COLLEGE CODE:9111

COLLEGE NAME : SRM Madurai College For Engineering and Technology

DEPARTMENT : B.Tech Information Technology

NAME	STUDENT NM-ID
Irudhaya Albert	16C4819348EEF23BC7949300C8AACEBC
Nishanthan S	6FD89376531AD80AC6F4FF5782127907
Thameej Ahamed	ACB9D50E54865509DF44C356F445FCFD
Pradeep S	1636178DC69762CAF364A6BC4F0A1684
Vikram KV	E6E2E5AB586408AB3A741885965448AF

DATE:15-09-2025

Completed the project named as

Phase _1_TECHNOLOGY PROJECT

NAME : PORTFOLIO WEBSITE

SUBMITTED BY,

Irudhaya Albert
Nishanthan S
Thameej Ahamed
Pradeep S
Vikram KV

1. Problem Statement

In the rapidly evolving digital era, **personal branding** has become just as important as academic qualifications and technical skills. A strong online presence allows individuals to stand out from the crowd, whether they are applying for jobs, internships, freelance opportunities, or higher studies. Traditionally, resumes and CVs have served as the primary medium for showcasing qualifications and achievements. While useful, these documents have significant limitations:

1. **Static in Nature:** Resumes are static and must be manually updated each time new skills or projects are acquired. This results in frequent version management problems.
2. **Lack of Interactivity:** They cannot demonstrate interactive elements such as live project demos, videos, or direct links to GitHub repositories.
3. **Not Universally Accessible:** A resume must be shared directly through email or hard copy. It cannot be globally accessed unless uploaded to third-party sites.
4. **Poor Recruiter Engagement:** With limited space, resumes often fail to highlight creativity or provide a complete picture of a candidate's abilities.

In contrast, **portfolio websites** serve as **dynamic resumes** that can be accessed worldwide with a single link. They combine design, interactivity, and real-time updates to create an engaging experience for viewers.

Yet, current solutions such as free templates and static site generators still leave gaps:

- Many are **static HTML/CSS templates**, meaning a candidate must manually change the source code to update a project.
- Free templates often lack **scalability**, **personalization**, and modern UI/UX practices.
- Without a backend, there is no support for **dynamic CRUD operations**, making content management difficult.
- Mobile responsiveness is sometimes missing, leading to poor experience on smartphones, even though most recruiters browse on mobile.

Hence, there is a **clear gap** between what students/professionals need and what is currently available.

Our project aims to bridge this gap by developing a **Portfolio Website** using:

- **Frontend (React / HTML, CSS, JavaScript)** for responsive and interactive user interface.
- **Backend (Node.js + Express.js)** for handling API requests and dynamic updates.
- **Database (MongoDB)** for storing projects, skills, and recruiter messages securely.

This project provides a **personalized, scalable, and dynamic solution** for professional self-presentation. It not only addresses technical challenges but also aligns with **placement requirements** and **industry expectations**.

👥 2. Users and Stakeholders (Expanded)

The Portfolio Website will have different categories of users and stakeholders, each with their unique goals and interactions.

Users

1. **Students / Professionals (Primary Users):**
    - Build a professional identity online.
    - Showcase academic qualifications, certifications, and achievements.
    - Display technical skills and completed projects.
    - Keep their portfolio updated dynamically without modifying raw code.

    Example: A B.Tech IT student applying for campus placements can add their newly completed project in Machine Learning directly through the backend API without editing HTML.

2. **Recruiters / Employers (Secondary Users):**
    - Quickly view portfolios during the hiring process.
    - Evaluate candidate profiles based on skills, projects, and achievements.
    - Contact students directly through the built-in contact form.

    Example: A recruiter from TCS visiting the college for placements can browse multiple student portfolios and shortlist candidates faster than going through PDF resumes.

3. **Faculty / Mentors:**
    - Use the website as part of academic evaluation.
    - Encourage students to maintain a professional digital identity.
    - Provide structured feedback on design, content, and usability.

    Example: A faculty mentor in charge of projects can evaluate not only the technical functionality but also the student's ability to present themselves professionally.

4. **Visitors / Peers:**
    - Explore the student's portfolio to get inspiration.
    - Collaborate for hackathons, startups, or academic projects.
    - Share the portfolio link as a reference in student networks.

Stakeholders

- **Project Owner (Student):**
  Responsible for creation, updates, and branding of the portfolio.
- **Development Team:**
  Handles frontend, backend, and database integration (in case of team project).

