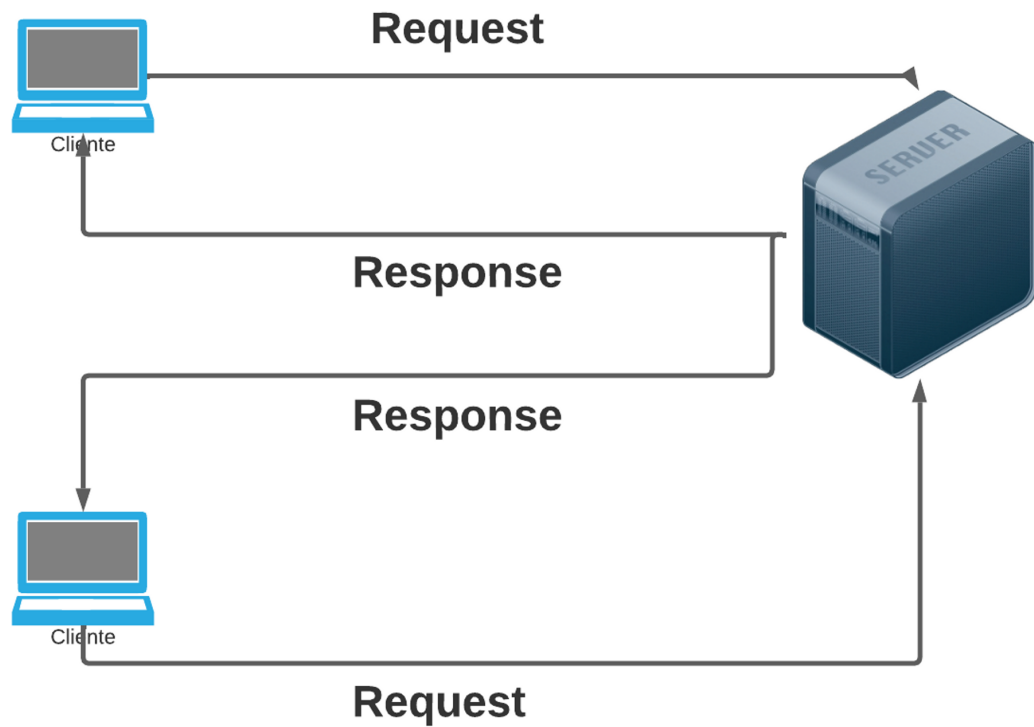


# **AULA 1 | PSW - O RETORNO**

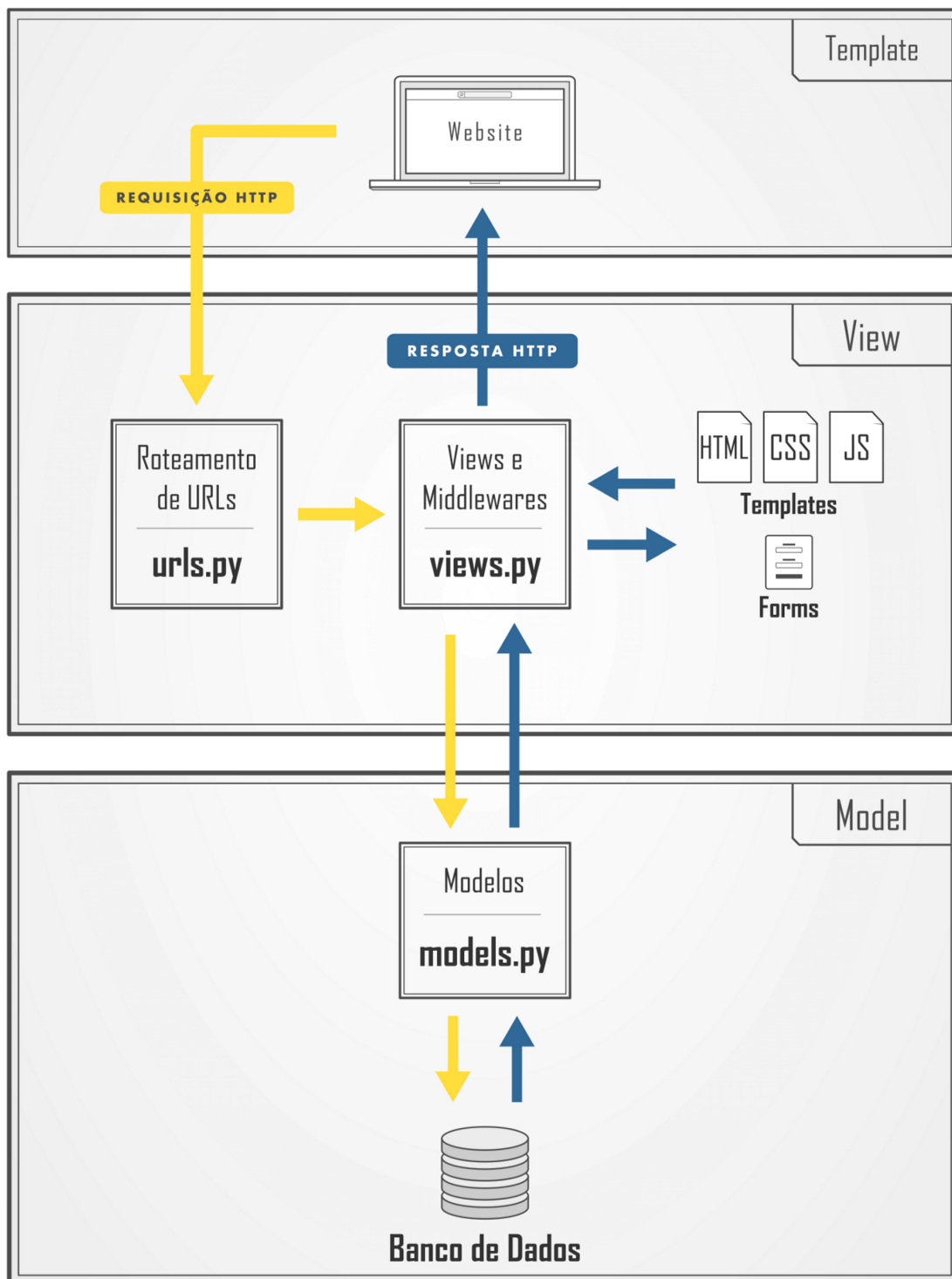
## ▼ Conceitos



Cliente servidor:



Fluxo de dados no Django:

# ARQUITETURA DO django



 Fluxo da Requisição  
 Fluxo da Resposta

## ▼ A aplicação



**ADO.TE**

## Cadastro-se

Seu cadastro salva vidas.

---

Nome

E-mail

Senha

Confirme senha

Digite sua senha

[SALVAR](#) [Já tenho uma conta](#)



ADO.TE

## QUERO DIVULGAR

Nome:

Estado:

Cidade:

Teclasse para contato:







Tags:

☐ Idiô  
☐ Sociolinguagem com crianças  
☐ Multicultural  
☐ Catequismo

Raça:

ADO.TE  
Home  
Adotar  
Divulgar

### Seus pets

Foto	Nome	Raça	Status	Ação
	Theo	SRO	<a href="#">Adoptar</a>	<a href="#">Reservar</a>
	Marvin	Golden	<a href="#">Ver perfil</a>	<a href="#">Reservar</a>
	Pluto	Poodle	<a href="#">Ver perfil</a>	<a href="#">Reservar</a>
	Theo	SRO	<a href="#">Adoptar</a>	<a href="#">Reservar</a>
	Marvin	Golden	<a href="#">Ver perfil</a>	<a href="#">Reservar</a>
	Pluto	Poodle	<a href="#">Ver perfil</a>	<a href="#">Reservar</a>

ADO.TE

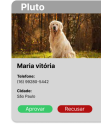
---

Home

Adolar

Dinwiger

### Solicitações de adoções









ADO.TE

---

Home

Adopter

Designer

Adote				
	Busque pela cidade...	Busque pela raça...	FILTROS	
Foto	Nome	Raça	Status	Cidade
	Theo	SRD	<a href="#">VER ANUNCIO</a>	São Paulo
	Marvin	Golden	<a href="#">VER ANUNCIO</a>	São Paulo
	Theo	SRD	<a href="#">VER ANUNCIO</a>	São Paulo
