

MASHRAFIE RAHIM SHEIKH

✉ rahimsheikh990@gmail.com | [in Mashrafie Rahim Sheikh](#) |
🌐 portfolio-m3tal10.vercel.app

EDUCATION

University of The People | *BSc. in Computer Science*

California, USA

2022-2025

- Cumulative GPA: 3.98 /4.00.
- Named to the President's list 9 times.
- Named to the Dean's list 2 times.
- Awarded Sustainability Badge for contributing to Uopeople's global sustainability program.
- Selected as a UoPeople Ambassador to support initiatives aimed at innovating and expanding access to quality education.

SKILLS

Programming languages: JavaScript, Python, Java, C, C++.

Frontend Development: ReactJS, NextJS, Tailwind CSS.

Backend Development: Node.js, Express.js, Django.

Database Management: MongoDB, PostgreSQL.

DevOps & Cloud Infrastructure: Docker, AWS, Nginx.

Version Control & Collaboration: Git, GitHub.

Others: Data Structures & Algorithms, Problem Solving, Debugging, Code Refactoring, Analytical ability.

PROJECTS

Kolibri (Open Source) | [GitHub Link](#)

- Implemented lazy loading functionality to the images, significantly improving the Kolibri app's performance on small screen-sized devices.
- Fixed a constant 404 API response by handling all cases of 400 responses and added corresponding tests for integration testing.

PERSONAL PROJECTS

Devsearch | [Live Link](#) | [GitHub Link](#)

- Django for building the backend server to handle client requests.
- Used PostgreSQL as the database to store the application data.
- Used Django channels to refactor a simple messaging feature to a real-time chat feature.
- Used docker and docker-compose to containerize the application for scalability.
- Used AWS EC2 service to deploy the application.
- Used Nginx to handle all network traffic inside the AWS EC2 instance which resulted in a 50% better traffic handling and load balancing.

Natours | [Live Link](#) | [GitHub Link](#) | [Documentation](#)

- RESTful architecture for stateless server communication. (Please check the documentation)
- Node.JS and Express.JS for building the backend server to handle requests.
- Used MongoDB as the database and Mongoose on top to model and validate the data.

- Used PUG template engine to render the frontend website on the server.
- Used Nodemailer with Brevo to send emails after successful signup and to reset password.
- Integrated payment gateway with Stripe to handle payment and Stripe webhook for secure transactions.

The wild oasis | [Live Link](#) | [GitHub Link](#)

- Used Supabase to handle database and user authentication for handling the server system to make the production process faster.
- Used React to build the Front-end.
- Used React query for managing remote states to make a robust state management pattern.
- Used CSS styled-components for styling to make a scalable and readable app.
- Used Recharts for generating Pie and Grid charts to show data-rich and visually appealing statistics.

WorldWise | [Live Link](#) | [GitHub Link](#)

- Node.JS and Express.JS to build the backend server to handle requests. Mainly authentication and authorization.
- Used ReactJS to build the front end.
- Used React leaflet to integrate a map in the app for a better user experience.
- Used React ContextAPI for state management.
- Used React router for controlled routing.
- Used CSS modules to design the app.

Fast Pizza Co. | [Live Link](#) | [GitHub Link](#)

- Used React to build the front-end website.
- Used Redux Toolkit for state management to make a robust, readable, and scalable application.
- Used Tailwind CSS for designing the app to make the app responsive and usable for every device.

CERTIFICATIONS

- **HTML, CSS,& JavaScript Essentials** | [Verification Link](#)
- **Git & GitHub Essentials** | [Verification Link](#)
- **Developing Front-End Apps with React** | [Verification Link](#)