

Prasoon PR

Kochi | prasoonpr2003@gmail.com | 0977 858 28 49 | www.prasoonpr.tech

linkedin.com/in/prasoon-pr-099157295 | github.com/prasoonpr

About Me

Self-taught MERN Stack Developer passionate about building scalable web applications. Skilled in crafting seamless user experiences, optimizing performance, and developing robust backend systems. Committed to writing clean, maintainable code while continuously learning and adapting to new challenges.

Education

BROTOTYPE, MERN Full Stack Development April 2024– March 2025

MG-University, Bachelor of Computer Application Dec 2021– April-2024

Technologies

Expertise: JavaScript, TypeScript, Node.js, Express.js, React.js, Next.js, NestJS, PostgreSQL, MongoDB, RESTful APIs, Redux, MVC Architecture, SOLID Principles, DSA (Data Structures and Algorithms), Docker, Axios

Also Used: JWT, Postman, Tailwind CSS, Bootstrap, Mantine, React Hook Form, Figma, Git

Integrations: Razorpay, Cloudinary, AWS, Nginx, Firebase, OAuth2.0, Nodemailer, Multer, Versal

Experience

Freelance Backend Developer, CodeEdex March 2024 – Present

Contributing to the development of a scalable e-commerce platform by designing RESTful APIs, managing user authentication, integrating payment gateways, handling product and order management, and implementing efficient database schemas using MongoDB and Node.js with Express.

Professional Development Program, Packapeer Academy April 2024 – March 2025

Engaged in a comprehensive self-learning program featuring hands-on projects, personalized mentorship from industry experts, weekly performance reviews, and real-world applications, fostering innovative problem-solving, technical proficiency, and a commitment to continuous improvement.

Projects

Freelance Backend Developer – CodeEdex (Ongoing) github.com/prasoonpr

- Collaborating with CodeEdex since March 2025 as part of a freelance team to develop the backend for a feature-rich e-commerce platform.
- Developed and integrated core modules including order management, coin-based rewards system, and secure admin operations.
- Utilized AWS S3 for scalable media handling and implemented OneSignal for real-time push notifications.
- Ensured modular architecture and seamless integration with frontend and DevOps pipelines.

Technologies Used

- **Backend:** Node.js, Express.js, MongoDB, JWT, AWS S3, Multer, OneSignal
- **Tools:** Postman, GitHub

Feature-Rich E-Commerce Website for Stones

[Live Link](#)

- Developed a scalable full-stack e-commerce platform with advanced filtering, authentication, and payment features.

- Built a secure backend using Node.js, Express.js, MongoDB, JWT, Google OAuth, and MVC architecture.
- Designed a responsive UI with React.js, Redux Toolkit, RTK Query, and Tailwind CSS.
- Integrated Razorpay and PayPal for payments, OTP verification via Nodemailer, and image storage via Cloudinary.
- Deployed using Docker, Nginx (backend), and Vercel (frontend) for a performant and scalable infrastructure.

Technologies Used

- **Backend:** Node.js, Express.js, MongoDB, JWT, Bcrypt, Multer
 - **Frontend:** React.js, Redux Toolkit, RTK Query, Vite, React-Toastify
 - **DevOps:** Docker, Nginx, Vercel, Git, Postman
 - **Integrations:** Razorpay, PayPal, OAuth2.0, Cloudinary, Nodemailer
-

Minimal Job Portal Web App

[Live Link](#)

- Built a clean and functional job portal with an admin dashboard for creating and listing job posts.
- Users can browse available jobs with filtering options and apply directly from the frontend.
- Developed using NestJS (backend) and Next.js (frontend) with PostgreSQL for data persistence.
- Used React Hook Form and Mantine for elegant UI and form handling.

Technologies Used

- **Backend:** NestJS, PostgreSQL, TypeORM
 - **Frontend:** Next.js, Mantine UI, React Hook Form
 - **DevOps:** Railway, Vercel, Git, Postman
-

Task Management Application

[Live Link](#)

- **Efficient Task Management:** Developed a user-friendly to-do app for creating, updating, and deleting tasks with priority levels and deadlines.
- **State Persistence:** Stored tasks in localStorage to maintain data even after page refresh.
- **Optimized Frontend Performance:** Built with React.js, using Redux Toolkit for state management and React-Toastify for notifications.
- **Task Categorization and Filtering:** Implemented task categories, search, and filtering options for better organization.

Technologies Used

- **Frontend:** React.js, Redux Toolkit, Vite, React-Toastify
- **DevOps and Tools:** Git, GitHub, ESLint, Prettier