SHOBHIT KUMAR SHARMA

📱 +91-7428116302 ✉️ sharmashobhit1000@gmail.com ⚫ Github 🌐 Portfolio ⟨/⟩ Leetcode 📍 Delhi, India

Summary

I learn best by doing and enjoy turning new challenges into learning moments. Whether it's building something from scratch or improving what exists, I focus on growing with each step. I value working with others to create things that are thoughtful, useful, and well-built.

Education

Jamia Hamdard University   **Sep. 2021 – May 2025**
Bachelor of Technology in Computer Science & Engineering CGPA : 7.8   *New Delhi, Delhi*

Relevant Courses: *Data Structures and Algorithms, Object-Oriented Programming, Database Management Systems, Operating Systems*

Technical Skills

Languages: C++, Python, JavaScript, TypeScript, SQL, NoSQL
Frameworks/Technologies: HTML/CSS, Tailwind, React.js, Redux.js, Node.js, Express.js, Flask
Databases: MySQL, MongoDB, Redis
Tools: AWS (EC2, S3, IAM, Lambda – familiar), Docker, Kafka, Prisma, Postman, Git, Jest

Experience

CodSoft   **Jan 2024 - Feb 2024**
FrontEnd Developer (Intern)   *Remote*
- Built responsive and interactive frontend interfaces, completing all assigned tasks within the **1-month deadline**, using tools like **Postman** for API testing and **ShadCN UI** for fast and consistent component-based development.
- Resolved over **15+ UI and logic bugs** through structured debugging, thorough testing, effective use of **Git/GitHub** for version control, and consistent self-driven research using online documentation and dev forums.
- Gained confidence in handling complete feature workflows from building and styling to testing and deployment while improving **component design**, layout structuring, and practical problem-solving through focused, hands-on learning.

Projects

Concisio | *(React, Node, Express, MongoDB, Gemini)*
- Followed **MVC architecture** and proper modularity from the start, keeping the code organized and easy to maintain, which helped speed up feature development by around **50%**.
- Faced duplicated logic and messy data flow, refactored components with **single responsibility** and centralized state management, reducing debugging time by nearly **40%** and improving app reliability.
- Backend was complex — modularized code with **clean practices** and used **MongoDB Atlas** for smooth data handling. Added PDF conversion and downloader features (including **HTML-to-PDF**), enabling quicker fixes and updates.

Linkly | *(React, Node, Express, Redis, MongoDB, Firebase)*
- Created a fast, custom URL shortener with **Base64** encoding that significantly improves link sharing and readability by over **70%**, making URLs neat and easy to use across all devices and platforms.
- Built user dashboards enabling **custom URLs**, click and time limits, and automatic deletion on expiry. Used **Redis hashmaps**, counters, and **Rate limiting** for efficient, real-time URL tracking and request control.
- Implemented secure authentication using **Firebase Auth** and **JWT tokens**, enabling verified Google and GitHub login with personalized access to manage URLs and view detailed click stats.

VoltSpot | *(React, Node, Express, MongoDB, OpenStreetMap)*
- Built a fully **searchable and filterable listing** of 25+ EV charging stations displayed on **OpenStreetMap**, offering smooth map interactions and detailed station data for intuitive user navigation.
- Implemented **MVC architecture** with clean folder structuring, well-defined **schema design**, and robust **data validations** using **express-validator** and client-side checks to ensure data integrity across 20+ form fields.
- Developed secure **routing** with protected endpoints and role-based access, integrated **server-side logging** using `morgan` for request tracking, and designed a fully **responsive UI** tested across 5 device types.

Honors / Awards

- Solved **400+** DSA problems on platforms like **LeetCode**, **GeeksforGeeks**, and **CodeStudio**.
- Secured a rank in the **top 1.5%** among **1.4 lakh+** participants in **CodeKaze 2024**.
- Earned a **Gold Badge** and ranked in the **top 1%** in NPTEL courses: **Programming in Java** and **Object-Oriented Programming**.