Harshana Lakshara Fernando

Panadura, Sri Lanka • +94 783 310 520 • 2020thlf@gmail.com • lakshara.21@cse.mrt.ac.lk
Portfolio • LinkedIn • GitHub

Summary

Backend-focused Computer Science Engineering undergraduate with a rigorous foundation in **System Design**, **Distributed Systems**, and **Cloud-Native Engineering**. Proven experience architecting high-performance, fault-tolerant applications using **Microservices**, **Serverless Architectures**, and **Event-Driven Design** with **Java (Spring Boot)**, **Go**, **AWS Lambda**, and **Kafka**. Validated through the delivery of high-throughput systems handling massive load, seeking to solve complex **concurrency** and **low-latency** challenges within high-velocity engineering teams.

Education

B.Sc. (Hons) in Engineering – Computer Science and Engineering *2021 – Present*
University of Moratuwa, Sri Lanka

Relevant coursework: Programming Fundamentals, Program Construction (OOP), Data Structures and Algorithms, Advanced Algorithms and Advanced Data Structures, Parallel and Concurrent Programming, Operating Systems, Programming Languages, Theory of Computing, Advanced Software Engineering, Database Systems, Computer Networks, Computer Architecture, Calculus, Linear Algebra, Numerical Methods, Operational Research, Statistics, Introduction to AI, IoT Devices

Experience

Software Engineer Intern *Dec 2024 – May 2025*
Information Systems Associates (Pvt) Ltd — Aviation Technology Company

- **AeroOrder — Revamped Order Management Microservice (Java 18, Spring Boot)**
  - Engineered a **distributed**, **high-throughput** unique ID generator for Passenger Name Records (PNRs) using a **Kafka**-backed pool model, ensuring uniqueness and security across scaled instances while adhering to aviation domain standards.
  - Optimized the PNR generation service to achieve **25+ PNRs/sec throughput** per node with **<100ms latency**, validating performance and resilience through extensive load, stress, and spike testing with **Apache JMeter**.
  - Resolved critical **production** issues, including **Kafka** consumer lag and offset resetting, by implementing robust shutdown hooks and enhanced retry mechanisms to improve fault tolerance and system recoverability.
  - Led **Research and Development** for **asynchronous load testing**, establishing a monitoring stack with **Prometheus** and **Grafana** to collect and visualize key performance metrics like queue time and latency.
  - Investigated and refactored inefficient **Kafka Streams** processing logic, building a proof-of-concept to validate a more efficient **Producer-Consumer** model that significantly reduced consumer lag and improved system stability.
  - Modernized the logging infrastructure by implementing structured **JSON logging**, which removed legacy dependencies and improved log parsing efficiency across all microservices.
- **AeroMart — Legacy Monolithic Passenger Reservation System (Java 8 ,Ant)**
  - Architected and implemented a hybrid integration strategy by embedding a **gRPC** client and **Kafka** consumer into a **legacy Java 8** monolith (AeroMart), enabling seamless, backward-compatible communication with the new **AeroOrder microservices** platform.
  - Developed dynamic, database-driven **feature toggles** to control PNR routing logic, minimizing risk during the phased **production rollout** by allowing for granular control and immediate fallback.
- **Other**
  - Authored comprehensive **design documents**, **sequence diagrams**, and **integration plans** on Confluence; facilitated knowledge transfer sessions to ensure seamless handover to the core development team.
  - Actively contributed to the **agile development lifecycle**, participating in all ceremonies, performing **code reviews**, and collaborating with cross-functional teams to deliver **production-ready** features using **Jira** and **Bitbucket**.

