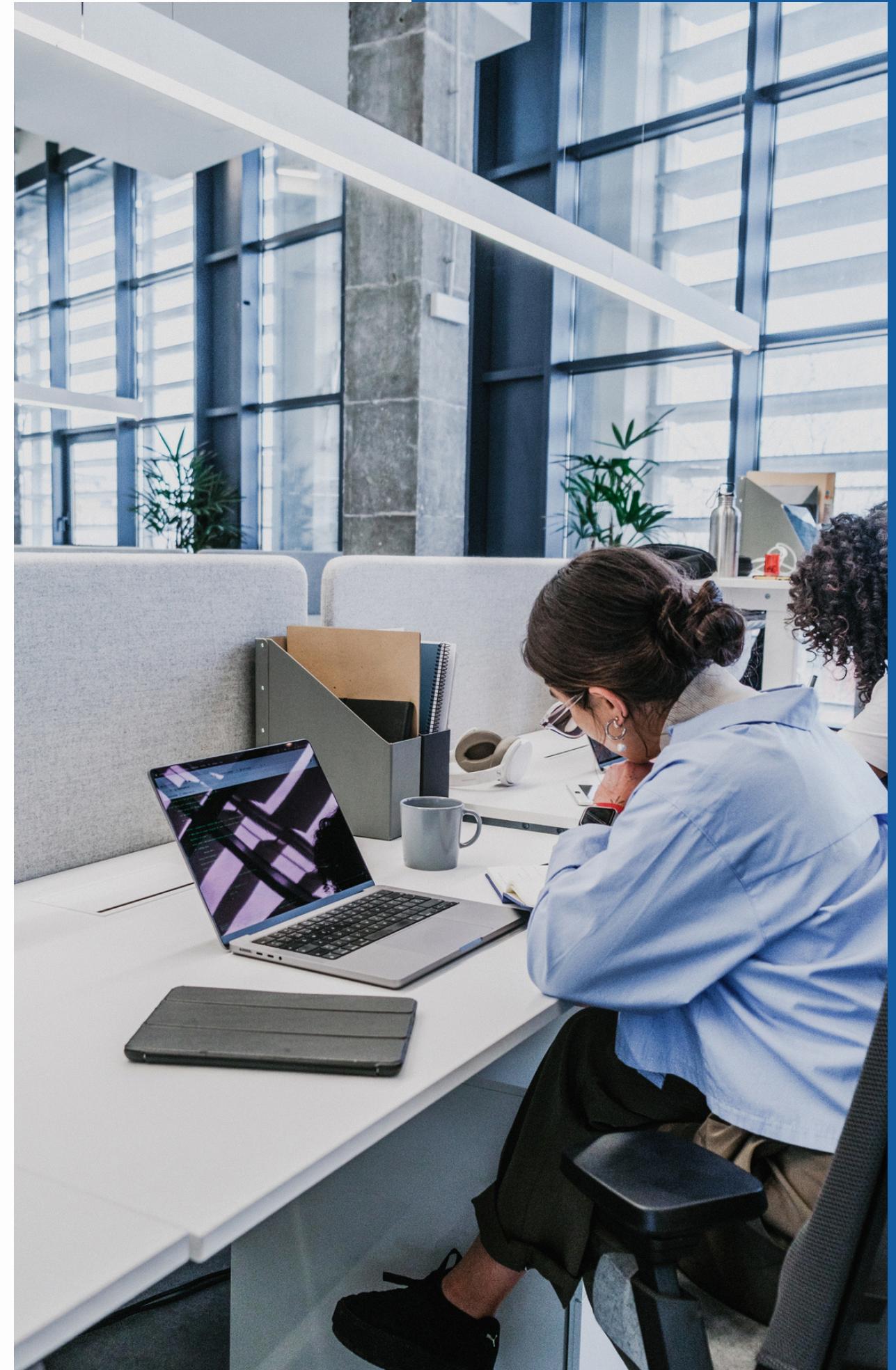
# Ariesian Tech

Discover---Create---Excel



# About us

Our company develops advanced AI and machine learning products to enhance safety by predicting, preventing, and mitigating risks across various domains. We focus on integrating AI/ML with real-time data, IoT devices, and automated systems to detect hazards, optimize response times, and provide actionable insights, empowering users to make informed decisions and stay protected.





