



Rajesh Paul

Full Stack Web Developer

CONTACT

Rp0399451@gmail.com

+91 8178766470

North-West, Delhi

 [LinkedIn](#)

 [Github](#)

PROFESSIONAL SUMMARY

Responsible, self-disciplined, motivated and hard-working full-stack web developer. Intrapreneur, always curious about learning and implementing with a keen interest in software development. Aspires to enhance individual skills and knowledge while working for an organisation.

EDUCATION

Full-Stack Web Development

Masai School, Bangalore

July 2021 - Present

EXPERIENCE

Instructional associate at Masai School

Duration : 8 months

SKILLS AND FRAMEWORKS

HTML | CSS | Javascript | Java | React |
Redux | ExpressJs | MongoDB

SOFT SKILLS

Team Player | Professionalism | Work-
ethics | Self-motivation

INTERESTS

Chess | Current affairs | Technology

B.COM(HONS)

School of Open Learning, New Delhi

June 2019 - Present

PROJECTS

SMALLCASE

Smallcase is a web application for investing in different finance products/schemes.

Features:

- User authentication
- Search with debouncing
- Api for different tasks
- Products sorting by Price, Strategy and level of risk

Tech Stack: EJS | Tailwind | Auth-2 | Express | MongoDB | Redis | Razorpay

Areas of responsibility:

- Designed backend system including Models, Controllers.
- Implemented sorting, pagination through backend and created UI for that as well which is available in smallcases page.
- Created optimised blog page where user can search for the news with the debouncing functionality used news-api for that.
- Incorporated razorpay for the payment mechanism.
- Created watchlist & investments pages, if the payment is done it will go in investments which will be specific for every user as stored in database similarly implemented add to watchlist feature as well.

A collaborative project built by a team of 5 executed in 10 days.

Dineout

Find best restaurants offering discounts on food & drinks, check out menu, reviews and also book a table through dineout for free.

Features:

- Login & Signup with two tokens access & refresh for security and storing them in cookie and accepting on every request to check auth.
- Role based authorisation : Two roles are there 'Restaurant' & 'User' user can't add restaurant and restaurant can't find other restaurants.
- Parallel sorting through backend where user can parallelly sort and apply various filters parallelly, used aggregation pipeline for that.
- Implemented razorpay for checkout.

Tech Stack: React | Redux | Express | MongoDB | Meterial-UI | Cookie-parser | JWT |

Areas of Responsibility:

- Created complete backend singlehandedly.
- Designed two roles 'User' & 'Restaurant'.
- Created authentication system in backend and created frontend for it also and connecting them using cookie (access & refresh tokens).
- Implemented Razorpay.
- Implemented parallel sorting with backend with the concept of aggregation.
- Designed UI for homepage

A collaborative project built by a team of 5 executed in 5 days.