	Marvin	Golden	<a href="#">VER ANUNCIO</a>	São Paulo
	Theo	SRD	<a href="#">VER ANUNCIO</a>	São Paulo
	Marvin	Golden	<a href="#">VER ANUNCIO</a>	São Paulo

ADO.TE  
Home  
Adotar  
Divulgar

### Informações do pet



**Descrição:**  
is simply dummy text of the printing and typesetting industry. Lorem ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown

Raça

Viro-lata

### Informações residenciais

**Estado:** São Paulo **Cidade:** Franca  
**Telefone para contato:** (16) 99280-5440

[Solicitar adoção](#)

ADO.TE

---

Home

Adonar

Donalgar

## Dashboard

Animais para adoção por estado



Animais para adoção por espécie



## PSW - O RETORNO

Last modified by PyCursos Python • 5 days ago

## ▼ Configurações iniciais

Primeiro devemos criar o ambiente virtual:

```
# Criar # Linux python3 -m venv venv # Windows python -m venv venv
```

Após a criação do venv vamos ativa-lo:

```
#Ativar # Linux source venv/bin/activate # Windows  
venv\Scripts\Activate # Caso algum comando retorne um erro de  
permissão execute o código e tente novamente: Set-ExecutionPolicy -  
Scope CurrentUser -ExecutionPolicy RemoteSigned
```

Agora vamos fazer a instalação do Django e as demais bibliotecas:

```
pip install django pip install pillow
```

Vamos criar o nosso projeto em Django:

```
django-admin startproject adote .
```

Crie um novo app:

```
python manage.py startapp usuarios
```

Crie uma URL para o app usuários:

```
path('auth/', include('usuarios.urls')),
```

