




FRAMEWORKS PARA DESENVOLVIMENTO WEB

Professor Paulo Honório
Primeiro Dia - 11/02/2023



Agenda

- Apresentação
- Motivação
- O que está por vir?
- Python versão 3
- Relevância da linguagem
- Frameworks em Python
- Ambientes virtuais
- Atividade 1
- MVC
- Estrutura da aplicação no Django
- Migração da base de dados
- Django-admin
- Atividade 2
- Estudo de Caso
- Django Auth
- Container Docker
- Atividade 3
- Questionário

Apresentação

■ Paulo Honório Filho

- *Doutorando no Programa de Pós-Graduação em Engenharia Elétrica na UFC*
- *Pesquisador no Laboratório de Condicionadores de Energia – LCE*
- *Lattes: <https://lattes.cnpq.br/7658949372926009>*
- *LinkedIn: <https://www.linkedin.com/in/paulo-hon%C3%B3rio-filho-4827236b/>*



DESENVOLVIMENTO WEB

Motivação

WEB 3.0 & new marketing world



Web 1.0

"Read Only",
Decentralized



Web 2.0

Participatory,
Centralized

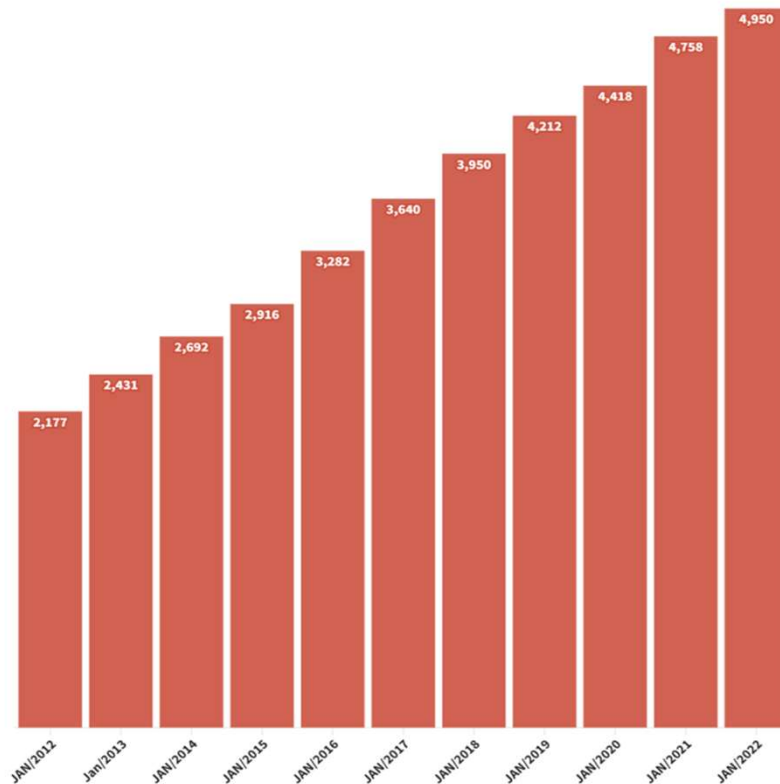


Web 3

No Intermediaries,
Decentralized

EVOLUÇÃO DO NÚMERO DE USUÁRIOS ATIVOS DE INTERNET

Em dez anos, número de internautas dobra no mundo (em bilhões)

