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 (Fullstack Focus) *Dec 2024 – May 2025*
Information Systems Associates (Pvt) Ltd — Aviation Technology Company

- 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.
- 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.
- 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.
- Integrated frontend components with backend services to deliver end-to-end features.

Keywords: Java 18, Spring Boot, React, gRPC, Kafka, REST API, PostgreSQL, Docker, Microservices, Agile.

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

Built a scalable, event-driven food delivery platform inspired by systems like Uber Eats, using a modern microservices architecture. As part of the core team, I focused on designing and developing the Restaurant Service.

- Developed the Restaurant Module with secure sign-up/login using **Spring Security** and **JWT** (with extended claims).
- Built CRUD APIs for menu and item management; integrated **Redis caching** to improve performance and reduce database load, with appropriate cache invalidation strategies.
- Enabled order sync via **Kafka**: when an order is placed, an event is published to the 'order_created' topic, which the restaurant service consumes to update order details.
- Implemented real-time order state updates through Kafka's publish/subscribe model to keep restaurant and rider services in sync.
- Integrated **WebSockets** for real-time restaurant notifications (e.g., new orders, rider assignments).
- Built a secure "Forget Password" flow using time-sensitive tokens for authentication recovery.
- Implemented robust exception handling and API protection through middleware guards and JWT verification.
- Dockerized all components (frontend, backend, Kafka, Redis, PostgreSQL) and managed them via **Docker Compose**.
- Worked in Agile sprints using **Jira** for task tracking and **Bitbucket** for version control.

Keywords: Java, Spring Boot, gRPC, Kafka, Redis, PostgreSQL, WebSocket, Docker, React.js, Ant Design

4. Maison Vella – E-Commerce Admin Panel (2024 | Individual) Architecture | Live | Github

Comprehensive admin interface for the Maison Vella e-commerce ecosystem, facilitating centralized management of products, orders, and content assets.

- Engineered a responsive **Next.js 14** application using the **App Router**, **ShadCN UI**, and **Tailwind CSS**, ensuring a consistent and accessible design system.
- Implemented a robust backend architecture utilizing **Server Actions** for form handling and **tRPC** with **TanStack Query** for type-safe client-server communication.
- Orchestrated content management by integrating **Sanity** as a **Headless CMS**, consuming data via **GraphQL** for flexible and efficient retrieval.
- Managed transactional data using **Prisma ORM** with **Supabase (PostgreSQL)** in a **Serverless** environment.

- Architected the system using **CQRS**, the **Repository Pattern**, and a **Layered Architecture** within a feature-sliced structure to ensure scalability and maintainability.
- Secured administrative access and role-based operations using **Clerk Auth**.
- Optimized asset delivery and storage by implementing **Cloudflare R2** (S3-compatible) and a global **CDN** for high-performance media handling.

Keywords: Next.js 14 (App Router), Server Actions, TypeScript, tRPC, GraphQL, Prisma, Supabase, Sanity (Headless CMS), Serverless, Cloudflare R2 (CDN), CQRS

5. Virtual Me – 3D Animated AI Chatbot (2024 | Individual)

Created an interactive 3D virtual assistant with lifelike animations and AI-driven conversations.

- Integrated **Hugging Face** (Mixtral model) for natural language dialogue.
- Implemented **Eleven Labs** TTS for human-like voice output.
- Synchronized lip movements and facial expressions using **Rhubarb**.
- Built immersive 3D interface with **React** and **Three.js**.
- Developed backend using **Node.js** and **Express.js** for API orchestration and audio processing.

Keywords: React (Three.js), Node.js, Express.js, Hugging Face, Eleven Labs, Rhubarb, FFmpeg

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.