Sandaru Wijewardhana

SOFTWARE ENGINEER

No.630/17/20/1, Nedungamuwa, Gampaha, Sri Lanka +94 72 935 6781
sandaruwijewardhana@gmail.com github.com/sandaruwijewardhana LinkedIn Medium

Summary

Computer Engineering graduate (GPA 3.83/4.00, First Class) who builds and runs production systems end to end. Founder and sole engineer of **BayWork**, a multi-tenant SaaS, and author of the **Harbor Kubernetes operator** behind WSO2's Registry as a Service. Active open-source engineer with merged upstream contributions to **Prometheus**.

Education

University of Peradeniya
BSc Eng (Hons), Computer Engineering, GPA : 3.83/4.00 – First Class — *2022 – Oct 2026*

Ananda College, Colombo 10
G.C.E. Advanced Level, 3 A's — *2012 – 2020*

Experience

1. Site Reliability Engineer Intern - WSO2 — Apr 2026 – Oct 2026

Go · Kubernetes · Kubebuilder · Helm · Harbor · Harvester HCI · Rancher · RKE2 · cert-manager · Longhorn · Terraform · Argo Workflows · Prometheus

- Architected and built **Registry-as-a-Service** as the sole engineer for **Open Cloud Datacenter** (WSO2's open-source private cloud) designed CRDs, operator reconcilers, and **3-level vertical auto-scaling** that reduced wasted resource usage by up to **80%**.
- Automated multi-tenant **7-pod Harbor deployments in under 2.5 minutes** slashing manual setup time from **1 hour (>95% setup time reduction)**.
- Engineered end-to-end testing for the VM-provisioning capability module within a team building an automated **Harvester HCI upgrade acceptance-testing suite** using Argo Workflows, Terraform, and Go assertions.

2. Founder & Sole Engineer - BayWork — May 2024 – Present

TypeScript · Node 20 · Express 5 · Drizzle ORM · PostgreSQL 15 · React 19 · Vite · TailwindCSS · Docker · AWS · Cloudflare

- Founded and single-handedly run a multi-tenant B2B SaaS (baywork.lk) ERP system **8 stations deployed, 1st paying customer onboarded**. With no co-founder or team, I own **market research, pricing, branding, architecture, backend, frontend, database, security, infrastructure, and support**.
- Implement **schema-per-tenant isolation** on PostgreSQL 15 and **OWASP ASVS 5.0 L2 security** (JWT, RBAC, and `search_path` SQLi defense).
- Delivered **p95 API latency < 120 ms** on a base footprint. Designed a multi-AZ AWS (**ECS on EC2 + RDS Multi-AZ**) using **Cloudflare Tunnels**, slashing unit costs to **$0.90/station (reducing 30% of)** at 50 station.

3. Undergraduate Teaching Assistant - University of Peradeniya — Apr 2025 – Apr 2026

- Ran lab sessions and marked assessments for **CO225** Software Construction, **CO322 / CO2030** Data Structures & Algorithms I & II, and **3YP** third-year IoT and web projects

Open Source Contributions

- **prometheus/prometheus #19522** (★ 65.8k 10.8k) - **merged**. The remote-write "Send Batch Duration" panel queried `_bucket` series only, so it rendered empty for anyone scraping native histograms. Added two `histogram_quantile` targets beside the classic ones so one panel works under either representation with no configuration.
- **keycloak/keycloak #51935** (★ 36.4k 8.9k) - open. Only remove filtered values from multivalued simple attributes, fixing consent-screen data loss on user profiles.
- **helm/helm #32568** (★ 30.2k 7.8k) - open. `fix(provenance)`: decode each armored keyring block from its own reader, so a keyring holding multiple public keys no longer fails verification after the first block.
- **argoproj/argo-cd #29255** (★ 24.0k 7.8k) - open. `fix(ui)`: hide the pod-highlight control when a resource has only one pod.
- **ceph/ceph-csi #6482** (★ 1.6k 609) - open. Refactored subvolume pinning to use `go-ceph`'s typed `PinSubVolumeInGroup()` API, removing hand-rolled JSON command marshalling and dead helper code.

