

FREDY OSWALDO LÓPEZ DAZA

Estudiante de Ingeniería de Sistemas | Desarrollo de Software Full-Stack

+57 321 644 4954 | lopezdazafredy@gmail.com ·

github.com/fredylopez01 | www.linkedin.com/in/fredy-oswaldo-lopez-daza

PERFIL PROFESIONAL

Estudiante de Ingeniería de Sistemas y Computación, con experiencia práctica en desarrollo de aplicaciones Full-Stack y diseño de servicios backend. He trabajado principalmente con Java/Spring Boot y Node.js/NestJS, desarrollando APIs REST, sistemas de autenticación, aplicaciones en tiempo real e integración con bases de datos relacionales y NoSQL, además de Nextjs y React en el fronted. Cuento con experiencia académica en arquitecturas modulares, microservicios, Clean Architecture, comunicación asíncrona y sistemas distribuidos, además de herramientas de contenerización, observabilidad y CI/CD. Familiarizado con Git, GitHub y flujos de trabajo Gitflow, así como con metodologías ágiles como Scrum. Me caracterizo por el aprendizaje continuo, la capacidad para analizar problemas técnicos y la disposición para trabajar colaborativamente, asumir responsabilidades y adaptarme a nuevas tecnologías y entornos de desarrollo.

FORMACIÓN ACADÉMICA

Ingeniería de Sistemas y Computación — En curso

Universidad Pedagógica y Tecnológica de Colombia (UPTC) · 2022 – 2027

PROYECTOS RELEVANTES

Sistema de Blackjack Distribuido

Nov. 2025 | [repositorio](#)

- Diseñé una arquitectura distribuida compuesta por cuatro servicios backend independientes, utilizando RabbitMQ para comunicación asíncrona y desacoplamiento de procesos.
- Implementé comunicación en tiempo real mediante WebSockets (Socket.IO) para gestionar sesiones multijugador y sincronización de eventos entre clientes.
- Configuré NGINX como balanceador de carga y mecanismos de redundancia orientados a mejorar la disponibilidad del sistema ante fallos.
- Integré Prometheus y Grafana para monitoreo de métricas, health checks y observabilidad de los servicios.

Sistema de Parqueaderos Cliente-Servidor

Sep. – Nov. | [repositorio backend](#) | [repositorio frontend](#) | [demo](#)

- Desarrollé una aplicación Full-Stack con API REST, autenticación mediante JWT y autorización basada en roles.
- Implementé una arquitectura por capas, separando controladores, servicios y acceso a datos para facilitar el mantenimiento y las pruebas.
- Gestioné la persistencia mediante Prisma ORM, incluyendo migraciones, validación y manejo estructurado de errores.
- Desarrollé el frontend con Next.js y React, integrándolo con el backend mediante APIs REST y Axios.

Sistema Integral de Gestión de Gimnasios

Jun. 2025 – Actualidad

- Diseño y desarrollo de una aplicación Full-Stack orientada a la gestión integral de gimnasios.
- Implementé servicios backend con Spring Boot, JPA/Hibernate y JWT, estructurados para facilitar la incorporación de nuevos módulos.
- Diseñé una estructura modular preparada para funcionalidades como gestión de pagos, nutrición, informes y administración.

- Desarrollé el frontend con Next.js y React, integrando los flujos de autenticación y consumo de servicios REST.

Aviator — Sistema Distribuido en Tiempo Real

2025 | [Repositorio](#)

- Desarrollé un sistema distribuido capaz de gestionar múltiples usuarios concurrentes mediante comunicación en tiempo real con WebSockets.
- Implementé mecanismos de replicación de PostgreSQL y failover orientados a mejorar la disponibilidad y resiliencia del sistema.
- Desarrollé la lógica backend para programación de eventos, cálculo de resultados y distribución del estado en tiempo real a los clientes conectados.

FIFA World Cup 2026 — Attendance Predictor

2026 | [Repositorio API](#) | [demo](#)

- Desarrollé un sistema de Machine Learning para predecir la asistencia y tasa de ocupación de partidos del Mundial 2026 a partir de datos históricos de 1930–2014.
- Implementé modelos de aprendizaje supervisado y no supervisado mediante Random Forest con GridSearchCV y K-Means, incluyendo selección de hiperparámetros y evaluación de clusters.
- Construí una API REST con Flask siguiendo Clean Architecture, separando dominio, casos de uso, infraestructura e interfaces.
- Desarrollé una SPA con Next.js y Tailwind CSS para consumir las predicciones y visualizar información del modelo.
- Implementé endpoints para predicciones, metadatos, clusters y health checks, además de registro de predicciones para facilitar la trazabilidad.

HABILIDADES TÉCNICAS

Lenguajes: Java, JavaScript (ES6+), TypeScript, Python

Backend: Spring Boot, Node.js, NestJS, Express.js, Flask, APIs REST, JWT, WebSockets / Socket.IO

Frontend: Next.js, React, Tailwind CSS, Axios, HTML5, CSS3

Bases de datos: PostgreSQL, MySQL, Oracle Database, MongoDB, Redis

ORM y persistencia: JPA, Hibernate, Prisma

Arquitectura y diseño: Arquitectura por capas, Arquitectura modular, Clean Architecture, Microservicios, Sistemas distribuidos, Arquitecturas orientadas a eventos, Comunicación asíncrona

Resiliencia y datos: Tolerancia a fallos, Redundancia, Failover, Replicación de bases de datos

Mensajería: RabbitMQ, Apache Kafka, Workers / Background Jobs

DevOps y control de versiones: Docker, Docker Compose, NGINX, Git, GitHub, Gitflow, GitHub Actions, CI/CD

Observabilidad: Prometheus, Grafana, Health Checks

Metodologías y prácticas: Agile, Scrum, Desarrollo iterativo, Pull Requests, Code Review

IDIOMAS

Español: Nativo | Inglés: B2 – Intermedio