ASMIT SINGH

India | ashmitsingh068@gmail.com | LinkedIn | Github

Objective

Computer Science graduate seeking an entry-level Software Engineer position to leverage strong skills in Go, Node.js, and Typescript.

Experience

Software Engineer Intern, Raftlabs **(Nov[2024] - May[2025])**

- Collaborated on the **backend development** of Draftly an **AI-powered content creation** platform focused on LinkedIn post generation
- Built and maintained **serverless infrastructure** using **AWS Lambda functions** for scalable and efficient processing of user requests
- Implemented **GraphQL** API endpoints with **Hasura** to facilitate seamless communication between frontend and database
- Designed and optimized **PostgreSQL** database schemas to store user content and platform analytics
- Created a sophisticated **web crawler service** that safely extracts post data and analytics from LinkedIn while adhering to platform policies

Academic

Galgotia College of Engineering Technology **(2020 - 2024)**
Bachelor of Technology (Information and Technology) CGPA: 6.5

Army Public School No.2 Roorkee (81%) **(2018 - 2019)**

Skills

- **Relevant Coursework :** DBMS, Computer Networks, Data Structures & Algorithm, OS
- **Languages & Frameworks:** Go, TypeScript, Node.js, React, SQL
- **Databases & Messaging:** PostgreSQL, MongoDB, Redis, NATS
- **DevOps & Cloud:** Docker, Kubernetes, Jenkins, Git, AWS

Personal Projects

1. **GoRedis:** Redis-like Key-Value Database **Github** **(Go)**
   - Developed a **Redis-compatible** key-value database **server and client** in **Go**
   - Implemented **RESP protocol** for command parsing and communication
   - Utilized **goroutines** for concurrent request handling and **networking** concepts
   - Achieved **compatibility with standard Redis clients**

2. **Network Packet Analyzer:** CLI-based Network Analysis Tool Github **(Go)**
   - Developed a **CLI application** in **Go** for real-time network packet analysis
   - Integrated **Berkeley Packet Filter (BPF)** for customizable packet filtering
   - Provided **time-series data visualization** for protocol distribution and traffic patterns

3. **LSM-Tree Based Key-Value Store**: **Github** **(Go)**
   - Implemented core LSM-tree components: **in-memory balanced tree (AVL), memtable, SSTables, and compaction**
   - Designed a **Write-Ahead Log** (WAL) for crash recovery and data durability
   - Implemented concurrent access using Go's synchronization primitives (mutexes, channels)
   - Implemented **background compaction process** to optimize storage and query performance

4. **Multiplayer Pong Game**: Real-time Multiplayer Web Game **Github** **|** **Live** **(Javascript)**
   - Implemented **WebSockets** for real-time communication between players
   - Utilized **HTML5 Canvas** for smooth game rendering

5. **Video Editor**: Server-side Video Processing Application **Github** **(Javascript)**
   - Implemented **Node.js child processes** and **cluster module** for efficient task management
   - Created a **custom database** using server disk storage for file management
   - Integrated a **job queue** to manage video processing workload effectively
   - Enabled features like video resizing, audio extraction, and processed file downloads