# Instructions

- 1) For the internship ,you will need to complete your tasks ,convenience for successful completion of the internship.**
- 2) Create a new Github repository and name it Ariesian Tech and share the link of the Github repo in the submission form( it will be shared later through email .**
- 3) Update your LinkedIn profile to reflect your current internship position with Ariesian Tech. Ensure your profile is professional and accurately represents your role and responsibilities.**
- 4) Feel free to use online resources, such as Google Search and official documentation, to assist you in completing your tasks. Make sure to understand and properly implement any external resources you refer to.**

# Submissions

- 1) A task submission form will be shared via email once you've completed your selected tasks. Please ensure all tasks are finished before submission.**
- 2) Create a new GitHub repository titled "Ariesian Tech" and upload all relevant code and documentation for your tasks. Ensure that the repository is well-organized and includes a README file that explains your work.**
- 3) Post the demo video on your LinkedIn profile to demonstrate your work to peers and potential employers. This will also serve as proof of task completion. Also tag @AriesianTech.**
- 4) Please add #ariesiantech in each of your task video postings on LinkedIn, Additionally, you can also add hashtags such as #internship #webdevelopment. for more reach and visibility.**



# Task 1

**Submission date: 15th september  
from 1st sep to 15th sep  
Event Management Website**

## Tasks and Requirements:

- **Hero Section:**
  - Design a visually appealing banner with an introduction to the event management services.
  - Implement a backend system to dynamically fetch and display featured events on the banner.
- **Event Highlights:**
  - Display a few upcoming events with basic details like date, location, and a "Learn More" button.
  - Connect to a database to pull event data and update the homepage automatically as new events are added.
- **Navigation Bar and Footer:**
  - Create a navigation bar linking to main sections: Events, Gallery, Blog, Contact, and Registration.
  - Add a footer with quick links, social media icons, and a newsletter signup form.
  - Implement backend logic to store newsletter signups in the database.

## 1. Home Page



## **2. Events Page**

### **Tasks and Requirements:**

- **Event Calendar:**

**Design a calendar that displays upcoming events; users can click on a date to see more details.**

**Develop a backend system to pull event data from the database and populate the calendar.**

- **Event List:**

**Create a list of all upcoming events with options to filter by category or date.**

**Integrate with the backend to fetch event details dynamically, ensuring the list is always up-to-date.**

- **Event Details:**

**Each event should link to a detailed page with full information (e.g., date, location, schedule).**

**Implement ticket purchasing functionality where users can select ticket quantities and proceed to checkout.**

**Set up a backend system to handle ticket inventory, process payments, and manage user registrations.**

## **3. Registration Page**

### **Tasks and Requirements:**

- **Registration Form:**

**Design a user-friendly form with fields for name, email, phone number, and event selection.**

**Connect the form to the backend to store registration details in the database.**

- **Ticket Purchase Integration:**

**Allow users to select ticket quantity and process payments directly from the registration form.**

**Implement backend logic for payment processing, inventory management, and sending confirmation emails.**

## **4. Gallery Page**

### **Tasks and Requirements:**

#### **Photo/Video Gallery:**

- **Display images and videos from past events in an attractive grid or carousel layout.**
- **Integrate with the backend to automatically update the gallery as new media is added.**

#### **Media Upload and Management:**

- **Create an admin interface for uploading new photos and videos to the gallery.**
- **Develop backend functionality to store media files, categorize them, and manage their display on the front-end.**

## **5. Blog Page**

### **Tasks and Requirements:**

#### **Blog Listing:**

- **Display a list of blog posts with titles, excerpts, and a "Read More" link.**
- **Integrate with a backend system to fetch and display blog posts dynamically from a database.**

