

MOHAMMED MASROOR AHMED

Chicago, Illinois | +1 (773) 987-2423 | masroorahmed4021@gmail.com
masroor.codes | linkedin.com/in/mohammed-masroor | github.com/masroorahmed402

EDUCATION

DePaul University

Chicago, IL

Bachelor of Science in Computer Science (Concentration in Software Development)

Aug. 2022 – June 2027

Relevant Coursework: Software Testing, Design & Analysis of Algorithms, Computer Systems, Data Structures (Java), Data Analysis, Python Programming, Discrete Mathematics, Object-Oriented Programming, Technical Writing.

Achievement: Recipient of the State Scholar Plus Scholarship — \$100,000 (\$25,000/year over four years) awarded for outstanding academic performance and extracurricular involvement.

Certifications: Amazon Web Services (AWS) Certified Cloud Practitioner (In Progress — Expected Aug. 2026).

SKILLS

Languages: TypeScript, JavaScript, Java, Python, SQL (Structured Query Language), NoSQL

Frontend: React, HTML, CSS, Responsive Web Design

Backend: Node.js, Express, REST APIs, TypeScript/Java Services, ETL & Data Pipelines, Microservices

Databases: PostgreSQL, Redis, Relational & NoSQL Data Stores, SQL Query Optimization, Schema Design

Cloud & DevOps: AWS (Lambda, S3, EC2, RDS, IAM, CloudWatch), CI/CD Pipelines, Infrastructure-as-Code (IaC)

Tools: Git, Postman, Jira, WebSockets, API Integration, Unit Testing, Code Reviews, Agile/Scrum

Concepts: Data Structures & Algorithms, Object-Oriented Programming, API Design, Debugging & Testing

WORK EXPERIENCE

Fynite AI

Chicago, IL

Software Development Engineer Intern

May 2026 – Present

- Develop and optimize AI-powered workflow automation features using TypeScript, JavaScript, Node.js, React, and REST APIs, supporting integrations across 50+ enterprise systems and improving operational efficiency for internal workflows.
- Built scalable backend services and serverless functions on AWS (Lambda) for real-time data processing and AI-driven decision workflows, contributing to systems handling thousands of automated executions weekly.
- Supported data pipeline and integration workflows by writing and optimizing SQL queries against PostgreSQL for real-time processing, validation, and reporting of operational datasets.
- Designed and ran Postman test suites and unit tests against backend APIs to validate behavior and catch regressions ahead of CI/CD releases; reproduced, debugged, and verified features end-to-end in the platform's test environment.
- Collaborated in an Agile/Scrum environment, tracking work through Jira and participating in sprint ceremonies and code reviews to ship production-ready features, improving system reliability and API response performance.

PROJECTS

Task Queue System | *Node.js, PostgreSQL, Redis*

[GitHub](#)

- Designed and developed a task queue system using Node.js, PostgreSQL, and Redis, leveraging scalable architecture patterns commonly used in cloud-native systems.
- Implemented a worker-based architecture with retry mechanisms and dead-letter queues to support fault-tolerant and resilient asynchronous job processing.
- Enabled concurrent task execution across multiple workers, reducing processing time by ~30% under simulated load while improving throughput efficiency.

AI GitHub Repository Explainer | *OpenAI API, Node.js*

[GitHub](#)

- Built an AI-powered tool using the OpenAI API to analyze 50+ GitHub repositories and generate structured summaries.
- Developed backend services for API integration and optimized response handling, reducing analysis time by ~40%.
- Extracted key insights including architecture, tech stack, and features, minimizing manual code-review effort.

Real-Time Collaborative Whiteboard | *Node.js, Socket.io*

[GitHub](#)

- Developed a real-time collaborative whiteboard using Node.js and Socket.io, leveraging an event-driven architecture commonly used in scalable real-time communication systems.
- Implemented low-latency live synchronization (<100ms), undo functionality, and board-history replay to support responsive multi-user collaboration.
- Designed an event-driven architecture to handle 50+ real-time events/second across multiple clients, ensuring data consistency.