

# João Pedro Camargo Pinheiro

Desenvolvedor de Software | Backend

Rio de Janeiro • Email: joaoppinheiro2005@gmail.com • linkedin.com/in/joao-pedro-p/ • github.com/j-ordep

## RESUMO PROFISSIONAL

Desenvolvedor Backend com experiência prática em Golang, atuando no desenvolvimento de APIs REST, integrações com PostgreSQL, autenticação e autorização com Keycloak/JWT e arquiteturas desacopladas. Atuação em projetos desde a fase inicial, com foco em qualidade de código, testes automatizados, segurança, boas práticas (SOLID / Clean Code) e entrega contínua em ambientes ágeis.

## Experiência

### 3CON Consultoria e Sistemas

Remoto

Desenvolvedor Backend (Aprendiz) / Golang

Jun 2025 – Atual

- Atuação no **desenvolvimento backend** desde a fase inicial de projetos para um dos **maiores varejistas farmacêuticos do país**, construindo **APIs REST** em **Golang** com o framework Echo.
- Atuo como **responsável** pelo desenvolvimento da **API Backend em Golang** de um projeto interno de **Gestão de Licenças de Produtos e Sistemas**, desde a modelagem até a integração com **APIs externas e banco de dados PostgreSQL**, sob supervisão direta do **Tech Lead**.
- Implementação de **autenticação e autorização com Keycloak (IAM)** e **JWT**, padronizando o controle de acesso à API.
- Criação de **testes unitários** e de **integração** utilizando **Go testing** e **SQLMock**, alcançando **91%** de cobertura de código.
- Uso contínuo do **SonarQube** para análise de **qualidade, cobertura de testes** e identificação de melhorias no código **Go**.
- Implementação de armazenamento de arquivos e imagens utilizando **Bucket S3 (MinIO)**.
- Definição e implementação de fluxos de negócio relacionados ao controle, validação e gerenciamento de licenças, garantindo consistência de dados e segurança.
- Participação ativa no levantamento de requisitos, refatorações e adaptações do sistema conforme necessidades do negócio, seguindo metodologias ágeis (**Jira / Kanban**).

## Habilidades

- Backend:** Golang, Java, Spring Boot, PostgreSQL, JWT, Keycloak, Microsserviços
- Ferramentas & Infra:** Git, Docker, Kafka, SonarQube, Swagger, MinIO
- Boas Práticas:** SOLID, Testes Unitários e de Integração
- Idiomas:** Português (Nativo), Inglês (intermediário)

## Educação

### Centro Universitário UniCarioca

Rio de Janeiro

Análise e Desenvolvimento de Sistemas

Conclusão Dez 2026

## Cursos

### Trilha Senior

Remoto

Dev Poliglota - Victor Alisson (Senior Staff Software Engineer)

Jan 2026

- Arquitetura & System Design:** Monólito, Monólito Modular, Microsserviços, Event-Driven, CQRS).
- Linguagens de programação:** Go, Java, Kotlin, Rust, Elixir, TypeScript.
- Paradigmas:** Programação Orientada a Objetos, Programação funcional, Procedural.

- **Design de Software:** Essência de um Bom Design, Reconhecendo Smells, além da visão de grandes autores como Martin Fowler, Gerald Jay, DHH, José Valim, Sam Newman.

## Java10x

Remoto

*Java/Spring Boot - Horacio Mota (Developer @DigitalOcean)*

*Nov 2024 – Dez 2025*

- Java, Spring Boot, JPA/Hibernate, orientada a objetos (POO), Lombok, multithreading, testes unitários Mockito, JUnit,
- Estruturas de dados (listas, filas, pilhas, mapas)
- Complexidade de algoritmos, Big O.

## Projetos Pessoais

---

### Gateway de Pagamento

(<https://github.com/j-ordep/Go-Gateway>)

*Golang - NestJS - Apache Kafka - Docker - REST - Event-Driven Architecture*

- Desenvolvimento de um **Gateway de Pagamentos em Go**, responsável pelo processamento de transações, gerenciamento de contas e publicação de eventos de pagamento.
- Arquitetura baseada em **microserviços** com comunicação assíncrona via **Kafka**.
- Construção do serviço **Antifraude** em **NestJS**, consumindo mensagens do Kafka.
- **Docker e Docker Compose** para orquestração dos serviços (Gateway, Kafka, Antifraude).

### API de Gestão de Eventos - Java

(<https://github.com/j-ordep/Event-Flow>)

*Java - Spring Boot - JPA/Hibernate - Docker - REST - MySQL*

- Desenvolvimento de API RESTful em Java e Spring Boot para gerenciamento de eventos, inscrições e convites.
- Arquitetura **MVC**, CRUD completo utilizando **Java e Spring Boot + Spring Data JPA com MySQL e docker**.