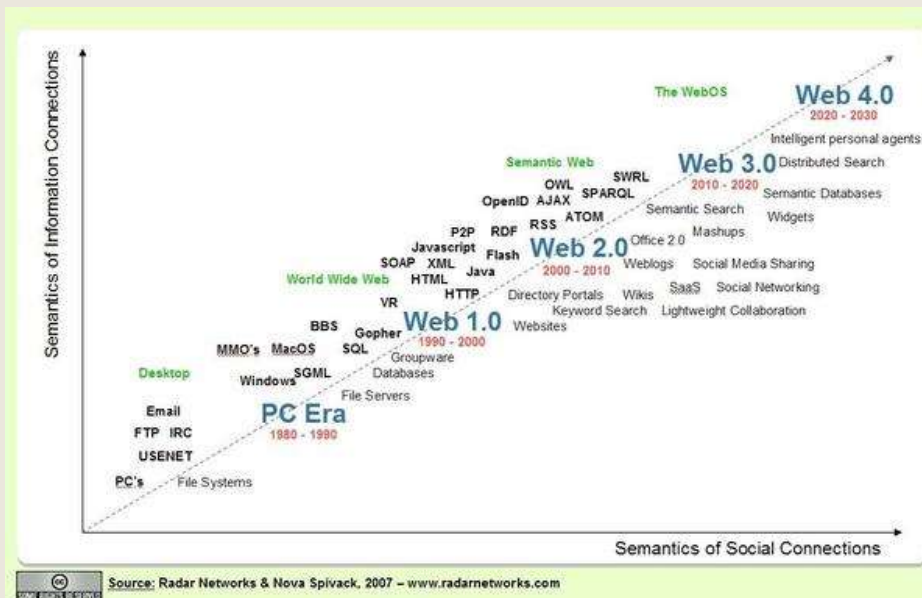


Fonte: Datareportal.com (Digital 2022: Global Overview Report)

Motivação

- “Mundo se aproxima da marca de 5 bilhões de usuários de internet, 63% da população.” [Insper, matéria publicada em fevereiro de 2022](#)

O que está por vir?



- [Nova Spivack - Semantic Web Talk 2008](#)
- A IA está se tornando mais significativa em muitos elementos de planejamento e design de paisagens urbanas com o advento da web 4.0 e da computação centrada no ser humano. [CHEN -2023]
- Algo semelhante acontecerá novamente em breve, quando a Web 4.0 oferecer novas oportunidades resultantes de uma interação simbiótica mais profunda entre homem e máquina, incluindo trocas emocionais entre eles. Embora relevante, esse cenário ainda tem um longo caminho a percorrer antes de ser alcançado em larga escala. [FARUK-2022]

Python Versão 3

Versão	Lançamento	Principais características e atualizações
Python 3.0	2008	Correções gerais e reorganização na biblioteca-padrão
Python 2.6	2006	Inclusão de bibliotecas para multiprocessing
Python 2.4	2004	Adição de decoradores de função/método
Python 2.3	2003	Módulos novos e melhorias de type/class system
Python 2.2	2001	Adição do generator e iterators
Python 2.1	2001	A licença foi renomeada para Python Software Foundation License
Python 2.0	2000	Correção e implementação de list comprehension
Python 1.0	1994	Inclusão de ferramentas para programação funcional – como lambda, map, filter e reduce, por exemplo
Python 0.9.0	1991	Primeira versão, lançada por Guido Van

Relevância da linguagem



Frameworks em Python

Em geral são separados em três tipos:

- Full Stack Framework

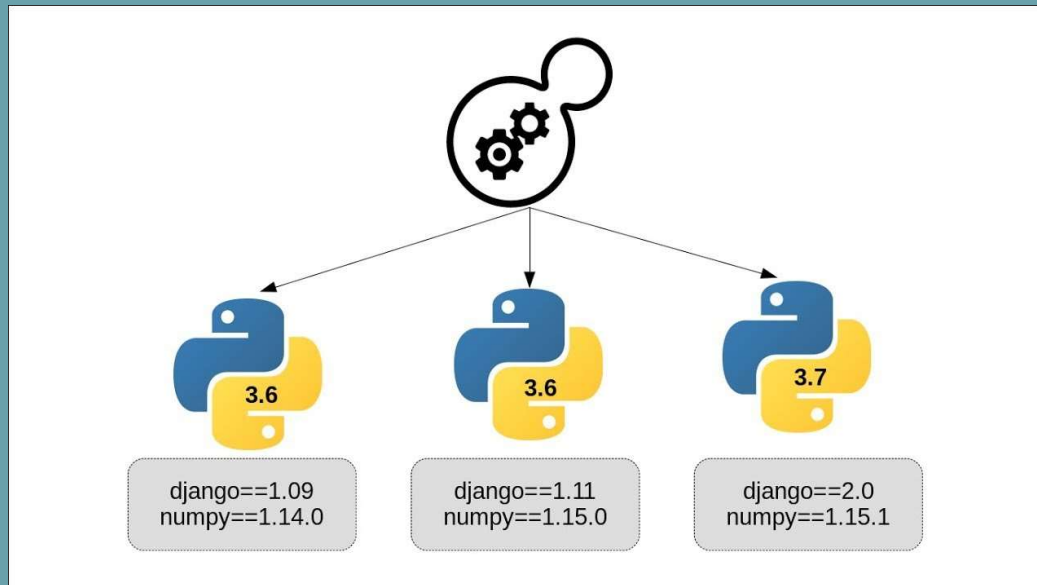
- *É um pacote completo de aplicações para construir e manter interfaces do lado do cliente e do servidor. Exemplos: Django, Web2Py e Pyramid.*

