

**Gerónimo Giordano**  
**Software Engineer | Backend & DevOps**

Mendoza, Argentina • [github.com/gerogiordano08](https://github.com/gerogiordano08) • <https://gerogiordano08.github.io/Landing-Page/> •  
gerogiordano08@gmail.com • 2612507730 • [traderegistry.tech](https://traderegistry.tech)

Estudiante de Ingeniería en Informática y desarrollador de software enfocado en el desarrollo de arquitecturas backend escalables y despliegues cloud-native, con un enfoque proactivo en la seguridad de sistemas.

## EDUCACIÓN

### Universidad de Mendoza

Mendoza, Argentina

Ingeniería en Informática

Finalización prevista: 2028

Materias destacadas: Sistemas Operativos, Redes de Datos, Diseño de Bases de Datos, Arquitectura de Computadoras, Análisis de Sistemas, Matemática Discreta.

**Idiomas:** Inglés (C1 - Avanzado) , Español Nativo

## CONOCIMIENTOS TÉCNICOS

- **Lenguajes & Backend:** Python (Django, Pygame, Pytest), SQL (PostgreSQL), JavaScript.
- **Cloud & DevOps:** Oracle Cloud Infrastructure (OCI), Docker, Docker Compose, Git, GitHub, Nginx, Gunicorn, Linux (Ubuntu/Debian).
- **Bases de datos & Caching:** PostgreSQL, Redis.
- **Automatización:** Cron Jobs, Django Management Commands, Shell Scripting.
- **Frontend:** Bootstrap 5, HTML5, CSS3

## PROYECTOS TÉCNICOS

*The Trade Registry | Lead Full-Stack Developer & DevOps | Julio, 2025 – Present*

- Diseñé la arquitectura de un dashboard financiero con Django y Bootstrap 5, integrando un esquema relacional en PostgreSQL y APIs RESTful para procesar y analizar datos de mercado en tiempo real.
- Desplugué un entorno de producción contenedorizado en Oracle Cloud (OCI) utilizando Docker y Docker Compose, logrando un 100% de paridad entre entornos mediante builds seguras de múltiples etapas.
- Fortalecí la seguridad y el rendimiento del sistema mediante la implementación de un honeypot personalizado, blacklisting de IPs automatizado y caching con Redis para optimizar la recuperación de datos y minimizar la latencia de la API.

*Backgammon Engine | Lead Software Engineer | Aug 25, 2025 - Nov 1, 2025*

- Desarrollé un motor de juego modular en Python utilizando un patrón Controller-Core para desacoplar la lógica principal de las interfaces CLI y Pygame.
- Alcancé un 90% de cobertura de código mediante Unittest, garantizando la fiabilidad del sistema a través de pruebas rigurosas de mecánicas de juego complejas y transiciones de estado.
- Implementé Redis como una capa de persistencia de alto rendimiento para el guardado automático del estado del juego y la recuperación fluida de sesiones.