Mohammed Monjorul Hoque

Toronto, ON
437-673-4175 | monjorulmohammed@gmail.com | github.com/MMH576 | monjorul.vercel.app

Education

York University **Sep 2023 – May 2027 (Expected)**
Bachelor of Science in Computer Science (Honours) *Toronto, ON*

Relevant Coursework: Data Structures & Algorithm, Advanced OOP (Java), Programming in Python, Software Tools Software Tools (C, Bash, Linux), Discrete Math, Theory of Computation, Computer Architecture

Relevant Projects

Transactional Order & Payments Platform
NestJS, TypeScript, PostgreSQL, Prisma, Stripe, Docker — 2024

- Built a transactional order-processing backend enforcing **ACID guarantees** to safely handle concurrent checkout requests without inventory inconsistencies
- Verified transactional correctness through **unit and integration tests**, validating **order state-machine** transitions and simulating concurrent checkouts to ensure inventory consistency under **serializable database isolation**
- Integrated **Stripe payments** via server-side webhook verification, handling success, failure, and retry scenarios without trusting client-side state
- Enforced **JWT-based role separation** (Admin / Customer) and persisted **immutable audit records** capturing all inventory and order mutations

Event Processing Platform
NestJS, TypeScript, PostgreSQL, Redis, BullMQ, Docker — 2024

- Designed an **event-driven backend architecture** that decouples producers from consumers using **Redis queues**, allowing independent evolution of ingestion and processing logic
- Persisted all incoming events in **PostgreSQL**, enabling safe replay and reprocessing across **5+ distinct event types** without data loss
- Orchestrated **parallel background workers** with **BullMQ** to process events asynchronously, keeping API request latency independent of downstream workloads
- Implemented **idempotent event handlers** to safely handle duplicate deliveries and retries without corrupting system state

Webhook Delivery System
Node.js, TypeScript, ExpressJS, PostgreSQL, Redis, BullMQ, Docker — 2024

- Built an event-driven webhook delivery service guaranteeing **at-least-once delivery** to external HTTP endpoints by decoupling request ingestion from delivery execution using **Redis queues (BullMQ)**
- Implemented bounded retry policies with e**xponential backoff (up to 5 attempts)** to handle transient downstream failures while preventing unbounded retry storms
- Validated system correctness through isolated **unit and integration tests**, covering authentication, **HMAC signature verification**, retry and **dead-letter queue** logic, and **REST API behavior** using mocked **PostgreSQL**, **Redis**, and queue dependencies
- Routed unrecoverable delivery failures to a **dead-letter queue**, enabling manual inspection and replay without blocking healthy webhook traffic

Technical Skills

Languages: TypeScript, JavaScript, SQL, Python, Java, C
Frameworks / Technologies: NodeJS, NestJS, ExpressJS, ReactJS, PostgreSQL, Redis, BullMQ, REST APIs, JWT Authentication, Supabase
Developer Tools: Git, GitHub, Docker, Docker Compose, GitHub Actions, CI/CD, Postman, VS Code

Extracurricular

UNHACK Hackathon **Jan 2024**
Participant

- Created a project-collaboration prototype enabling students to post project ideas, define roles, and review applicants
- Worked in a time-boxed hackathon setting to scope requirements and deliver a functional demo