## ▼ Cadastro

Crie uma URL para realizar o cadastro de usuários:

```
from django.urls import path from . import views urlpatterns = [
    path('cadastro/', views.cadastro, name="cadastro"), ]
```

Crie a função cadastro:

```
def cadastro(request): if request.method == "GET": return
    render(request, 'cadastro.html')
```

Configure onde o Django irá procurar os arquivos html:

```
os.path.join(BASE_DIR, 'templates')
```

Crie o arquivo cadastro.html

```

<!doctype html> <html lang="pt-BR"> <head> <meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<title>Cadastro</title> <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-
alpha1/dist/css/bootstrap.min.css" rel="stylesheet"> </head> <body>
<div class="container-fluid"> <div class="row"> <div class="col-md-5
bg-side"> <div class='header'> <img src="" width="9%"><span
class="font-logo">ADO.TE</span> </div> <div class="body-form">
<h2>Cadastre-se</h2> <p>Seu cadastro salva vidas.</p> <hr> <form
action="" method="POST"> <span>Nome:</span> <input type="text"
class="form-control" placeholder="Digite seu nome..." name="nome">
<br> <span>E-mail:</span> <input type="email" class="form-control"
placeholder="Digite seu email..." name="email"> <br> <div class="row">
<div class="col-md"> <span>Senha:</span> <input type="password"
class="form-control" placeholder="Digite sua senha..." name="senha">
</div> <div class="col-md"> <span>Confirmar senha:</span> <input
type="password" class="form-control" placeholder="Digite sua senha..."
name="confirmar_senha"> </div> </div> <input type="submit"
value="SALVAR" class="btn-custom-primary"> <a href="" class="link-
float">Já tenho uma conta</a> </form> </div> </div> <div class="col-
md-7"> <div class="side-logo"> <img class="logo-margin-top" src=""
width="20%"> <h1>ADO.TE</h1> <p>Nos ajude a salvar vidas.</p> </div>
</div> </div> </div> <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-
alpha1/dist/js/bootstrap.bundle.min.js"></script> </body> </html>

```

Configure os arquivos estáticos:

```

STATIC_URL = '/static/' STATICFILES_DIRS = (os.path.join(BASE_DIR,
'templates/static'),) STATIC_ROOT = os.path.join('static') MEDIA_ROOT
= os.path.join(BASE_DIR, 'media') MEDIA_URL = '/media/'

```

Crie o arquivos cadastro.css



```
.bg-side{ background-color: #391D34; color: white; height: 100vh; }
.header{ border-bottom: 1px solid white; padding: 20px; } .body-form{
padding: 20px; } .font-logo{ margin-left: 10px; font-size: 40px; }
.btn-custom-primary{ background-color: #3D7B80; border: none; box-
shadow: 2px 2px 5px 1px black; color: white; font-size: 20px; padding:
5px 20px 5px 20px; border-radius: 10px; margin-top: 20px; } .link-
float{ text-decoration: none; float: right; margin: 20px; } .side-
logo{ text-align: center; margin-top: 60px; }
```

No html carregue os estáticos:

```
{% load static %}
```

Importe o CSS:

```
<link href="{% static 'usuarios/css/cadastro.css' %}"
rel="stylesheet">
```

Carregue a logo:

```
{% static 'usuarios/img/logo_adote.png' %}
```

Envie os dados do formulário para a view cadastro:

```
<form action="{% url 'cadastro' %}" method="POST">
```

Crie a view cadastro completa:

```

from django.contrib.auth.models import User
def cadastro(request):
    if request.method == "GET":
        return render(request, 'cadastro.html')
    elif request.method == "POST":
        nome = request.POST.get('nome')
        email = request.POST.get('email')
        senha = request.POST.get('senha')
        confirmar_senha = request.POST.get('confirmar_senha')
        if len(nome.strip()) == 0 or len(email.strip()) == 0 or len(senha.strip()) == 0 or len(confirmar_senha.strip()) == 0:
            return render(request, 'cadastro.html')
        if senha != confirmar_senha:
            return render(request, 'cadastro.html')
        try:
            user = User.objects.create_user(username=nome, email=email, password=senha, )
        except:
            return render(request, 'cadastro.html')

```

Configure as mensagens:

## ▼ Login

Crie a URL para o login:

```

from django.contrib.messages import constants
MESSAGE_TAGS = {
    constants.DEBUG: 'alert-primary',
    constants.ERROR: 'alert-danger',
    constants.SUCCESS: 'alert-success',
    constants.INFO: 'alert-info',
    constants.WARNING: 'alert-warning',
}

```

```

path('login/', views.logar, name="login"),

```

Crie a view logar:

```

from django.contrib import messages
from django.contrib.messages import constants

```

```

def logar(request):
    if request.method == "GET":
        return render(request, 'login.html')

```

Exiba as mensagens no HTML:

```

{% load static %}
<!doctype html>
<html lang="pt-BR">
<head>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<title>Cadastro</title>
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet">
<link href="{% static 'usuarios/css/cadastro.css' %}" rel="stylesheet">
</head>

```