- Microframework

- *É uma estrutura de desenvolvimento de aplicativo leve que vem com uma gama definida de recursos. Convencionalmente, se combina um microframework com um framework full stack para desenvolver sistemas maiores. Exemplos: Bottle, Flask e Falcon.*

- Framework Assíncrono

- *É basicamente uma computação paralela que funciona de forma independente em comparação com os threads de aplicativos subjacentes. Geralmente é usado quando os programadores precisam combinar as grandes redes coexistentes. Exemplos: AIOHTTP e Sanic.*



AMBIENTES VITUAIS

virtualenvs

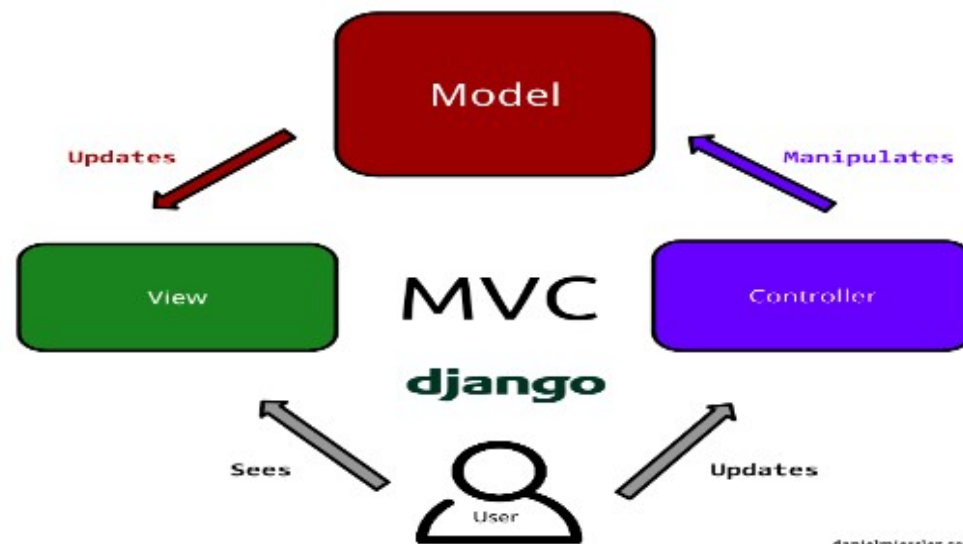
Atividade 1

Configuração do ambiente Django

Clonar o repositório <url>

1. Gerar uma Branch com o seu nome.
2. Gerar um ambiente virtual
3. Ativar o Ambiente virtual
4. Efetuar instalação do Django
5. Gerar lista requerimentos
6. Efetuar *commit* da lista de requerimentos
7. Realizar *push* para *branch* com o **SEU NOME**