Research

Explainable-AI-Driven Zero-Trust Anomaly Detection for Encrypted Traffic *FYP, group* | Repo

Python · Decision Trees · Deep Dictionary Learning · Isolation Forest · LIME / SHAP

- Detecting attacks inside encrypted traffic without decrypting it, under Zero-Trust assumptions, and explaining every decision. As lead architect I designed the two-stage pipeline, built the Decision Tree stage, and did the packet analysis and feature engineering.
- Full explainable analysis cost around **1.2 s per stream**, unusable inline. Splitting it into a fast Decision Tree triage stage (**1.2 ms**) that escalates only suspicious streams to deep analysis brought the weighted average to **463 ms - a 2.6× speed-up** - while holding **96.69% accuracy, 99.31% recall and 95.12% F1** over 3,000 streams, with only **6 attacks leaked**. The triage classifier alone scores **99.96% attack recall over 523,534 flows** at **0.04% leakage**.
- **Accepted for Publication:** Paper accepted for publication at the **iPURSE 2026** conference.

Projects

1. BayWork - Multi-Tenant SaaS Platform *Solo, in production* | baywork.lk | Demo

Node 20 · Express 5 · Drizzle ORM · PostgreSQL 15 · React 19 · TailwindCSS 4 · TanStack Query · Zod · Docker · ECS · RDS · Cloudflare Tunnel · Terraform · GitHub Actions

- Workshop management for customers, vehicles, job cards, invoicing, inventory and role-scoped staff, with a per-station dashboard. with TypeScript across a 10-module API and a React console, all written alone; **p95 API latency under 120 ms**, load-tested to **200 req/s** on a single 2-vCPU host - about **10× projected peak** at 50 stations.
- Chose **schema-per-tenant** over a shared `tenant_id` column so a forgotten `WHERE` clause cannot leak another station's data - an approved registration builds **13 tables, 4 enum types and 27 indexes in one transaction in under 500 ms**, rolling back cleanly on failure. Closed a `search_path` injection path by validating every schema identifier before it reaches SQL, and cover the isolation and rollback paths at ~**70% API test coverage**.
- Designed the low cost deployment ECS on EC2 with Multi-AZ RDS, fronted by a Cloudflare Tunnel. Dropping the ALB and NAT Gateway saves **$53/month** and leaves **zero inbound ports open at 50 stations ($0.90/station)** - cheaper at **99.5% business-hours availability, 5-minute RPO and 15-minute RTO**.

2. Registry-as-a-Service - Kubernetes Operator *WSO2, individual* | Repo

Go · Kubebuilder · controller-runtime · Helm Go SDK · Harbor REST API · Harvester HCI · RKE2 · cert-manager · Longhorn · Prometheus

- Architected and built a production Go Kubernetes operator (controller-runtime) automating per-namespace Harbor registries. Implemented two CRs `RegistryBackend` (per-tenant server) and `RegistryInstance` (scoped projects), mirroring cert-manager's model.
- Cut per-tenant storage ~**80% (100 GB → 20 GB)** via level-triggered autoscaling on a 3-tier plan growing at an 80% quota threshold and shipped **full tests (78% REST client coverage)**.
- Designed isolation, resilience, and scale architecture across tenants: engineered a PrivateLink-style network model achieving isolation with **0 routable IPs (vs. every namespace loadBalancers)** while closing unauthenticated cross-tenant paths; implemented ~**80% CPU reduction and 10× throughput** moving image traffic from L7 to L4; and reduced backup volume **24%** using a database-before-blobs ordering rule for self-healing garbage collection.

3. Harvester Upgrade Acceptance-Test Suite *WSO2, team - in progress* | Repo

Go · Ginkgo v2 / Gomega · client-go · dynamic client · Terraform · Argo Workflows · JUnit · structured JSON reporting

- A reusable pipeline that proves the platform capabilities we depend on still work after a Harvester or Rancher upgrade, instead of finding out from a customer. I own the **VM-provisioning capability module** - fixtures, **20+ behavioural assertions** and evidence collection.
- Terraform can prove a resource was applied but not that it behaves, so the suite pairs Terraform-managed throwaway fixtures with Go acceptance tests that wait on real reconciliation against explicit deadlines.
- Keeping the runner in a separate **Host** cluster from the **Target** under test means logs, results and cleanup survive even when the Target capability is the thing that broke - turning **a day of manual post-upgrade checks into a 15-minute automated run**, with guaranteed Terraform teardown and a scheduled janitor so no fixture is left orphaned.

4. LIGHTIFY - Smart Indoor Lighting System *Group, 3rd-year project* | Repo | Page

ESP32 (C/C++) · MQTT · AWS IoT · Java Spring Boot · MongoDB · React Native

- An IoT system that senses room occupancy and ambient light, adjusts indoor lighting automatically, and reports state to a mobile app in real time. I led the firmware - the ESP32 code that reads the sensors, debounces occupancy and publishes to the cloud.
- Chose **MQTT** over HTTP polling to keep devices responsive on an unreliable Wi-Fi link, and moved occupancy debouncing onto the device so flickering sensor reads never reached the backend.
- Sensor-to-light response held **under 1 second**, and occupancy-driven switching cut idle lighting time substantially against an always-on baseline.

