

# MD. MAHBUBUL HOQUE SHOUROV

✉ [mahbubulhoqueshourov@gmail.com](mailto:mahbubulhoqueshourov@gmail.com) | 📞 +8801521528480 | [GitHub](#) | [LinkedIn](#) | [Portfolio](#)

## Problem Solving

- **Codeforces:** Solved 276 problems with a peak rating of 1304, participated in 40+ contests **Handle:** [Mahbub\\_Shourov](#)
- **Codechef:** Achieved 1753 rating (4★), completed 64 problems across 20+ competitions **Handle:** [shourov\\_06](#)
- **LeetCode:** Solved 163 problems, focusing on data structures and algorithms **Handle:** [shourov\\_mahbub](#)
- **HackerRank:** Earned 2★ in Problem Solving and 3★ in Python **Handle :** [@Shourov](#)

## Education

**Bachelor of Science in Mechatronics & Industrial Engineering**  
Chittagong University of Engineering and Technology (CUET)

2018 - 2023 | CGPA: 3.27 / 4.00

## Skills

**Programming Languages:** JavaScript , TypeScript , Python , C++

**Front-End:** React.js, Next.js, HTML5, CSS3, Tailwind CSS, Styled Components, Redux, Zustand

**Back-End & Database:** Node.js (Basic), SQL, Prisma ORM, Appwrite

**Developer Tools:** Git, GitHub, VS Code, Chrome DevTools, Webpack, Vite

**Methodologies:** Responsive Design, Component-Based Architecture, REST API Integration

## Projects

### MFS – Mobile Finance Service (MERN)

[Live Demo](#) | [Backend](#) | [Frontend](#) | [API Docs](#)

**Technology :** Next.js, Redux Toolkit, Tailwind CSS, Node.js, Express, MongoDB, JWT

- Developed a full-stack Mobile Financial Service platform with **RBAC** dashboards, **JWT** authentication, and financial transactions inspired by bKash and Nagad.
- Built a scalable frontend with **Next.js**, managing state via **Redux Toolkit**, integrating client-side validation, reusable UI components, and real-time toast notifications.
- Designed and implemented **RESTful** backend APIs using **Node.js**, **Express**, and **MongoDB** with **JWT** authentication, **PIN salting encryption**, and admin workflows for approvals and user management.
- Deployed backend on **Render** and frontend on **Vercel**, ensuring environment-based API integration and production readiness.

### Chat Application (MERN & Real-Time Communication)

[Backend](#) | [Frontend](#)

**Technology :** React 18, Vite, Zustand, Tailwind CSS (DaisyUI), Express.js, MongoDB, Socket.io, JWT

- Developed a modern real-time chat platform featuring user authentication, profile management, and responsive UI using **React 18** with **Vite** and **Zustand** for state management.
- Implemented backend APIs with **Express.js** and **MongoDB**, including **JWT**-based secure authentication, **RESTful** endpoints, and real-time messaging powered by **Socket.io**.
- Designed reusable frontend components, **dark/light theme** support, and user feedback mechanisms like **toast notifications** and **skeleton loaders** with **DaisyUI** and **React Hot Toast**.
- Built **robust real-time communication**, handling online user status, message delivery, and settings management and documented backend with **Swagger UI**.

### Socially - Full-Stack Blog Platform

[Live Demo](#) | [GitHub](#)

- Engineered a responsive Social platform using **Next.js** App Router and **TypeScript**, resulting in faster page load times compared to previous React implementations
- Implemented user authentication with **Clerk**, reducing authentication development time.
- Designed and built a normalized database schema with **Prisma ORM**, enabling efficient data retrieval and relationships
- Created reusable UI components with **Shadcn UI** and **Tailwind CSS**, achieving consistent design language across the application

### Sorting Visualizer - Interactive Algorithm Learning Tool

[Live Demo](#) | [GitHub](#)

- Created an interactive tool that visually demonstrates **6 different sorting algorithms**, helping users understand algorithm complexity.
- Implemented time complexity analysis features, allowing users to compare algorithm performance in real-time
- Built **custom animation** system using **React hooks** to visualize the step-by-step process of each algorithm
- Designed an intuitive UI with **styled-components** that allows users to adjust array size and sorting speed