



Portfolio Website Task - 4



Portfolio Website

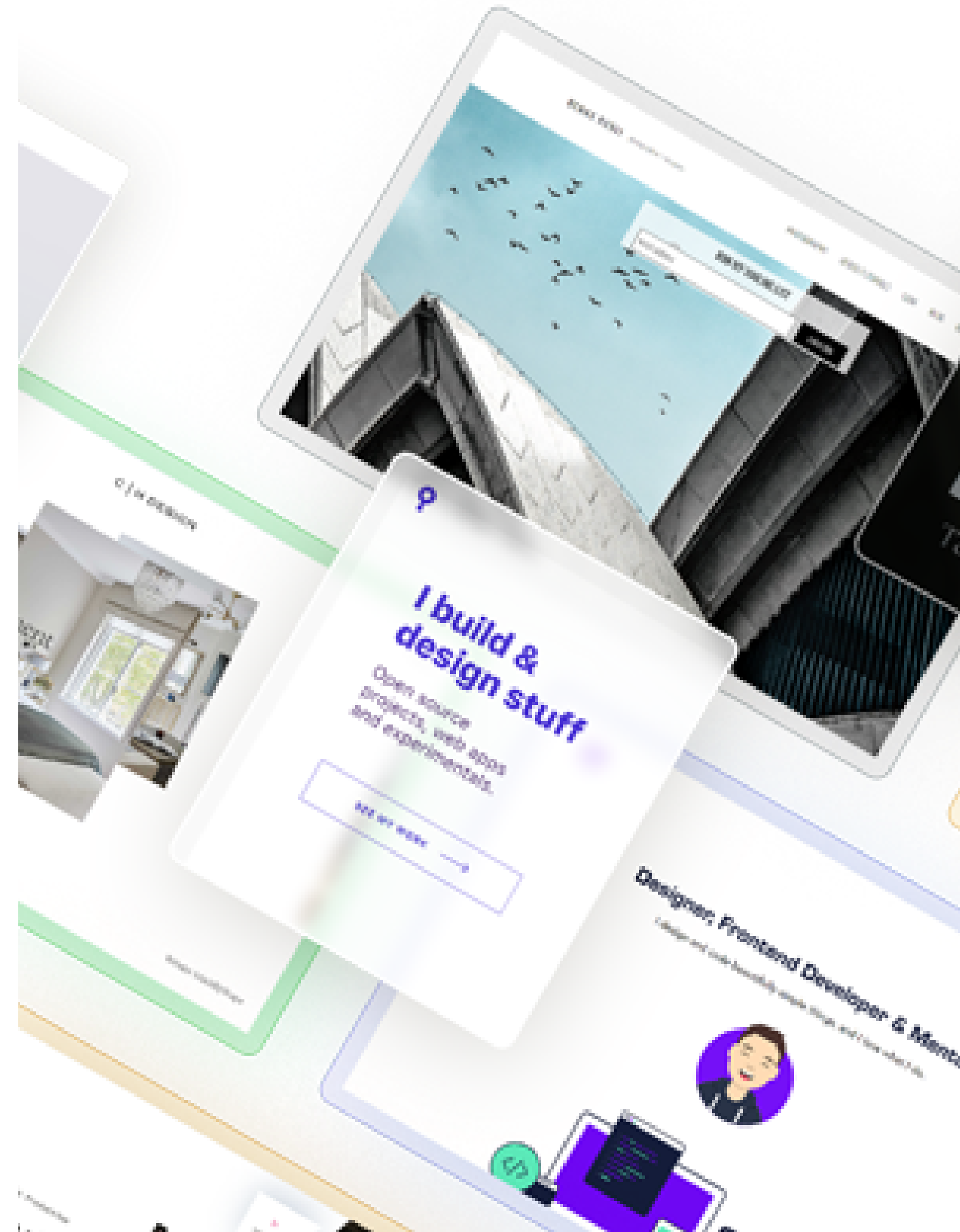
This project aims to create a digital display of your work, like an online exhibition. It lets visitors explore your projects, get in touch using a contact form, and stay updated through a blog. It's your creative space on the web, presenting your skills and experiences.

