

Fabian Mauro Rivera Morales

+51902755143 | fabianriveraabian3@gmail.com | Ate - Lima, Perú
www.fabianrivera.me | github.com/Fabian1803 | linkedin.com/in/Fabian-Rivera

Educación

Universidad Tecnológica del Perú

Egresado de Ingeniería de Software

Lima, Ate

March 2021 — December 2025

Británico

Institucion de inglés

Lima, Santa Anita

February 2026

Experiencia

AESI

August 2025 - November 2025

- Desarrollé y mantuve endpoints REST utilizando Next.js 15.3.4 (API Routes) y TypeScript 5.8, orientados al consumo desde frontend en React 19.1.0
- Implementé sistemas de paginación de artículos y búsquedas dinámicas para mejorar el rendimiento y la experiencia del usuario
- Diseñé la estructura del backend considerando escalabilidad y una futura migración a NestJS
- Integré Prisma ORM para el modelado de datos y operaciones CRUD sobre PostgreSQL
- Definé relaciones entre entidades y estructuras de tablas según requerimientos funcionales
- Configuré contenedores Docker para backend y base de datos, facilitando entornos locales reproducibles
- Implementé comunicación cliente-servidor mediante Axios 1.10.0 para el consumo de APIs
- Separé capas de rutas, lógica de negocio y acceso a datos siguiendo buenas prácticas de arquitectura
- Participé en el diseño de flujos de autenticación y manejo de estados desde el backend
- Gestioné el repositorio del proyecto usando Git y GitHub, aplicando control de versiones y organización de ramas
- Colaboré en la integración frontend-backend, resolviendo incidencias relacionadas con APIs
- Apliqué criterios de mantenibilidad y preparación para crecimiento futuro del sistema

Plataforma eCommerce

November 2025 - Actualidad

- Diseñé y desarrollé una plataforma eCommerce modular usando Next.js 16.0.10, React 19.0.0 y TypeScript, enfocada en alto rendimiento y personalización desde base de datos
- Implementé arquitectura basada en features y layouts parciales para recargar solo secciones necesarias de la aplicación
- Diseñé sistema de configuración server-side que carga estilos, carruseles, tarjetas y anuncios desde Supabase, evitando solicitudes repetidas
- Utilicé campos JSONB en PostgreSQL para almacenar dinámicamente el diseño visual del sitio
- Implementé manejo de estado global con Zustand 5.0.10 para compartir configuración y datos entre componentes
- Desarrollé dashboard administrativo para gestionar productos, categorías, marcas y configuración visual sin modificar código
- Implementé lógica flexible para habilitar o deshabilitar carrito y autenticación, permitiendo usar el sistema como eCommerce o catálogo con consultas vía WhatsApp
- Gestioné imágenes con Cloudinary 2.8.0 y carruseles dinámicos con Swiper 12.0.2
- Apliqué diseño responsive con TailwindCSS 3.4.1, Headless UI y Heroicons
- Organicé el desarrollo usando ramas en GitHub para pruebas y cambios sin afectar producción
- Preparé el proyecto para despliegue en Vercel, priorizando performance, mantenibilidad y reutilización comercial

Plataforma educativa - microservicios (Proy. universitario)

March 2025 - July 2025

- Diseñé arquitectura de microservicios para plataforma educativa utilizando FastAPI (Python 3.11), NestJS 11 y Spring Boot 3.4
- Construí API backend consumida por aplicación móvil desarrollada en React Native 0.76.9 con Expo 52
- Implementé comunicación desacoplada mediante Kong Gateway 3.6, centralizando el acceso a los servicios
- Configuré Consul 1.15 como service discovery, permitiendo registro dinámico de microservicios y detección automática de fallos
- Implementé endpoints de salud (heartbeat) en cada microservicio para monitoreo de disponibilidad
- Orquesté todos los servicios usando Docker Compose, incluyendo backend, bases de datos y Gateway
- Integré PostgreSQL 15 y MongoDB 6.0 según el tipo de datos (relacional y contenido académico)
- Diseñé microservicios independientes para usuarios, instituciones, cursos, foros y contenido educativo
- Implementé health checks automáticos para validación y reinicio de servicios

