

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. Employee Management System (EMS) (2025 | Individual) Architecture | Github

Designed and built a comprehensive, cloud-native microservices application for employee and department management, featuring a full-stack architecture with a React/TypeScript frontend and Spring Boot backend.

- Architected a distributed system with independent microservices for employees, departments, and configuration, using **Spring Cloud** for service discovery (**Eureka**), centralized configuration, and an API gateway.
- Implemented robust fault tolerance using **Resilience4j**, incorporating **Circuit Breaker**, **Retry**, and **Rate Limiter** patterns to ensure system stability during partial failures.
- Enabled dynamic configuration management using **Spring Cloud Config Server** and **Spring Cloud Bus** with **RabbitMQ** for real-time property updates across services.
- Established a robust observability stack, integrating **Zipkin/Sleuth** for distributed tracing, the **ELK Stack** for centralized logging, and **Prometheus/Grafana** for real-time metrics and monitoring.
- Containerized all services using **Docker** and orchestrated local deployments with **Docker Compose**, while targeting **Kubernetes (OKE)** for production.
- Designed and managed a CI/CD pipeline using **GitHub Actions**, with **ArgoCD** for GitOps-based deployments to the Kubernetes cluster.

Keywords: Java 17, Spring Boot, Spring Cloud, React, TypeScript, Vite, Nginx, RabbitMQ, Resilience4j, MySQL, Docker, Kubernetes, GitHub Actions, ArgoCD, Prometheus, Grafana, ELK Stack, Nexus, GHCR

2. GoTogether – Intelligent Travel Companion for Sri Lanka (2024 | Group) Architecture | Live | Github

Developed and deployed a full-stack travel assistant platform tailored for Sri Lankan tourism, offering AI-based itinerary generation, real-time transport data, and social interaction features through a cloud-native microservices architecture.

- Led development of the web frontend using **Next.js** with secure authentication, session handling via cookies, SSR/CSR hybrid rendering, and route protection for authenticated access.
- Designed and implemented the **User Management module** within the **Social Media Service (Spring Boot)** to manage profiles, preferences, and social connections, integrating securely with **Keycloak** for identity management.
- Built a dedicated **Auth Service using Keycloak**, managing OAuth2/OIDC flows, token issuance, and automated realm/client boot-time import scripts.
- Solved startup dependency issues between services by writing a custom wait script to delay the social media service until Keycloak initialization completed, improving reliability.
- Configured **HTTPS for production with NGINX + Certbot + DuckDNS**, ensuring compatibility with Vercel's secure API policies.
- Developed the **API Service (Go)** to provide AI-based itinerary generation (via Gemini API) and map services using **gRPC** for high-performance inter-service communication.
- Optimized container builds using multi-stage Docker builds (Go service) and manual JAR injection (Spring Boot service) to reduce image size and speed up builds.
- Created a robust **deploy.sh script** to support local builds, selective service orchestration, and GitHub Actions-based CI/CD.
- Migrated backend infrastructure from **AWS EC2 to Oracle Cloud Free Tier**, improving stability, memory headroom, and deployment speed.

Keywords: Next.js, React Native, Go, Spring Boot, Kong, Keycloak, PostgreSQL, Cloudflare, Vercel, NGINX, Certbot, DuckDNS, Gemini API

3. Serverless Auction Platform (2025 | Individual) Architecture | Github

A capstone project built using AWS, following a microservices architecture with an event-driven, serverless (FaaS) design.

- Architected an event-driven microservices system (Auctions, Auth, Notifications) using **AWS Lambda**, **API Gateway**, and **EventBridge** for scheduled tasks.
- Implemented asynchronous service communication using **SQS** and **SNS** to decouple components and handle email notifications via **SES**.
- Designed a high-performance data model in **DynamoDB** utilizing **Global Secondary Indexes (GSI)** for efficient scheduled querying and **On-Demand Capacity** for auto-scaling.
- Secured API endpoints with a custom **Lambda Authorizer** that validates **JSON Web Tokens (JWT)**, ensuring robust role-based access control.