5. Cardiovascular Artery Scanning System *Group, CO227* | Repo

Java · Java Swing · SQL · OpenCV · image processing · ML models

- A clinical tool that measures artery diameter from live scanner video so clinicians can track how a patient's readings move over time rather than reading each scan in isolation. I architected the system flow and the patient-history schema.
- Stored every measurement against a **timestamped patient record** instead of overwriting the last one - which is what made longitudinal comparison possible - and replaced manual caliper readings with automated measurement free of operator variance.

6. SkinBot - CNN Skin Type Analyser *Group, Neural Networks* | Repo

Python · TensorFlow · Convolutional Neural Networks · Tkinter

- Classifies facial skin as oily, dry or normal from a photograph. I trained the classifier and built the interface.
- The first model learned lighting conditions rather than skin texture, which inflated validation accuracy on an unbalanced set - fixed with augmentation and class re-balancing, then re-validated on a held-out split.

Technical Skills

Languages: Go, Java, TypeScript, JavaScript, Python, SQL, C, Bash, Verilog

Backend: Node 20, Express 5, Spring Boot, REST APIs, Drizzle ORM, JWT / OAuth2, RBAC, multi-tenancy

Frontend: React 19, Vite, TailwindCSS, TanStack Query, React Hook Form + Zod, React Native, Java Swing, HTML5

Data & Messaging: PostgreSQL, MySQL, MongoDB, Redis, Memcached, RabbitMQ, MQTT, JDBC

Cloud & Infrastructure: Kubernetes (operators, CRDs, kubebuilder), Docker, Helm, Terraform, Vagrant, Nginx, Tomcat, Harvester HCI, Rancher, RKE2, Cloudflare (Tunnel, R2, WAF); **AWS** - EC2, ECS, RDS, S3, ECR, IAM, CloudWatch

DevOps, Observability & Testing: GitHub Actions, Jenkins, GitLab CI, Argo Workflows, Argo CD, Prometheus, Grafana, Linux, Git; Ginkgo v2 / Gomega, envtest, Vitest, JUnit

Security, Networking & ML: Zero-Trust architecture, TLS / cert-manager, NetworkPolicies, SQL-injection prevention, TCP/IP, Wireshark, Packet Tracer; TensorFlow, OpenCV, CNNs, Decision Trees, LIME / SHAP, ESP32, Arduino

Concepts: Data Structures & Algorithms, SOLID, design patterns, level-triggered reconciliation, zero trust architecture

Certifications & Training

1. **The Complete Full-Stack Web Development Bootcamp** *Dr. Angela Yu - Udemy | 62.5 Hours*
   - Comprehensive full-stack web development training covering **HTML5, CSS3, JavaScript (ES6+), React.js, Node.js, Express.js, PostgreSQL, MongoDB, RESTful APIs**, and authentication (**JWT, bcrypt**).
2. **Decoding DevOps - From Basics to Advanced Projects with AI** *Imran Teli - Udemy | 64 Hours (378 Lectures)*
   - Hands-on cloud & DevOps engineering course covering **Linux/Bash, Docker, Kubernetes, Terraform (IaC), Jenkins, GitHub Actions, AWS/GCP, Prometheus/Grafana**, and **ArgoCD (GitOps)**.
3. **React Development Course** *DP Education IT Campus | 71 Modules*
   - In-depth training on modern React frontend development: **ES6+ JavaScript prerequisites, Single Page Applications (SPAs), JSX, React Components, Props & State management, React Hooks**, and **Vite tooling**.

Extra-Curricular Activities

1. **Manager & Team Leader** 2021 – 2026
   AIESEC - University Of Peradeniya (LC Kandy) - Team Leader (Marketing), Specialist (IR), Best Marketing TL '22; Trailblazer '22 OC.
2. **Team Member** 2023
   Freshers' Hockey Team of University of Peradeniya
3. **Maths, Science Tutor** Jul 2023 – Dec 2023
   Arunella Education Welfare Programme
4. **Chief Organizer** 2018 – 2020
   Traffic Warden Unit - Ananda College

References

Dr. Suneth Namal Karunarathna
Senior Lecturer - Department of Computer Engineering
Faculty of Engineering, University of Peradeniya
namal@eng.pdn.ac.lk

Deependra Ariyadewa
Architect - WSO2
deep@wso2.com
+9471 403 5996