Gabriel Afonso

Desenvolvedor Backend

Email: gabriellafonso.dev@gmail.com
Telefone: (27) 99245-3811
Cidade: Cariacica - ES

Portfólio: gabrielafonso.com.br
GitHub: github.com/GabriellAfonso
Linkedin: gabriellafonso

Resumo Profissional

Desenvolvedor backend com 2 anos de experiência prática, focado em Python e Django. Atuo de ponta a ponta (API, infra e front) com ênfase em backend, aplicando boas práticas de arquitetura, Docker e processamento assíncrono em projetos reais.

Projetos

IPB Castelo Branco – API RESTful de gestão para igreja local | Link do projeto

API REST completa para gestão de membros, escala mensal, repertório musical e galeria de fotos, utilizada em app publicado na Play Store.

- Reduz em ~60% o tempo de organização do repertório do culto, de ~20 minutos para ~5 minutos
- Autenticação com JWT e login social via Google OAuth 2.0
- Documentação automática com drf-spectacular (Swagger / ReDoc)
- Arquitetura em camadas (domain, application, infrastructure) com dependency-injector
- Servidor ASGI com Gunicorn + Uvicorn, containerizado com Docker Compose

Shorter – Encurtador de URLs | Link do projeto

Encurtador de URLs com foco em performance e analytics

- Implementei cache de redirecionamentos com Redis, reduzindo a latência em ~55% — de ~23ms para ~10ms — sustentando 40 req/s com zero erro em 1 vCPU
- IPs dos visitantes anonimizados antes de persistir, sem comprometer o analytics
- Registro de cliques em segundo plano com reenvio automático em caso de falha, sem perda de dados
- Analytics por link com breakdown de dispositivos, origens de tráfego e histórico diário de cliques

DumCrown – Jogo de cartas online | Link do projeto

Inspirado em Legends of Runeterra, com partidas em tempo real.

- Lógica de jogo implementada do zero: sistema de turnos, fases de combate e mecânica de energia
- Matchmaking assíncrono com fila thread-safe e sistema de salas com transferência de dono
- Comunicação em tempo real via Django Channels & WebSockets com rate limiting por ação
- Histórico completo de partidas persistido ao fim de cada jogo

Tecnologias

Python, Django, Django REST Framework, Celery, Redis, PostgreSQL, Docker, Nginx, Git / GitHub, WebSockets, JavaScript, TypeScript, Kotlin, C#