



# MUSAFFARUL ISLAM

## FULL-STACK DEVELOPER

+91 9562886328

dev.musaffarulislam@gmail.com

musaffarulislam

musaffarulislam

portfolio

Kannur, Kerala

## EDUCATION

### BSc Computer Science, Mathematics, Electronics

Christ University, Bangalore

2019 - 2022

### MERN Stack Development

Brototype, Calicut

2022 - 2023

## AREAS OF EXPERTISE

- Node.js
- Express.js
- React.js
- MongoDB
- TypeScript
- Clean Architecture
- SOLID Principle
- Micro Services
- Docker
- Socket.io
- Firebase
- MVC Architecture
- Data Structure
- SASS
- EJS & HBS Template Engine
- Nginx
- REST API's
- Git
- AWS EC2
- HTML
- Tailwind CSS

## ABOUT ME

As a self-taught MERN developer passionate about web development, I possess strong proficiency in crafting intuitive and engaging React frontends. My skillset extends to leveraging Docker for Node.js and TypeScript. I've deployed projects with Docker, involving multiple Node.js apps, MongoDB, Redis for caching, and a React frontend. Committed to readable, maintainable code using clean architecture and SOLID principles, I'm driven to continually learn in this dynamic field.

## PROJECTS

### Waywin

Online trainers time booking platform

- [Front-end Repo](#)
- [Back-end Repo](#)

[Live](#)

Waywin is a multi-venture project, serving as a comprehensive trainer time booking website catering to psychologists, career guidance counselors, and motivational speakers. The project incorporates cutting-edge technologies to ensure a seamless user experience.

- Front-end: Developed using **React** with **TypeScript** for a robust and intuitive user interface.
- Back-end: Powered by **Node.js** and **Express** with **TypeScript** for enhanced maintainability and **scalability**.
- Database: **MongoDB** is employed as the database system, facilitating efficient data storage and retrieval for the application.
- Architecture: Follows **Clean architecture** and **SOLID principles** for better code organization.
- Real-time Communication: Implemented with **Socket.IO** for seamless user-trainer interactions.
- Multiventure Platform: Accommodates various trainers, psychologists, counselors, and motivational speakers.
- Trainer Time Booking: Intuitive booking system for easy appointment scheduling.

### Bookworm E-commerce

Online book ordering and delivery platform

- [Git Rep](#)

[Live](#)

Bookworm is an innovative E-commerce website specializing in selling books. The platform offers a seamless shopping experience for book enthusiasts with its user-friendly interface and feature-rich functionalities.

## ALSO TRIED

---

- Jest
- Vitest
- CI/CD
- Redux & Redux toolkit
- Redis
- Tailwind
- Postman
- JWT
- Github
- RabbitMq
- Mocha
- Figma
- Photoshpe
- MySql
- Docker
- Java
- Python

## INTEGRATIONS

---

- Razorpay
- Stripe
- S3
- CloudFront
- Cloudinary
- Twilio
- Node Mailer
- Firebase Verification
- Multer
- Formik
- JOI
- YUP
- Chart.js
- ESLint
- Thunk
- Prettier

- **Front-end:** The front-end of Bookworm is developed using **HTML** and **EJS** (Embedded JavaScript), providing an interactive and dynamic user interface that allows customers to browse, search, and purchase books with ease.
- **Back-end:** The back-end of the website is powered by **Node.js** and **Express**, enabling smooth data processing, handling user requests, and managing the overall server-side operations.
- **Database:** Bookworm utilizes **MongoDB** as its database system, ensuring efficient data storage and retrieval for the extensive collection of books available on the platform.
- **MVC Architecture:** The project follows the **Model-View-Controller (MVC) architectural** pattern, ensuring a clear separation of concerns and enhancing code maintainability and scalability.

## MINI PROJECTS

---

### ○ Order Product

• [Git Repo](#)

- A **microservices** backend project centered around managing product addition and orders. This endeavor involves utilizing **Node.js** and **Express** to establish resilient backend services.
- microservices architecture approach, utilizing **RabbitMQ** as a message broker to facilitate smooth communication among microservices. This approach guarantees scalability and fault tolerance.
- I am engaged in this project as a means to delve into the study of microservices.

### ○ GymRoutine

• [Git Repo](#)

- A **MERN stack** project designed to create, display, delete, and update daily gym routines.
- The backend is built using **Node.js** and **Express**, ensuring robust and efficient data handling. The front end utilizes **React** for an interactive user interface, providing a seamless experience for users.
- Utilized **MongoDB** for database management, implementing a dynamic and organized structure for storing routine inform

### ○ Todo

• [Git Repo](#)

- A **React** project that enables users to create, store, and manage todos efficiently.
- The application uses local storage to persist todo data on the client-side, providing a smooth and responsive user experience.

## DECLARATION

---

I hereby declare that all the information furnished above are true and correct to the best of my knowledge

**Musaffarul Islam**