MVC

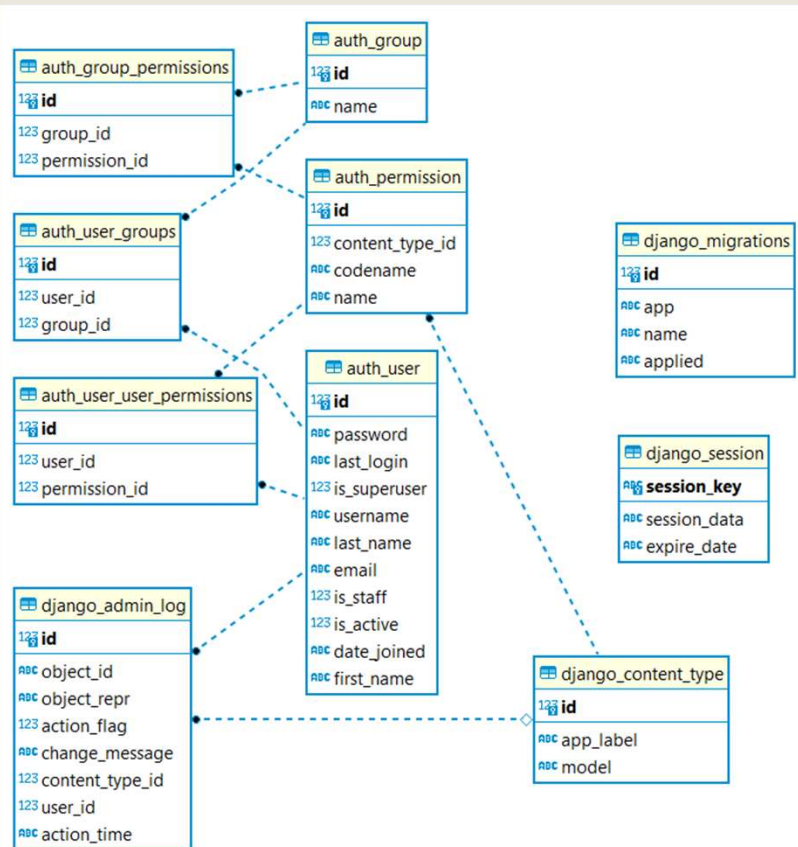


Model View Controller(MVC) in Django

Estrutura da aplicação no Django

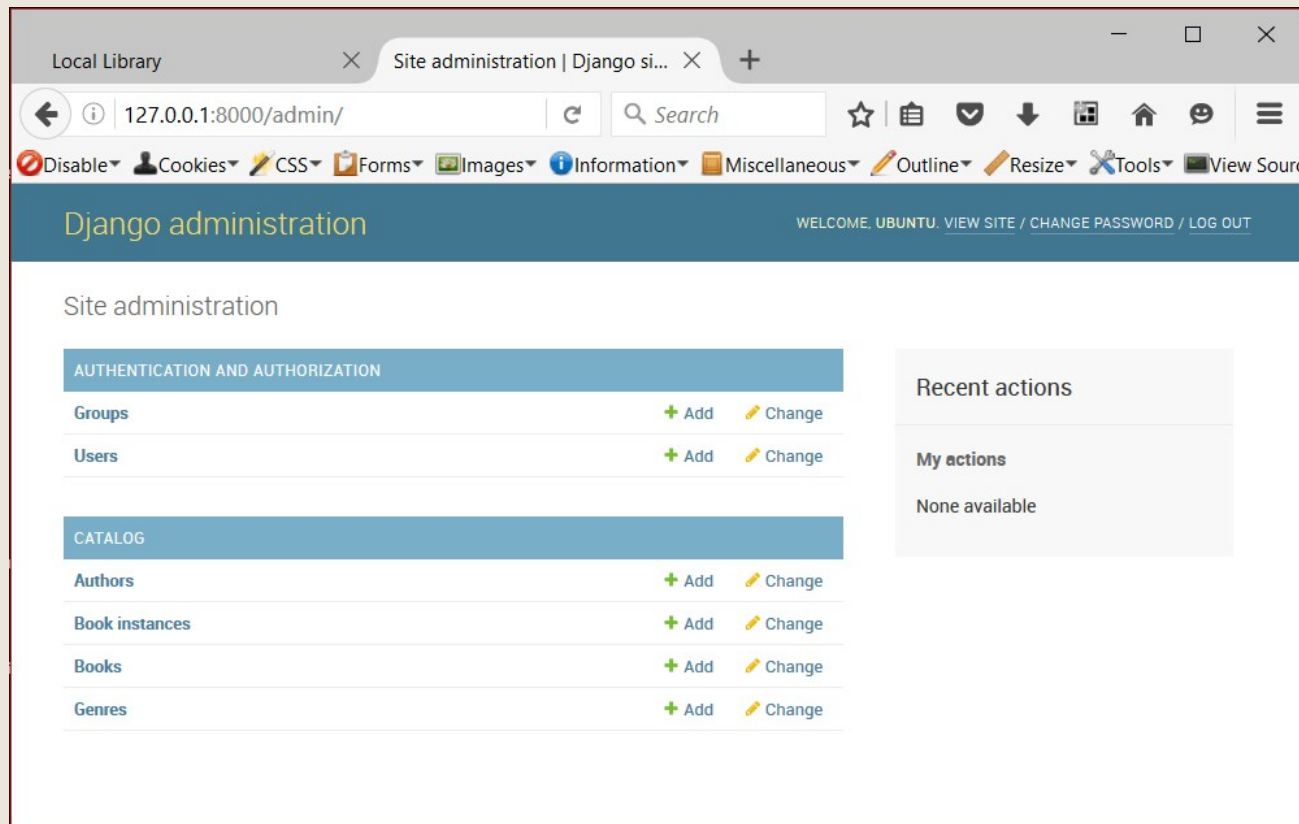
```
├── HelloWorld
│   ├── db.sqlite3
│   ├── HelloWorld
│   │   ├── __init__.py
│   │   ├── __init__.pyc
│   │   ├── settings.py
│   │   ├── settings.pyc
│   │   ├── urls.py
│   │   ├── urls.pyc
│   │   ├── wsgi.py
│   │   └── wsgi.pyc
│   ├── HelloWorldApp
│   │   ├── admin.py
│   │   ├── admin.pyc
│   │   ├── __init__.py
│   │   ├── __init__.pyc
│   │   ├── models.py
│   │   ├── models.pyc
│   │   ├── templates
│   │   │   └── helloDJ
│   │   │       ├── base.html
│   │   │       └── home.html
│   │   ├── tests.py
│   │   ├── views.py
│   │   └── views.pyc
│   └── manage.py
```

Migração da base de dados



```
$ python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying auth.0010_alter_group_name_max_length... OK
  Applying auth.0011_update_proxy_permissions... OK
  Applying auth.0012_alter_user_first_name_max_length... OK
  Applying sessions.0001_initial... OK
```

Django-admin



- é uma interface de administração automática que fornece uma interface de produção para adicionar conteúdo ao site.
- Semelhante a um Gerenciador de Conteúdo.

Atividade 2

Criar projeto Django

Iniciar projeto Django

1. Gerar e aplicar migrações da base de dados