#### **Blog Detail View:**

- **Each post should have its own page with the full content, related posts, and a comment section.**
- **Implement backend logic to manage blog content, including creating, editing, and deleting posts.**
- **Set up a comment system with backend moderation tools.**

## **6. Contact Page**

### **Tasks and Requirements:**

#### **Contact Form:**

- Create a form with fields for name, email, phone number, and message.**
- Integrate with the backend to store form submissions in a database and send email notifications.**

#### **Contact Information and Map:**

- Display the company's contact information (address, phone number, email) and embed a Google Map.**
- Use backend logic to update contact details and manage map integration.**

## **7. Admin Dashboard (For Internal Use)**

### **Tasks and Requirements:**

#### **Event Management:**

- Create an interface for admins to add, update, or delete events.**
- Develop backend functionality to manage event data, including ticket inventory and registration lists.**

#### **Media Management:**

- Implement tools for uploading and organizing photos and videos for the gallery page.**

#### **Blog Management:**

- Provide an interface for creating and managing blog posts, including scheduling and editing.**

#### **User Management:**

- Set up a system for managing user roles (admin, editor) and access to various parts of the website.**

# Task 2

Submission data: 1st october  
from 16th sep to 1st october

Website of an organization/company

Header Section:

Front-End Tasks:

Navigation Bar:

- Implement a sticky navigation bar with the company logo, links to other pages, and a dropdown menu for services.
- Add a search icon that expands into a search bar on click.

Back-End Tasks:

Dynamic Navigation:

- Develop an endpoint to dynamically fetch the navigation menu items and links from the database.
- Set up the schema for storing navigation items.

Main Banner:

Front-End Tasks:

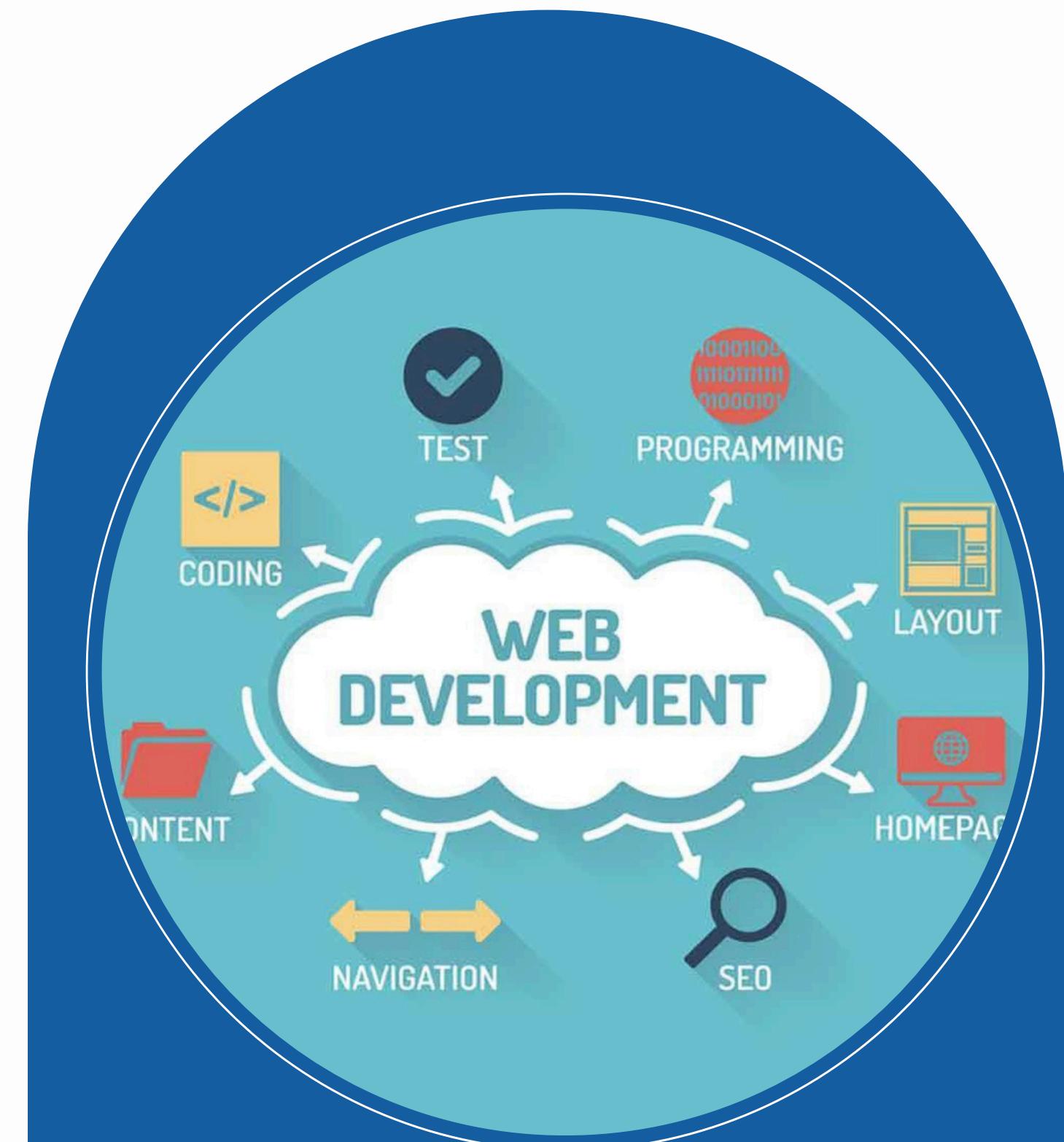
Hero Banner:

- Design a banner with a high-quality background video, a bold tagline, and two call-to-action buttons ("Learn More" and "Get Started").
- Implement a smooth scroll effect when clicking the buttons.

Back-End Tasks:

Banner Content Management:

- Develop an API to fetch banner text, background media, and call-to-action details from the database.
- Set up a schema to store banner content and associated media files.



## About Section

- **Front-End Tasks:**
  - **Company Overview:** Add a detailed company introduction with an emphasis on the company's unique values.
  - **Our Journey:** Implement a timeline feature to display the company's history, key milestones, and growth.
- **Back-End Tasks:**
  - **API Endpoints:** Develop an endpoint to fetch company information, milestones, and values.
  - **Database:** Set up schemas for storing company history, milestones, and values.

## Services Overview

### Front-End Tasks:

- **Service Cards:** Create dynamic service cards with icons, summaries, and expandable sections for more details.
- **Interactive Elements:** Implement hover effects and animations on service cards for an engaging user experience.

### Back-End Tasks:

- **API Endpoints:** Develop endpoints to fetch service summaries, icons, and detailed descriptions.
- **Database:** Set up schemas for storing service information and related media.

## Client Testimonials

### Front-End Tasks:

- **Slider Testimonials:** Display client testimonials in a sliding carousel with client images, quotes, and star ratings.
- **Video Testimonials:** Include video testimonials from clients.

### Back-End Tasks:

- **API Endpoints:** Develop an endpoint to fetch client testimonials and videos.
- **Database:** Set up schemas for storing testimonials, client information, and related media.

## Latest Projects/Case Studies

### Front-End Tasks:

- **Project Grid:** Showcase recent projects in a grid layout with images, brief descriptions, and links to full case studies.
- **Filter Options:** Provide filter options for users to sort projects by category, date, or client.

### Back-End Tasks:

- **API Endpoints:** Develop endpoints to fetch project details, images, and case study links.
- **Database:** Set up schemas for storing project information, categories, and related media.

## Footer

### Front-End Tasks:

- **Contact and Social Media:** Include detailed contact information, social media icons with hover effects, and a newsletter signup form.
- **Site Map:** Add a site map for easy navigation.

### Back-End Tasks:

- **API Endpoints:** Develop endpoints to fetch contact information, social media links, and handle newsletter signups.
- **Database:** Set up schemas for storing contact details, social media links, and newsletter subscribers.