LMS Username	Name	Batch
au910020104026	MOHANAPRIYA J	CC2
au910020104029	PADMAKSHI V	CC2
au910020104035	PRAVEENA R	CC2



Introduction

This project entails developing a personalized portfolio website to showcase professional skills and achievements. The website features a clean and responsive design, encompassing sections like a welcoming homepage, a comprehensive portfolio display, detailed background information. Additionally, it's optimized for search engines, seamlessly integrates social media links, and can optionally include testimonials. Regular updates and backups are integral to maintaining the website's relevance and security. The primary aim is to create a compelling online presence, catering to potential employers, clients, and collaborators.



Task 4

Dynamic Frontend

1. Identify Endpoint:

- Determine the specific API endpoint for data retrieval or interaction.

2. API Documentation:

- Review the API documentation for endpoint details and authentication requirements.

3. Base URL:

- Find the API's base URL, typically in the format `https://api.example.com`.

4. Integration:

- Integrate the API into your website code, handling headers and authentication as needed.

5. Testing and Documentation:

- Thoroughly test the integration, handling responses appropriately, and document your code for future reference.

EmailJS API:

- EmailJS is a service enabling email sending directly from client-side JavaScript in web applications. Developers can integrate this service into their apps without the need for server-side code.
- The API likely involves HTTP requests to specific endpoints for email composition and sending.
- Users typically authenticate via API keys obtained after signing up for an EmailJS account.
- The service may support the creation and configuration of email templates for streamlined email composition and sending.

Email Services

EmailJS

QUOTA REMAINING: 196 EMAILS

Welcome, Portfolio website

Docs

Support

Sign Out

Email Services

Email Templates

Contacts

Email History

Events

Statistics


Team Members

Account

-


Email Services


Add New Service


 Gmail

Service ID: **service_rbqyq44**

DEFAULT







Email Templates

EmailJS

QUOTA REMAINING: 196 EMAILS

Welcome, Portfolio website

Docs

Support

Sign Out

Email Services

Email Templates

Contacts

Email History

Events

Statistics

Team Members

Account

Email Templates

Create New Template

My Default Template

Template ID: `template_uoj32b5`

Public key

EmailJS

QUOTA REMAINING: 196 EMAILS

Welcome, Portfolio website

Docs

Support

Sign Out

Email Services

Email Templates

Contacts

Email History

Events

Statistics

Team Members

Account

Account

General

Subscription

Invoices

Security

API keys

Public Key

ZPWQEpxWEkQqn09Sh

Private Key

SMFk.....

Refresh Keys

Notifications

Evaluation Metric

1. User Engagement:

- Evaluate metrics like page views, time spent on the site, and bounce rate to understand user engagement and interest.

2. Portfolio Interaction:

- Monitor interactions within the portfolio section, including project clicks and user actions, to assess the effectiveness of showcasing your work.

3. Performance and Responsiveness:

- Measure website loading times, responsiveness across devices, and error rates to ensure a seamless user experience and address any technical issues promptly.

Learning Outcomes

Develop technical skills using HTML, CSS, and JavaScript for a dynamic portfolio website. Apply UX/UI principles for an engaging user experience. Integrate APIs, like EmailJS for client-side email functionality, enhancing website interactivity. Implement SEO practices and analytics, ensuring a strategic online presence and continuous improvement. Showcase adaptability through troubleshooting and leveraging third-party services for added functionality.

Step-Wise Description

1. Design and Build Core Pages:

- Design a user-friendly interface using UX/UI principles and build core pages (Home, Portfolio, About Me, Resume, Contact).

2. API Integration:

- Identify and integrate APIs (e.g., EmailJS for contact forms, social media API for live feeds) for dynamic content and enhanced functionality.

3. Testing and Optimization:

- Thoroughly test website functionalities, ensure responsiveness, and optimize performance for a seamless user experience.

4. Deploy and Monitor:

- Deploy the website using a hosting provider, implement basic security measures, and continuously monitor analytics for insights and improvements

Summary

Create a dynamic portfolio website using HTML, CSS, and JavaScript. Integrate APIs, like EmailJS for contact forms, ensuring dynamic content. Optimize for performance, deploy securely, and monitor analytics for continuous improvement.

GitHub



[View my code on GitHub](#)

Thank you!