- **Faculty / College Evaluators:**
  Assess whether the student is placement-ready and capable of handling industry-relevant tools.
- **Recruiters / Employers:**
  Final consumers of the portfolio, whose satisfaction decides the effectiveness of the platform.
- **UI/UX Designers:**
  Ensure professional, recruiter-friendly layouts.
- **Mentor / Project Guide:**
  Provides supervision and technical direction.

👥 3. User Stories (Detailed)

The Portfolio Website is driven by the following **expanded user stories**:

- **As a Student/Professional**, I want to create a **centralized digital hub** where all my projects, skills, certifications, and achievements are stored, so I can present myself professionally with just a single link.
- **As a Recruiter**, I want to **browse the portfolio quickly** without login requirements and immediately identify whether the student matches the job requirements, so I can make faster hiring decisions.
- **As a Faculty Mentor**, I want to **assess the student's personal branding skills**, technical presentation, and project quality by reviewing the portfolio, so I can guide them towards better employability.
- **As a Visitor/Peer**, I want to **view project details with GitHub links**, so I can understand how the student developed them and maybe collaborate on similar ideas.

These stories ensure that the portfolio serves **all target users**, from students building careers to recruiters hiring talent.

🔍 4. Minimum Viable Product (MVP) Features

Core Frontend Features

1. **Home Page:**
   - Profile photo, tagline, navigation bar.
   - Quick introduction with role (e.g., "B.Tech IT Student | Web Developer | Cloud Enthusiast").
2. **About Page:**
   - Personal introduction, education details, achievements.
   - Skillset presented with progress bars or cards.
3. **Projects Page:**
   - Showcase academic, personal, or internship projects.
   - Each project includes description, technologies used, GitHub/demo links.
4. **Contact Page:**

- Form with fields: Name, Email, Message.
- Submissions stored in backend for future reference.

Backend Features

- **Project Management APIs:** CRUD operations for adding/updating projects.
- **Skills Management APIs:** CRUD for managing technical skills.
- **Contact Form API:** Stores recruiter messages securely in database.
- **JSON Output:** Ensures smooth communication with frontend.

Additional Features

- Responsive design for desktop, tablet, and mobile.
- Clean UI/UX designed for recruiters.
- Page load under 3 seconds.
- Data validation and secure storage.

Future Enhancements

- Authentication system for portfolio owner.
- Integration with LinkedIn/GitHub APIs.
- Blog or "Articles" section.
- Dark mode / custom themes.
- Analytics dashboard to track portfolio visits.

👪 5. Wireframes / API Endpoint List

Wireframes (High-Level Layouts):

- **Home Page:** Profile image, name, tagline, navigation menu.
- **About Page:** Timeline for education, cards for skills, achievements.
- **Projects Page:** Grid layout of projects with clickable links.
- **Contact Page:** Simple, recruiter-friendly form.

API Endpoints

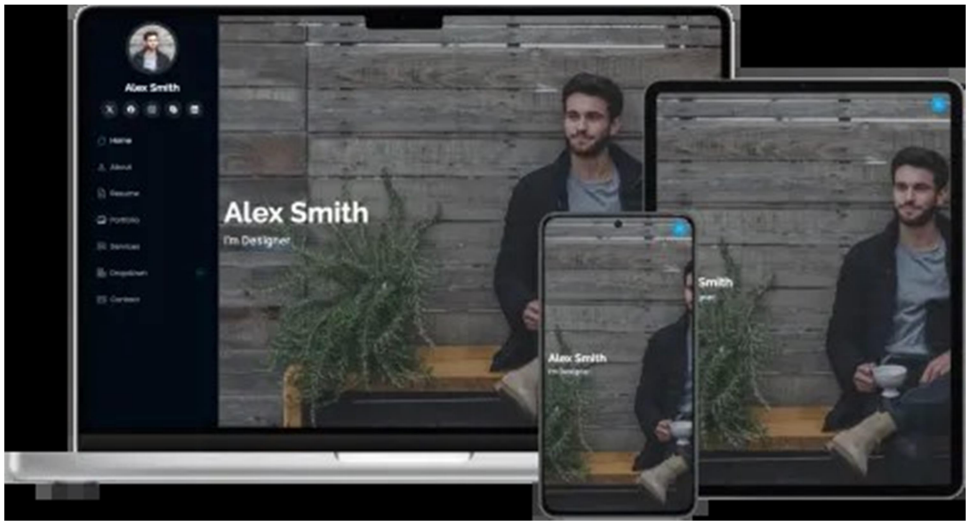
- **Projects:**
  - `GET /api/projects` → Retrieve all projects.
  - `POST /api/projects` → Add new project.
  - `PUT /api/projects/:id` → Update project.
  - `DELETE /api/projects/:id` → Delete project.
- **Skills:**
  - `GET /api/skills` → Fetch all skills.
  - `POST /api/skills` → Add new skill.
- **Contact:**

