

🐾 SaquaDot – Adoção Responsável em Saquarema

Um sistema web desenvolvido para facilitar o processo de **adoção responsável de animais**, conectando cuidadores e adotantes de forma simples, segura e intuitiva.

Transformando vidas, uma adoção de cada vez.



🐾 SaquaDot – Adoção
Responsável em Saquarema

Preparado por:

Heloisa Alice Silva de Oliveira

Karlos Eduardo Marinho de Sá

Leonardo dos Santos Velloso

Paulo Sérgio Nascimento dos Santos

Rafael Farias Coelho





Sumário do Projeto

01

Visão Geral do SaquaDot

Entenda o propósito e o contexto do projeto.

02

Funcionalidades Essenciais

Detalhes sobre o que o sistema oferece para adotantes e cuidadores.

03

Tecnologias Utilizadas

O stack técnico por trás do desenvolvimento (Django, Bootstrap).

04

Instalação e Configuração

Passos para executar o projeto em um ambiente local.

05

Estrutura do Projeto

Organização dos arquivos e módulos do SaquaDot.

06

O Futuro do SaquaDot

Recursos planejados para o próximo ciclo de desenvolvimento.

💡 Visão Geral: Promovendo a Adoção em Saquarema

O SaquaDot é uma plataforma web dedicada à causa animal em Saquarema – RJ. Nosso foco é facilitar a adoção responsável de cães e gatos, atuando como uma ponte digital que conecta de forma transparente e segura os animais resgatados aos seus futuros lares.

- Conexão direta entre cuidadores (protetores, ONGs) e adotantes.
- Prioridade na segurança e no bem-estar animal em cada processo.
- Interface simples e intuitiva para uma experiência acolhedora.



- Desenvolvido com **Django (Python)** e **Bootstrap**, o sistema é responsivo e otimizado para acesso em qualquer dispositivo.

✨ Principais Funcionalidades

1

Experiência do Adotante

Para quem deseja dar um novo lar a um animal:

- Cadastro e autenticação de usuário.
- Visualização e filtragem avançada de animais disponíveis (espécie, idade).
- Solicitação de adoção e acompanhamento do status em tempo real.

2

Ferramentas do Cuidador

Para protetores e voluntários:

- Cadastro completo de novos animais (fotos, descrição, dados de contato).
- Gerenciamento e atualização de seus próprios animais cadastrados.
- Recebimento, aprovação ou recusa de solicitações de adoção.



Sistema de Notificações

Alertas em tempo real sobre novos pedidos, atualizações de status e mensagens. Inclui contador de notificações não lidas para manter o usuário sempre informado.



Interface Amigável

Design moderno, utilizando Bootstrap 5, com ícones personalizados e feedback visual suave ao interagir com a plataforma, garantindo usabilidade.

Nota sobre Perfis: Atualmente, a plataforma opera com uma conta unificada, permitindo que o mesmo usuário atue tanto como Adotante quanto como Cuidador.



Tecnologias Utilizadas no Desenvolvimento

O SaquaDot foi construído utilizando um conjunto de ferramentas robustas e modernas para garantir performance, escalabilidade e uma ótima experiência de usuário.

Back-end	Python 3.x, Django 5 (framework web de alto nível)
Front-end	HTML5, CSS3, JavaScript, Bootstrap 5 (para design responsivo)
Banco de Dados	SQLite (ideal para desenvolvimento e testes locais)
Controle de Versão	Git e GitHub (para colaboração e histórico de código)
Ferramentas	Visual Studio Code (IDE), Figma (prototipagem), Git Bash

Instalação e Configuração Local

Siga os passos abaixo para preparar seu ambiente e rodar o projeto SaquaDot em sua máquina.