2. Iniciar o servidor
3. Criar aplicação
4. Gerar uma página de “Bem Vindos ao Django!”
5. Configurar o componente Administrador.
6. Acessar módulo de administração.
7. Efetuar *commit* da lista de requerimentos
8. Realizar *push* para *branch* com o **SEU NOME**

Estudo de Caso

- Uma agropecuária tem uma demanda de gestão da movimentação dos caprinos no pasto.
- Na primeira fase do projeto, cada ovino foi etiquetado com uma tag Beacon e roteadores no pasto conseguem calcular e comunicar ao servidor periodicamente a posição das tags.
- Para controle de cada ovino a fazenda precisa de um sistema cadastro que vincule o caprino a tag e registre o peso, data de nascimento, raça, cor, gênero e posição periodicamente.

Django Auth

- Aplicação nativa que permite:
 - *Manter o registro de acessos de usuários.*
 - *Permissões de acesso aos métodos da Views (ACL).*
 - *Atualização de senha e recuperação de senha.*



Container Docker



- Docker Compose é uma ferramenta para executar várias aplicações containers em um determinado Docker.
- Gunicorn: ou "Green Unicorn" é um servidor HTTP Python Web Server Gateway Interface leve nos recursos do servidor e rápido.
- NGINX é um servidor web com estrutura assíncrona e orientada à eventos, possibilitando o processamento de muitas solicitações ao mesmo tempo.

