

Documentação Técnica – ParkControl

Visão Geral

ParkControl é um sistema de gestão de estacionamento desenvolvido em **Django**, utilizando **Docker** para padronizar o ambiente de desenvolvimento. Isso evita a necessidade de configurar manualmente o Python, banco de dados e dependências em cada máquina.

Pré-requisitos

- Docker instalado
 - Git instalado
-

Como Rodar o Projeto

Clone o repositório:

```
git clone https://github.com/ifmt-cba/parkcontrol.git
cd parkcontrol
```

Suba os containers Docker:

```
docker-compose up --build
```

1. Isso irá:
 - Criar a imagem Docker
 - Instalar dependências do `requirements.txt`
 - Iniciar o servidor Django na porta **8000**
-

Acessando o Sistema

Após o build, acesse pelo navegador:

http://localhost:8000

Estrutura Docker

- **Dockerfile:** Define a imagem base (Python), dependências e execução.
 - **docker-compose.yml:** Define o serviço web, porta, volumes e variáveis de ambiente.
 - **.env:** Contém variáveis sensíveis como **SECRET_KEY** (não subir para o Git).
-

Estrutura de Pastas

```
parkcontrol/
├── apps/
│   ├── usuarios/      # Autenticação e gestão de usuários
│   ├── frentistas/    # Entrada/saída, cobranças
│   ├── clientes/      # Gestão de mensalistas e diaristas
│   ├── planos/        # CRUD de planos
│   ├── pagamentos/    # Registro de cobranças
│   ├── manutencao/    # Pedidos de manutenção
│   ├── relatorios/    # Relatórios financeiros e de vagas
│   ├── vagas/         # Gestão das vagas
│   └── core/          # Utilitários, mixins, modelos base
├── manage.py
├── parkcontrol/        # Configurações principais do Django
├── requirements.txt
├── .env
├── .dockerignore
├── .gitignore
├── Dockerfile
├── docker-compose.yml
├── README.md
└── docs/              # Documentação do projeto (BPMN, Casos de Uso etc.)
```

Comandos Úteis

Acessar o container:

```
docker exec -it parkcontrol_web bash
```

-

Executar migrações:

```
docker exec -it parkcontrol_web python parkcontrol/manage.py migrate
```

-

Criar superusuário:

```
docker exec -it parkcontrol_web python parkcontrol/manage.py createsuperuser
```

-

Colaboração

Crie uma nova branch por funcionalidade, nomeando conforme o app afetado.

Exemplo:

```
git checkout -b feature/frentista-tela-inicial
```

Variáveis de Ambiente

No arquivo `.env` (já listado no `.gitignore`):

```
SECRET_KEY=your-secret-key  
DEBUG=True
```

Atualizando o Docker com Alterações

1. Salve suas alterações no código.

Se o Docker estiver rodando, pare-o:

```
docker-compose down
```

2.

Rebuild da imagem com as alterações:

```
docker-compose up --build
```

3. O parâmetro `--build` é necessário sempre que você altera algo no código ou no Dockerfile.

(Opcional) Execute comandos dentro do container se necessário:

```
docker exec -it parkcontrol_web python parkcontrol/manage.py migrate  
docker exec -it parkcontrol_web python parkcontrol/manage.py createsuperuser
```