Ravikant

[redacted] [redacted] LinkedIn GitHub LeetCode

Education

Delhi Technological University **2022 – 2026**
B.Tech in Mathematics and Computing [redacted]
Web Development Head **Feb 2025 – Present**
Google Developer Student Club - DTU

Technical Skills

Programming Languages: C, C++, JavaScript, TypeScript, SQL
Frontend Technologies: ReactJs, NextJs, TailwindCSS, HTML, CSS, WebSockets
Backend Technologies: NodeJs, ExpressJs
Databases: **MongoDB**, MySQL, PostgreSQL, Prisma, Supabase
Cloud DevOps: AWS, Docker, Kubernetes
Developer Tools: Git, GitHub

Work Experience

AlgoUniversity (backed by Y-Combinator) **May 2024-July 2024**
Software Development Extern *Remote*

- Delivered an **Online Code Judge** capable of processing 50+ submissions with a 2s average response time by engineering an end-to-end evaluation pipeline from scratch.
- Boosted system efficiency by 30% during peak traffic by designing scalable solutions backed by comprehensive **High and Low-Level Design documentation**.
- Reduced API response latency by 20% by implementing advanced system design techniques learned through collaboration with engineers from **Google**, **Apple**, and **Microsoft**.
- Achieved 95% bug elimination during development by leading logical debugging sessions and facilitating **code quality reviews**.

Projects

Rubik's Cube Solver - using Korf's IDA* Algorithm **Source Code**

- Modeled a **virtual Rubik's cube (3x3)** in three different models/classes using standard data-structures present in C++.
- Achieved a solving time of under **3 seconds** for a Rubik's Cube **jumbled 8 times**, using **BFS**, **DFS**, **IDDFS**, algorithms.
- Implemented the **Korf's IDA* Algorithm** achieving a solving time under **10 seconds** for a Rubik's Cube **jumbled 13 times**.

Real-Time Collaborative Whiteboard Socket.IO ReactJS, NodeJs,ExpressJs, MongoDB **Deployed Link**

- Built a real-time collaborative whiteboard with sub-second latency for **multi-user sessions** by integrating Socket.IO with custom **event-driven logic**.
- Enabled seamless state management and **features like undo/redo** and **erasing** by utilizing React's **Context API** and optimized browser event handling.
- Delivered drawing functionalities (lines, rectangles, ellipses, arrows, text) by implementing intuitive UX using **raw JavaScript** and **canvas APIs**.
- Deployed scalable frontend/backend infrastructure via Vercel and Render, ensuring high availability and minimal downtime.

Online Code Compiler TypeScript, Next.js, Express, NodeJs, MongoDB **Deployed Link**

- Developed a **fully functional Online Judge** system by building a custom **multi-language compiler** from scratch, eliminating the need for third-party APIs.
- Achieved scalable backend performance by **containerizing compiler** services with **Docker**, deploying via **AWS EC2** and managing images on **AWS ECR**.