Ruoyu Lu

(+61)451642922 | roy.ruoyu.lu@gmail.com | Github | Linkedin | https://ruoyu.lu

Summary

Full-stack & backend developer with hands-on experience delivering event-driven, cloud-native systems in .NET, Node.js, React/Next.js, and React Native. Commercial experience across AWS and Azure, CI/CD, and microservices, owning features end-to-end from design to production. Actively seeking junior software engineer in cloud-native and product-focused teams.

Education

University of Melbourne | *Master of Engineering (Software)*	07/2023 — Present
Southwest Jiaotong University | *Bachelor of Engineering*	09/2018 — 06/2022

- Thesis: Ruoyu Lu. 2023. A hybrid parallelization approach based on workers grouping algorithm. *Applied and Computational Engineering*, 6, 340–347.

Technical Skills

- **Languages:** C#, Java, JavaScript/TypeScript, Python
- **Backend & Databases:** .NET Core, Express.js, Flask, Prisma ORM; skilled in designing and implementing RESTful APIs; experienced with SQL Server, PostgreSQL, DynamoDB, and Elasticsearch
- **Frontend Development:** React, React Native, Next.js; strong understanding of component-driven design, responsive layout, and modern UI frameworks
- **Cloud & Distributed Systems:** Hands-on experience deploying microservices on AWS (Lambda, S3, SES, DynamoDB, Cognito, CloudWatch) and Azure Functions; familiar with Kubernetes and Fission Serverless; solid grasp of event-driven and scalable system design
- **DevOps & Infrastructure:** Docker-based containerization, Terraform for IaC, CI/CD pipeline setup (GitHub Actions, Codefresh); proficient in monitoring, logging, and performance tuning
- **Tools & Workflow:** Linux, Shell scripting, Git/GitHub; Agile development (Scrum/Sprint); effective cross-platform collaboration and version control practices

Work Experience

1receipt® |*Full-Stack Developer (Part-time, previously Intern)*	07/2025 — Present

- Improved and maintained an AWS-based event-driven receipt processing pipeline, fixing message queue and concurrency issues to ensure reliable email ingestion, parsing, and data consistency across microservices.
- Implemented and optimized backend OCR and PDF parsing logic for receipt extraction and normalization, enabling robust handling of scanned and image-based receipts with improved accuracy and fault recovery.
- Developed and maintained Express.js microservices for receipt management, authentication, and analytics; containerized with Docker and deployed to AWS Elastic Beanstalk; integrated Slack-based error alerts for real-time visibility.
- Refactored parts of the React Native (Expo) mobile app, improving launch responsiveness and enhancing UI flow to deliver a smoother user experience.
- Enhanced team delivery workflows by refining multi-environment CI/CD pipelines and extending CloudWatch metrics for easier performance monitoring and debugging.

Projects

Kineo Learning Pathway Modeling and Analytics Platform

- Built as a capstone project for an enterprise client to help L&D teams visualise complex learning journeys; implemented a Next.js pathway editor so stakeholders can model Topic/Content graphs instead of maintaining spreadsheets.
- Integrated Totara LMS GraphQL and other content sources behind reusable selection components, reducing repeated integration work and letting non-technical users assemble programs from live catalog data.

- Designed and implemented Databricks-based pipelines and Azure Functions to keep learner progress data in sync across financial years, enabling consistent reporting even as source systems changed.
- Adopted an event-driven so long-running syncs would not block the UI and operational issues could be isolated instead of affecting the whole platform.
- Implemented RBAC, SAML SSO integration, and audit logging to respect multi-tenant access boundaries, and used Docker, Terraform, and GitHub Actions to ensure the environment can be reproduced and handed over cleanly.

.NET eShop Application

- Developed a modular e-commerce backend using .NET microservices architecture, separating Identity, Catalog, Ordering, and Basket domains to improve scalability and maintainability.
- Orchestrated distributed services with .NET Aspire for streamlined local development, service discovery, and dependency configuration, enabling consistent environment setup.
- Implemented asynchronous communication through an Event Bus to decouple order and basket workflows, enhancing responsiveness and fault tolerance under concurrent load.
- Integrated PostgreSQL and Redis for hybrid persistence, optimizing transaction consistency and cache performance; documented system architecture and deployment strategy for long-term maintainability.

Distributed Data Analytics Platform on Kubernetes

- Deployed Kubernetes clusters on Melbourne Research Cloud, using Fission Serverless for event-driven ingestion and serverless processing of heterogeneous data sources;
- Developed Fission functions with Python (Flask) RESTful APIs, using scheduled triggers for real-time weather/air quality data collection; designed asynchronous batch pipelines for 70k+ traffic accident data to address JSON bulk write failures;
- Optimized Elasticsearch index structures, applying multi-shard + replica strategies to improve throughput and query performance;
- Designed multi-tenant Row-Level Security (RLS) policies, ensuring isolation across cities and data sources.

Student Activity Management System

- Developed a lightweight Spring-inspired MiniFramework, implementing IoC/DI, dynamic proxies, annotation-driven inversion of control; supported RESTful controllers, service/DAO decoupling, transaction management, and lazy loading;
- Applied Data Mapper and Lazy Loading patterns to separate database operations from business logic;
- Designed multiple concurrency control strategies: • Database unique constraints + triggers to resolve concurrent RSVP insert issues; • Optimistic locking to handle concurrent modifications by club administrators.