- Enhanced code reliability and maintainability by integrating **Middy** middleware for input validation and error handling.
- Utilized **LocalStack** and **Docker** to create a comprehensive local development environment that emulates AWS cloud infrastructure.

Keywords: Node.js, Serverless Framework, AWS Lambda, DynamoDB, SQS, SNS, SES, JWT, Middy, LocalStack, Docker

Other Relevant Projects

4. Automated AI Shorts Video Creator (2025 | Individual)

An open-source project to automate the creation of short-form videos using a powerful combination of AI tools and workflow automation.

- Orchestrated a video creation pipeline using **n8n** as the central workflow engine.
- Containerized the entire stack using **Docker Compose** for one-click local deployment.

Keywords: n8n, Docker, AI, Workflow Automation, Video Generation

5. Food Delivery Application (2024 | Group) Architecture | Github

Built a scalable, event-driven food delivery platform using a modern microservices architecture.

- Developed the Restaurant Module with secure sign-up/login (Spring Security, JWT) and CRUD APIs with Redis caching.
- Enabled real-time order sync and notifications using Kafka and WebSockets.

Keywords: Java, Spring Boot, gRPC, Kafka, Redis, WebSocket, Docker

6. Go gRPC GraphQL – Multi-Tenant Microservices (2025 | Individual)

Architected a polyglot, multi-tenant microservices platform with a focus on scalability and unified data access.

- Developed backend services in Go using gRPC and Protobuf for inter-service communication.
- Implemented a GraphQL API Gateway and designed multi-tenant data storage with PostgreSQL and Elasticsearch.

Keywords: Go, gRPC, GraphQL, Protobuf, PostgreSQL, Elasticsearch

7. RPAL Interpreter – Functional Language Compiler (2024 | Group)

Engineered a complete compiler and interpreter for the RPAL language from scratch.

- Developed a Lexical Analyzer, AST/ST generator, and a CSE Machine for evaluation.
- Built a robust testing framework with Pytest covering over 168 test cases.

Keywords: Python, Pytest, GitHub Actions, Makefile

Professional Training, Coursework & Certificates (All credentials

- | | |
|--|--|
| <ul style="list-style-type: none"> • Ultimate AWS Certified Solutions Architect Associate SAA-C03 (Exam Prep Course)
Udemy — Instructor: Stephane Maarek | <i>Aug 2025</i>
View certificate |
| <ul style="list-style-type: none"> • Certified Kubernetes Administrator (CKA) with Practice Tests (Exam Prep Course)
Udemy / KodeKloud — Instructor: Mumshad Mannambeth | <i>Apr 2025</i>
View certificate |
| <ul style="list-style-type: none"> • Apache Kafka Series (v3)
Udemy — Instructor: Stephane Maarek | <i>Mar 2025</i>
Part 1 Part 2 Part 3 |
| <ul style="list-style-type: none"> • 100 Days of DevOps
KodeKloud — Trainer: Mumshad Mannambeth | <i>Nov 2025</i>
View certificate |
| <ul style="list-style-type: none"> • Implementing Software Architecture for Large-Scale Systems
Udemy | <i>Apr 2025</i>
View certificate |
| <ul style="list-style-type: none"> • Advanced OAuth Security
Udemy | <i>Mar 2024</i>
View certificate |
| <ul style="list-style-type: none"> • Next.js: Build Scalable React Apps with Page & App Routers
Udemy — Instructors: Anton Voroniuk, Denys Kohut, Anton Voroniuk Support | <i>Nov 2025</i>
View certificate |
| <ul style="list-style-type: none"> • Fundamentals of Deep Learning
NVIDIA Deep Learning Institute (DLI) | <i>Dec 2025</i>
View certificate |
| <ul style="list-style-type: none"> • Large Language Models (Level 1–3)
Udemy — H2O.ai University (Instructor: Andreea Turcu) | <i>Nov 2025</i>
Level 1 Level 2 Level 3 |
| <ul style="list-style-type: none"> • Mastering Solidity, the Ethereum Programming Language
Udemy | <i>In progress</i> |

- **Cardano Blockchain Certified Associate (CBCA)**
Cardano Academy

In progress

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.