- Preparé infraestructura inicial de observabilidad con Prometheus, Grafana y Loki (entorno DevOps)
- Implementé flujo inicial de autenticación con JWT y Spring Security

LiterAlura (Alura Latam)

September 2025 - January 2026

- Desarrollé backend en Spring Boot 3.2.3 (Java 21) para consumo de API externa de libros y persistencia local
- Implementé arquitectura hexagonal separando dominio, aplicación e infraestructura
- Construí endpoints REST para búsqueda, registro y filtrado de libros
- Integré PostgreSQL mediante Spring Data JPA, gestionando entidades y repositorios
- Implementé lógica para evitar duplicados y mantener consistencia de datos
- Configuré variables de entorno (.env) para manejo seguro de credenciales
- Desarrollé frontend ligero en Preact / React 19 para visualizar libros almacenados
- Integré frontend con backend usando Axios
- Implementé flujo completo: API externa - backend - base de datos - frontend – Versioné el proyecto con Git y GitHub

Freelance - sitios corporativo

August 2024 - March 2025

- Implementé prácticas básicas de SEO on-page (estructura, metadata, performance)
- Preparé sitios para campañas de Google Ads, optimizando velocidad de carga y experiencia móvil
- Personalicé temas y layouts según requerimientos del cliente
- Gestioné despliegue, dominio y configuración inicial de hosting
- Trabajé directamente con clientes, levantando requerimientos y entregando soluciones funcionales

Skills

- Languages and frameworks: Java (8, 17, 21), Spring Boot, Spring Web, Spring Data JPA, Spring Security (JWT), JPA / Hibernate, Lombok, NestJS (11), Node.js, TypeScript (5.x), REST APIs, TypeORM, FastAPI (Python 3.11), Uvicorn, Pydantic, SQLAlchemy, JavaScript ES2023, React, Next.js, Next API Routes, Prisma, Zustand (5.0.10), TailwindCSS, Angular (19), RxJS
- Prácticas de Desarrollo de Software: Pruebas unitarias con JUnit y Mockito, Testcontainers para testing con dependencias reales, revisión de código, análisis estático (SonarQube), seguridad en pipelines (SAST / DAST), principios básicos de Clean Code, integración continua (CI), control de calidad automatizado
- Patrones y Arquitectura: MVC, arquitectura por capas (Controller–Service–Repository), arquitectura modular por features, microservicios, API Gateway, separación dominio/infraestructura (enfoque hexagonal), diseño cliente–servidor
- APIs e Integración: Diseño e implementación de APIs REST (JSON), consumo de APIs externas, documentación básica con OpenAPI/Swagger, integración frontend–backend, arquitectura cliente–servidor, autenticación con JWT
- Cloud y Plataformas: Google Cloud Platform (Compute Engine, despliegue de aplicaciones), AWS (EC2, servicios básicos), Azure (conocimientos académicos), despliegue de aplicaciones backend, uso de máquinas virtuales, gestión básica de bases de datos en la nube
- DevOps y CI/CD: Docker, Docker Compose, GitHub Actions, GitLab CI/CD (conocimientos), pipelines básicos, despliegue manual y automatizado, gestión de variables de entorno, SAST/DAST inicial, SonarQube, OWASP ZAP, ZNIK
- Control de Versiones: Git, GitHub, GitLab (flujo feature branches, pull requests, revisión de código) Bases de Datos: PostgreSQL, MongoDB, Supabase, modelado relacional, consultas SQL, integración con ORM (JPA/Hibernate, Prisma)
- Metodologías y Trabajo Colaborativo: Scrum (entorno académico), trabajo por sprints, planificación básica, resolución de problemas, adaptación a proyectos existentes, comunicación técnica