Pré-requisitos de Software

Certifique-se de que as seguintes ferramentas essenciais estão instaladas:

- **Python 3.x** (Versão mais recente)
- **Git** (Sistema de controle de versão)
- **pip** (Gerenciador de pacotes do Python)



Obtenção do Código

Utilize o Git para clonar o repositório oficial do projeto SaquaDot:

```
git clone  
https://github.com/Sergio003  
/Saquadot.git
```

```
cd saquadot
```



Ambiente Virtual

Crie e ative um ambiente virtual para isolar as dependências do projeto:

```
python -m venv .venv
```

Ativação (Windows):

```
.\.venv\Scripts\activate
```

Passos para Executar o Projeto

1 Instalar as Dependências

Baixe o Django e a biblioteca Pillow (necessária para upload e manipulação de imagens de animais):

```
pip install django, pip install Pillow e pip install django-tinymce
```

2 Configurar Superusuário (Opcional)

Crie uma conta de administrador para acessar o painel de gerenciamento do Django (admin):

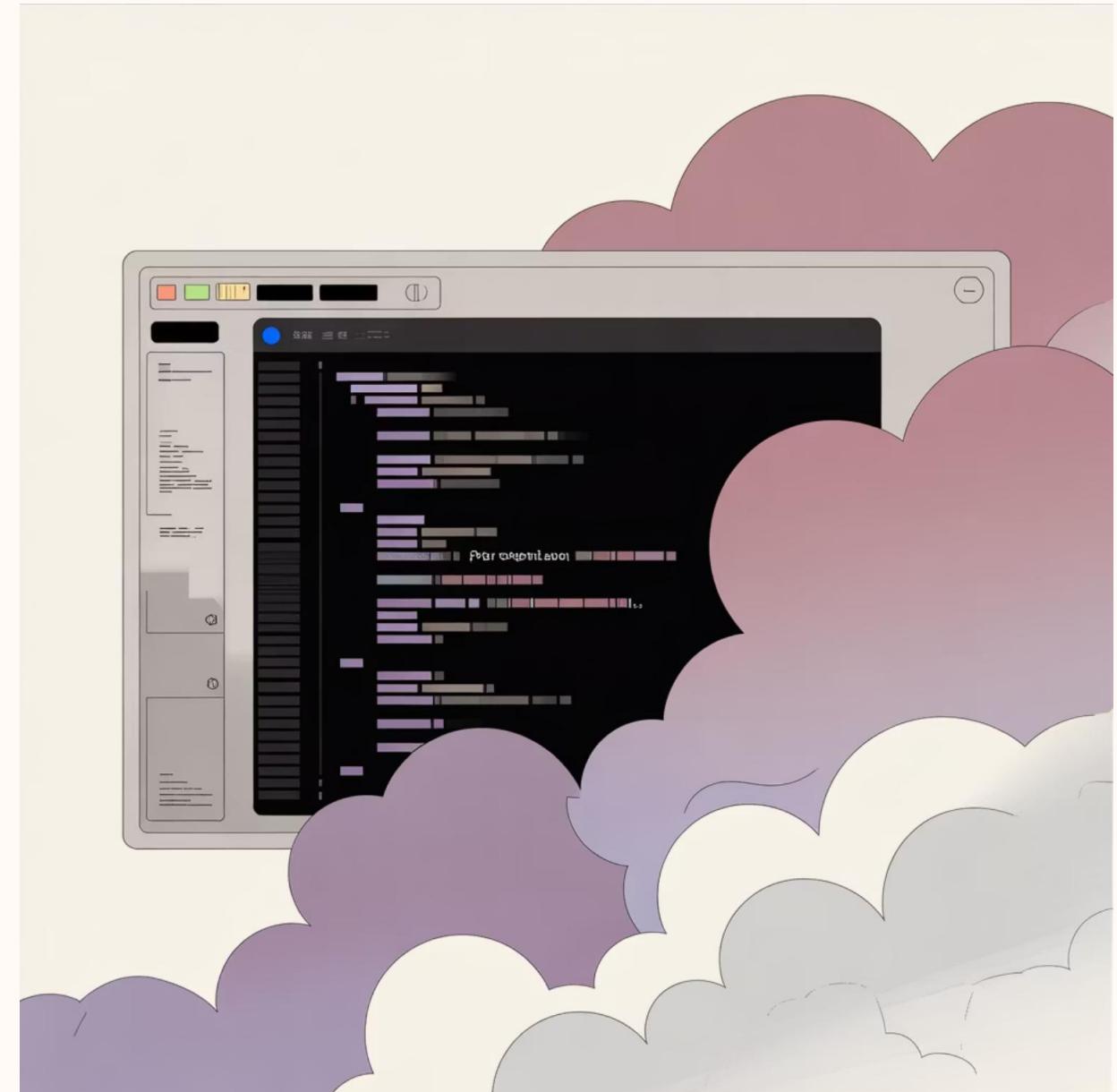
```
python manage.py createsuperuser
```