## **Blog Section:**

### **Front-End:**

- **Latest Articles:** Display a section for the latest blog articles with featured images, titles, and excerpts.
- **Categories:** Allow users to browse articles by category.

### **Back-End:**

- **API Endpoint:** Create endpoints to fetch blog articles, categories, and featured content.
- **Database:** Set up schemas for storing blog posts, categories, and tags.

## **Contact Us Page:**

### **Front-End:**

- **Contact Form:** Create a form with fields for name, email, phone number, preferred contact method, and message.
- **Interactive Map:** Embed an interactive Google Map with markers for multiple office locations.
- **Live Chat:** Include a live chat widget for immediate support.

### **Back-End:**

- **API Endpoint:** Develop an endpoint to handle form submissions and live chat messages.
- **Database:** Set up a schema for storing contact form submissions and chat logs.
- **Email and Notifications:** Configure email notifications and admin dashboard alerts for new contact form submissions and chat inquiries.

## **Why Choose Us Page:**

### **Front-End:**

- **Intro Section:** Provide a comprehensive overview of the company's core values, customer-centric approach, and key achievements.
- **Differentiators:** Highlight key differentiators with detailed descriptions and visual icons.
- **Success Stories:** Include success stories with images, videos, and client testimonials.
- **Recognition:** List awards and recognitions in a visually appealing layout with images and links.

### **Back-End:**

- **API Endpoint:** Develop endpoints to fetch company strengths, differentiators, success stories, and recognitions.
- **Database:** Set up schemas for storing company strengths, differentiators, success stories, and awards.

## **Services Details Page:**

### **Front-End:**

- **Service Sections:** Create detailed sections for each service with descriptions, client testimonials, and related case studies.
- **Service Comparison:** Include a comparison table for users to easily understand differences between services.
- **Interactive Demos:** Provide interactive demos or sample projects related to each service.

### **Back-End:**

- **API Endpoint:** Develop endpoints to fetch detailed service descriptions, comparisons, and related case studies.
- **Database:** Set up schemas for storing service details, comparisons, and project demos.

## **Hiring Vacancy Page:**

### **Front-End:**

- **Job Listings:** Display current job openings with titles, descriptions, requirements, and application deadlines.
- **Application Form:** Provide a form for direct online applications, with options to upload resumes and cover letters.
- **Employee Stories:** Share stories from current employees about their experiences and growth within the company.

### **Back-End:**

- **API Endpoint:** Develop endpoints to fetch job listings, descriptions, requirements, and handle job applications.
- **Database:** Set up schemas for storing job listings, application forms, and employee stories.
- **Resume Management:** Implement a secure system for storing and managing uploaded resumes.

## **Login Page:**

### **Front-End:**

- **Login Form:** Create a form with fields for email, password, and a "Remember Me" option.
- **Two-Factor Authentication:** Include an option for two-factor authentication during login.
- **Password Recovery:** Provide links for password recovery and account creation.

### **Back-End:**

- **Authentication:** Implement secure user authentication using JWT with two-factor authentication support.
- **API Endpoint:** Develop endpoints for user login, password recovery, and account management.

## **Logout Page:**

### **Front-End:**

- **Logout Confirmation:** Display a personalized confirmation message with a prompt to return to the login page.
- **Session Timeout Alert:** Include an alert for automatic logout after a period of inactivity.

### **Back-End:**

- **Session Management:** Ensure proper handling of user sessions, automatic logout, and token invalidation upon logout.
- **User Activity Logs:** Implement logging of user activities related to session management.

## **INTERNSHIP AIM**

### **Skill-building and Improvement**

The goal of this internship is to gain knowledge and develop skills. Our internship aims to support your learning and development. You have the freedom to seek guidance as you see fit. The tasks provided might seem either straightforward or challenging. We expect you to approach each task with a high level of professionalism and to dedicate the attention each one deserves.



# THANK YOU!