KEYWORDS: Java 18, Java 8, **Spring Boot**, **Spring Security**, Ant, **gRPC**, **Kafka**, React, Protobuf, REST API, Couchbase, Oracle DB, Oracle SQL, PostgreSQL, **JMeter**, Gatling, Ant Design, **Docker**, Docker Compose, Gradle, **Prometheus**, **Grafana**, Kibana, Confluence, Jira, Bitbucket, Microservices, **Kafka Streams**, JSON

Projects (All projects 🔗)

1. Improving Blockchain Scalability: Throughput Enhancement in ZK-Rollups (Ongoing | FYP)

Final Year Research Project (Ongoing) — BSc (Hons) in Computer Science & Engineering, University of Moratuwa

- Designed and implementing a modular ZK-Rollup prototype to empirically study the performance and economic trade-offs of batching, sequencing, and data availability strategies on Ethereum Layer 2.
- Architecting a runtime-swappable ZK-Rollup framework with clearly separated modules for batching, sequencing, state execution, proving, and data availability, to support controlled, reproducible experimentation.
- Implementing core rollup components including an off-chain sequencer, a Sparse Merkle Tree–based state manager, zk-SNARK circuits in Circom, and Solidity bridge + verifier contracts deployed on Ethereum testnets.
- Building a dedicated benchmarking harness to drive synthetic workloads and measure throughput (TPS), latency, gas per transaction, and proof time under different configurations.
- Evaluating the impact of batch size and batch frequency, sequencer scheduling policies (e.g., FIFO vs fee/priority-based), and data availability modes (L1 calldata vs EIP-4844 blob transactions) on end-to-end performance and cost.
- Generating empirical trade-off curves (e.g., throughput vs latency vs gas/tx) and configuration insights to guide the economic optimization of ZK-Rollup systems in the post–EIP-4844 environment.
- Planning to release the prototype, benchmarking scripts, and datasets as open source to support future research on rollup scalability and protocol tuning.

Keywords: Ethereum, Layer 2, ZK-Rollups, Zero-Knowledge Proofs, zk-SNARKs, Circom, Groth16/Plonk, Solidity, TypeScript, Node.js, Sparse Merkle Trees, EIP-4844, Blockchain Scalability, Performance Benchmarking, Docker

---

2. Blockchain Supply Chain Management DApp (2025 | Individual) Architecture | Live 🌐 | Github

Built a full-stack decentralized application (DApp) enabling end-to-end shipment tracking and secure escrow payments on the Ethereum blockchain (Polygon Amoy Testnet).

- Developed a production-grade Solidity smart contract (`Tracking.sol`) using `structs`, `mappings`, and a strict 3-state `enum` lifecycle (PENDING → IN_TRANSIT → DELIVERED), complete with guarded state transitions, timestamped event logs, and full on-chain auditability.
- Implemented a trustless escrow mechanism using `payable` functions—validating `msg.value`, locking funds on creation, and releasing payments only upon verified delivery via `completeShipment`.
- Built a responsive Next.js/React frontend powered by a global `TrackingContext`, providing real-time state management, shipment inspection, and lifecycle actions (create, start, complete).
- Integrated `Ethers.js` and `Web3Modal` for secure wallet connection, dynamic provider/signer selection, MetaMask interaction, and seamless network handling (Localhost Polygon Amoy).
- Added data-formatting utilities for user-friendly display, including UNIX→date conversion and Ether/Wei transformations using `ethers.utils.formatEther`.
- Implemented polished UI/UX behaviors—loading indicators, masked modals (`maskClosable=false`), and utility helpers (`shortenAddress`, `copyToClipboard`) for high-quality user interaction.
- Designed a manual CI/CD pipeline using GitHub Actions to automate contract deployment (with pre-flight balance checks) and frontend deployment to Vercel, fully parameterized via GitHub Secrets and environment isolation.
- Utilized Hardhat for development, including local blockchain simulation (`npx hardhat node`), compilation, contract verification, and controlled deployment workflows.

Keywords: Solidity, Hardhat, Ethers.js, Web3Modal, Next.js, React Context API, Tailwind CSS, Polygon Amoy, Smart Contracts, Escrow Payments, CI/CD

---

3. Governance DAO – Token-Weighted Voting DApp (2025 | Individual)

A decentralized autonomous organization (DAO) voting application on Ethereum.