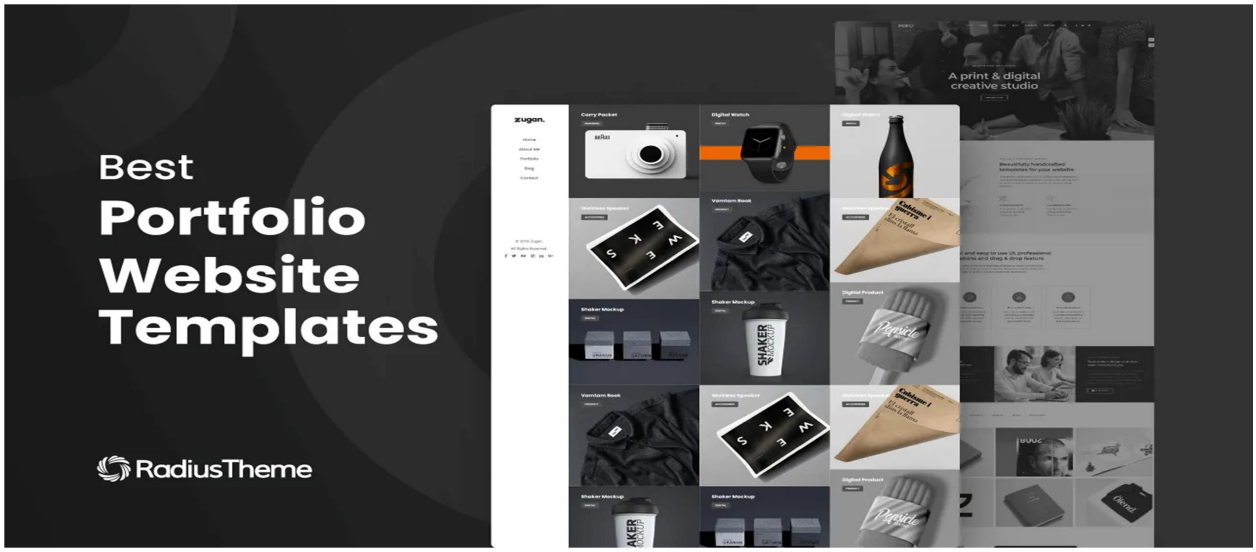
- `POST /api/contact` → Store recruiter's message.