Atividade 3

Fazer um CRUD dos carneiros

Iniciar projeto Django

1. Construir uma aplica Django
2. Pegar uma entidade independente do estudo de caso
3. Elaboras o modelo
4. Elaborar o formulário
5. Elaboras os métodos de CRUD
6. Construir páginas de CRUD
7. Restringir acesso com login
8. Efetuar *commit* da lista de requerimentos
9. Realizar *push* para *branch* com o SEU NOME

Questionário

- Prazo
16/02/2023!!!
- <https://forms.office.com/r/gccCMec3Fx>





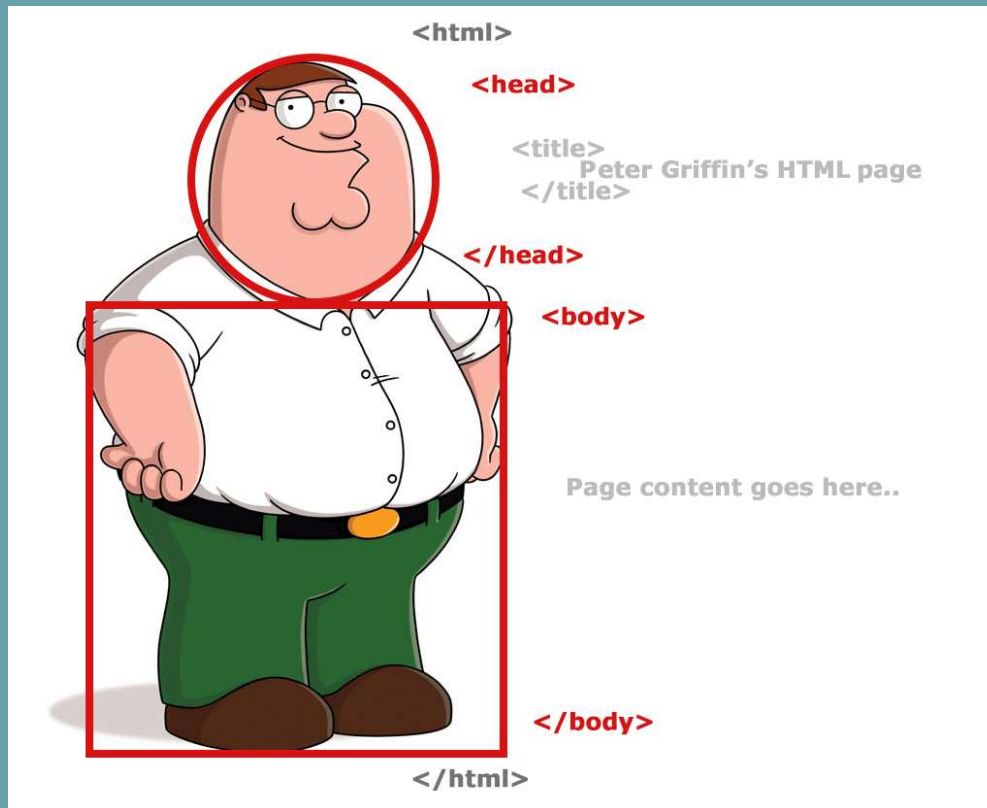
DUVIDAS?

Próximo encontro

- Estrutura de templates HTML Bootstrap
- Removendo o modo de depuração
- Permissões de acesso Django
- Swagger

Referências

- **CHEN -2023:** CHEN, Xiubo. Environmental landscape design and planning system based on computer vision and deep learning. **Journal of Intelligent Systems**, v. 32, n. 1, p. 20220092, 2023.
- **FARUK-2022:** FARUK, Mohammad et al. Adoption of big data analytics in marketing: an analysis in Bangladesh. **Journal of Data, Information and Management**, p. 1-14, 2022.
- Foundation, D. S. (2023a). Documentação do django. [Online; acessada 10-fevereiro-2023].



BONS
ESTUDOS!