



Framework Django – Overview



Estudantes: Gustavo Henrique de Barros Mestre;
Ana Flavia Perin.



01 O QUE É?

- Baseado em Python;
- Código Aberto;
- 7º framework mais utilizado;
- 1º framework de Python mais utilizado.



RESPONSÁVEIS PELO DESENVOLVIMENTO



COMUNIDADE

Em 2005, o World Online, tornou-se Django e um código aberto










Simon Willison



Adrian Holovaty

PRINCIPAIS CARACTERÍSTICAS

CARACTERÍSTICAS	DESCRIÇÃO
 Desenvolvimento Ágil	Minimizar linhas de código; FOCO na lógica.
 Segurança	Proteção contra ataques WEB comuns.
 Escalabilidade	Capacidade de lidar com volumes altos de tráfego.
 Abstração	Interface intuitiva e simplificada.
 Documentação	Muitas discussões e documentações.
 Padrão MTV	Model-Template-View.
 Integração	Funcionalidades prontas.



CONFIGURAÇÕES INICIAIS NECESSÁRIAS

A	Instalar o Python pelo site oficial;
B	Instalar o framework Django pelo gerenciador de pacotes pip;
C	Configurar o ambiente virtual, separando as dependências;

D	Criar um projeto Django;
F	Realizar configurações iniciais;
G	Configurar Banco de Dados;

Servidores Web Disponíveis

- WSGI: Interface padrão entre servidores web e aplicações ou frameworks web Python.

APACHE

- Open-source;
- Lida com altos níveis de tráfego;
- Portabilidade;
- Concorrência.

Gunicorn + Nginx

- Gunicorn: Atua como WSGI;
- Nginx: Proxy reverso;

Tipo de Licença: 3-cláusula BSD

- É gratuito e código-aberto;
- Permite a livre utilização, modificação e distribuição do software;
- Incluir direitos autorais, condições e avisos legais;
- Proibição do uso de nomes de autores e colaboradores para promoção comercial sem permissão escrita.





Vantagens:

Velocidade


Desenvolvimento rápido com configurações pré-definidas e estrutura MTV, ideal para startups.

Grande comunidade de suporte

Amplo suporte devido à sua base em Python e a vasta comunidade de usuários e desenvolvedores.

Segurança

Proteções integradas contra ameaças comuns, como XSS e SQL injection, com configuração prévia para economizar tempo.



Desvantagens:

Inadequado para projetos simples

Pode ser demasiado complexo para projetos pequenos, sendo Flask uma opção mais ágil e escalável.

Pode levar a sites lentos

O uso intenso de funcionalidades pode gerar sites lentos e dificultar a otimização, especialmente para novatos.

Falta de convenção

Não adere estritamente à convenção sobre configuração, o que pode resultar em progresso mais lento e descontentamento de desenvolvedores.

QUEM ESTÁ USANDO?



Considerações Finais:

01

Eficiência e Reutilização de código

Views genéricas e herança de templates.

03

Padronizações importantes

Estrutura inicial rápida e padrão.

02

Facilidade de encontrar material

Fóruns, comunidades, documentação bem construída, etc.

04

Estrutura "engessada"

Pode limitar a flexibilidade para projetos específicos

05

Simplifica o processo de execução

Back-end e front-end integrados em um único servidor.