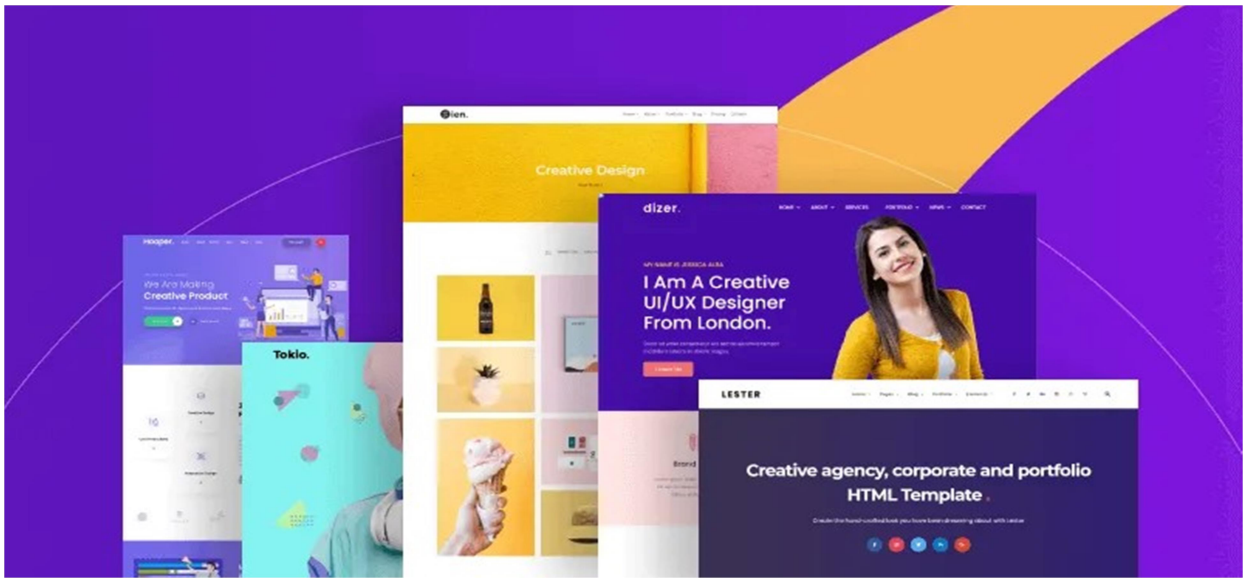
6. Acceptance Criteria

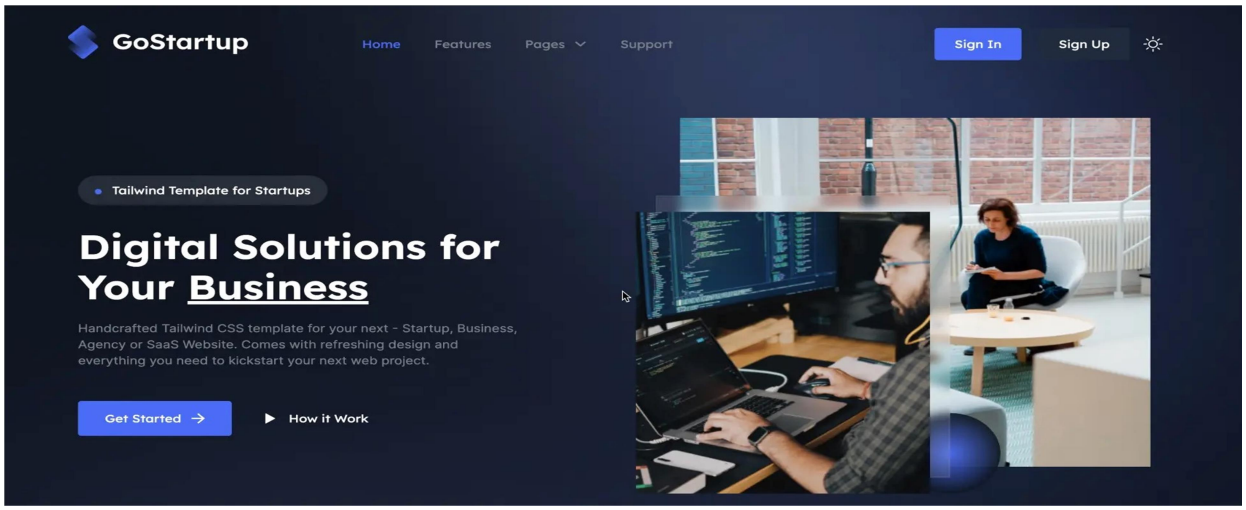
To ensure the project meets user expectations, the following **acceptance criteria** are defined:

Criterion	Description	Metric / Threshold	Verification Method
Responsiveness	Website works across desktop, tablet, and mobile.	No broken layout in multiple screen sizes	Manual UI testing
Page Load Speed	Quick loading of content.	< 3 seconds	Google Lighthouse testing
API Response Time	Backend APIs must return data efficiently.	< 500ms	Postman / Load testing
CRUD Operations	Add/update/delete project and skill entries.	100% correct in test cases	Unit testing + Integration testing
Contact Form	Submissions are validated and stored correctly.	Rejects invalid/empty inputs	Automated testing + manual checks
Data Accuracy	Content displayed must match database values.	100% match with MongoDB entries	Data validation checks
Security	Protects user data and recruiter info.	HTTPS, Input validation, DB sanitization	Security audit




Best
Portfolio
Website
Templates
RadiusTheme
A print & digital creative studio


Creative Design
dizer.
I Am A Creative UI/UX Designer From London.
LESTER
Creative agency, corporate and portfolio HTML Template.
Tokio.


GoStartup
Home
Features
Pages
Support
Sign In
Sign Up
Tailwind Template for Startups
Digital Solutions for Your Business
Handcrafted Tailwind CSS template for your next - Startup, Business, Agency or SaaS Website. Comes with refreshing design and everything you need to kickstart your next web project.
Get Started
How it Work