- Developed **Solidity** smart contracts for proposal creation and token-weighted voting.
- Implemented an ERC-20 governance token for voting power.
- Built a frontend using **React** and **Web3.js** to interact with the blockchain.
- Deployed contracts to the **Sepolia** testnet using **Hardhat**.
- Ensured security through comprehensive unit tests and access control modifiers.

Keywords: Solidity, Ethereum, React, Web3.js, DAO, Smart Contracts, Hardhat

---

4. Fintech & Crypto Trading Automation Lab (Ongoing | Group)

Industry-Mentored Initiative (Ongoing)

Selected to participate in an exclusive, industry-mentored initiative focused on cutting-edge development in fintech, virtual assets, and derivative markets.

- Selected based on strong development skills to work on intelligent automation and production-level projects with real-world deadlines.
- Developing high-performance trading automation tools and fintech solutions with opportunities for mutual ownership.
- Gaining deep exposure to technical analysis (TA), algorithmic trading, and asset management strategies across crypto and forex markets.
- Receiving direct industrial mentorship aimed at long-term growth in personal finance, wealth management, and professional software engineering practices.

Keywords: Fintech, Crypto Trading, Forex, Algorithmic Trading, Technical Analysis, Intelligent Automation, Industrial Mentorship, Personal Finance & Wealth Management

---

Professional Training, Coursework & Certificates (All credentials 🔗)

- **Ultimate AWS Certified Solutions Architect Associate SAA-C03 (Exam Prep Course)** *Aug 2025*
  Udemy — Instructor: Stephane Maarek — View certificate
- **Certified Kubernetes Administrator (CKA) with Practice Tests (Exam Prep Course)** *Apr 2025*
  Udemy / KodeKloud — Instructor: Mumshad Mannambeth — View certificate
- **Apache Kafka Series (v3)** *Mar 2025*
  Udemy — Instructor: Stephane Maarek — Part 1 | Part 2 | Part 3
- **100 Days of DevOps** *Nov 2025*
  KodeKloud — Trainer: Mumshad Mannambeth — View certificate
- **Implementing Software Architecture for Large-Scale Systems** *Apr 2025*
  Udemy — View certificate
- **Advanced OAuth Security** *Mar 2024*
  Udemy — View certificate
- **Next.js: Build Scalable React Apps with Page & App Routers** *Nov 2025*
  Udemy — Instructors: Anton Voroniuk, Denys Kohut, Anton Voroniuk Support — View certificate
- **Fundamentals of Deep Learning** *Dec 2025*
  NVIDIA Deep Learning Institute (DLI) — View certificate
- **Large Language Models (Level 1–3)** *Nov 2025*
  Udemy — H2O.ai University (Instructor: Andreea Turcu) — Level 1 | Level 2 | Level 3
- **Mastering Solidity, the Ethereum Programming Language** *In progress*
  Udemy
- **Cardano Blockchain Certified Associate (CBCA)** *In progress*
  Cardano Academy

Skills

Languages: Java, Go, Python, TypeScript, Solidity

Backend: Spring Boot, Spring Cloud, Spring Security, Express.js, Gin, Resilience4j, Node.js

Frontend: React, Next.js, React Native, Ant Design, ShadCN UI, Three.js, Vite

APIs , Messaging & Protocols: REST, gRPC, tRPC, WebSockets, Protobuf, Kafka, RabbitMQ JWT, OAuth 2.0, Kong

Databases & Caching: PostgreSQL, MySQL, Oracle DB, Oracle SQL, DynamoDB, Redis, Elasticsearch, Cassandra, Couchbase, Supabase, Hibernate, Prisma

Cloud & DevOps: AWS , OCI, Docker, Docker Compose, Kubernetes, Helm, Terraform, ArgoCD, GitHub Actions, Jenkins, Serverless Framework, Nginx, Nexus, GHCR, Certbot, DuckDNS

Blockchain: Ethereum, Solidity, Hardhat, Web3.js

Tools & Concepts: Git, System Design, Distributed Systems, Microservices, Event-Driven Architecture, Jira, JMeter, Gatling, Gradle, Ant, Confluence, Bitbucket, Pytest, Makefile, Postman,

References available upon request.