3 Executar o Servidor Local

Inicie o servidor de desenvolvimento para acessar o SaquaDot:

```
python manage.py runserver
```

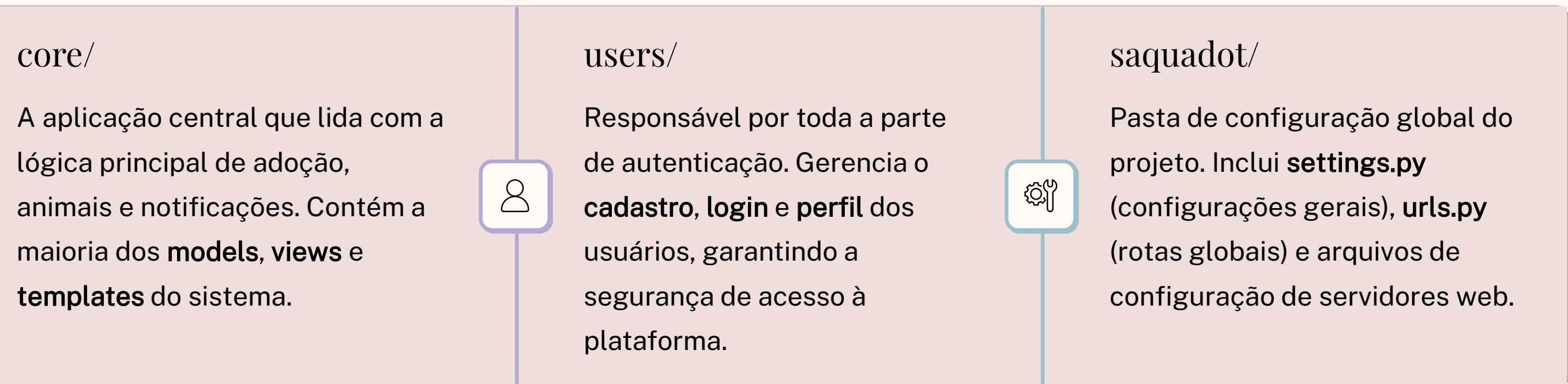
O sistema estará disponível no endereço: <http://127.0.0.1:8000/>





Estrutura Detalhada do Projeto

O SaquaDot é organizado em três aplicações principais, seguindo a arquitetura modular do Django, para garantir clareza e manutenção.



Outros componentes importantes incluem a pasta **media/** (para imagens dos animais) e **db.sqlite3** (o banco de dados local).



Rumo ao Futuro: Próximas Implementações

Para a próxima fase de desenvolvimento, o SaquaDot planeja adicionar recursos que aprimorarão a interação e a capacidade de gerenciamento da plataforma.



Chat Direto

Implementação de um sistema de **chat em tempo real** para comunicação segura e eficiente entre cuidadores e adotantes interessados.



Múltiplas Fotos

Permitir o upload de **múltiplas fotos** por animal cadastrado, oferecendo uma visão mais completa e atrativa para os adotantes.



Filtros Avançados

Expansão das opções de busca, incluindo **porte, raça, e localização** dentro de Saquarema, para facilitar a compatibilidade.



Campanhas para ONGs

Funcionalidade para ONGs parceiras cadastrarem e gerenciarem suas próprias **campanhas de arrecadação e adoção** diretamente na plataforma.

❤️ Desenvolvido com Paixão e Propósito

O SaquaDot é mais que um projeto acadêmico; é uma iniciativa movida pelo amor e dedicação à causa animal em Saquarema. Agradecemos a todos que apoiam a adoção responsável!



Adote, Não Compre.

Equipe SaquaDot

Este projeto foi desenvolvido com foco em tecnologia e responsabilidade social, unindo o poder do Python/Django com o desejo de impactar positivamente a comunidade local.

