

PROGRAMAÇÃO BACK-END

The Django logo, featuring the word "django" in a white, lowercase, sans-serif font with a small dot over the 'i', set against a dark green rectangular background.

The web framework for
perfectionists with deadlines.

Django Cadastro de Usuário Alterando forms

PROGRAMAÇÃO BACK-END

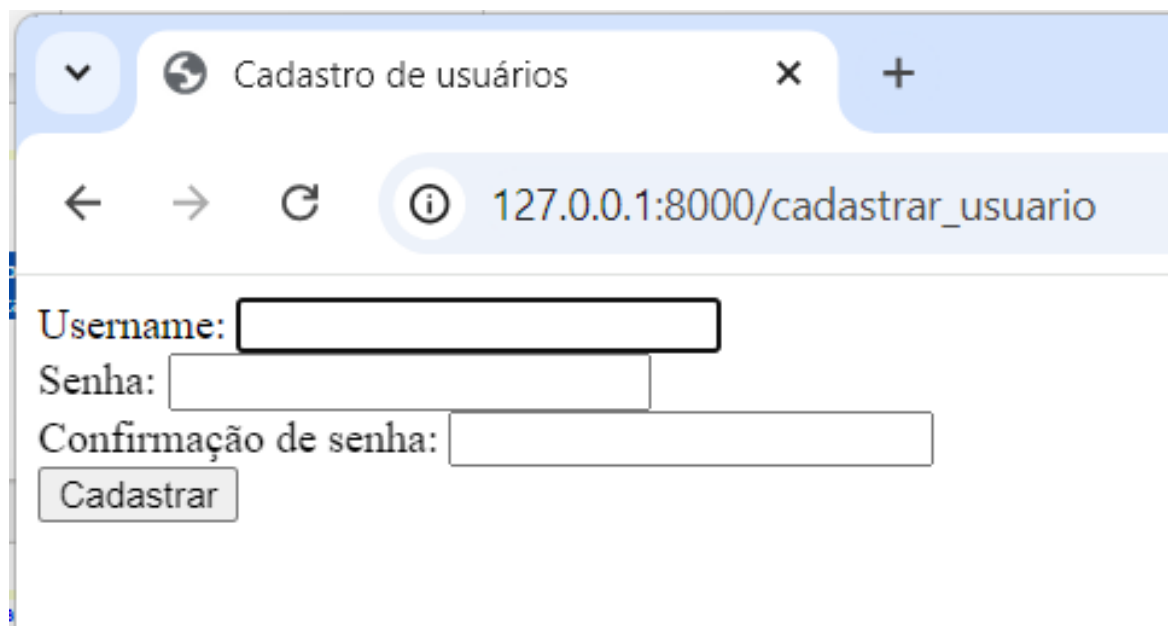
Usando forms no Cadastro de Usuário

No cadastro de usuários no Django, ele utiliza um formulário que limita aos campos de **username** e **password** (sendo esta é digitada duas vezes para confirmação).

Para acrescentarmos outros campos devemos utilizar um novo forms (**CustomUserCreationForm**) que derive (herdar) do modelo (**UserCreationForm**).

PROGRAMAÇÃO BACK-END

Se utilizarmos o **UserCreationForm** tem-se como resultado a tela abaixo:



The screenshot shows a web browser window with a single tab titled "Cadastro de usuários". The address bar displays the URL "127.0.0.1:8000/cadastrar_usuario". The page content includes a registration form with the following elements:

- A label "Username:" followed by a text input field.
- A label "Senha:" followed by a text input field.
- A label "Confirmação de senha:" followed by a text input field.
- A "Cadastrar" button located below the confirmation field.

PROGRAMAÇÃO BACK-END

Vamos supor que se quer acrescentar os seguintes campos no momento do **cadastro**:

- ✓ Endereco;
- ✓ Cidade;
- ✓ Estado;
- ✓ Cep;

PROGRAMAÇÃO BACK-END

Observação:

Para realizar os testes desta apresentação serão utilizados os arquivos criados na apresentação:

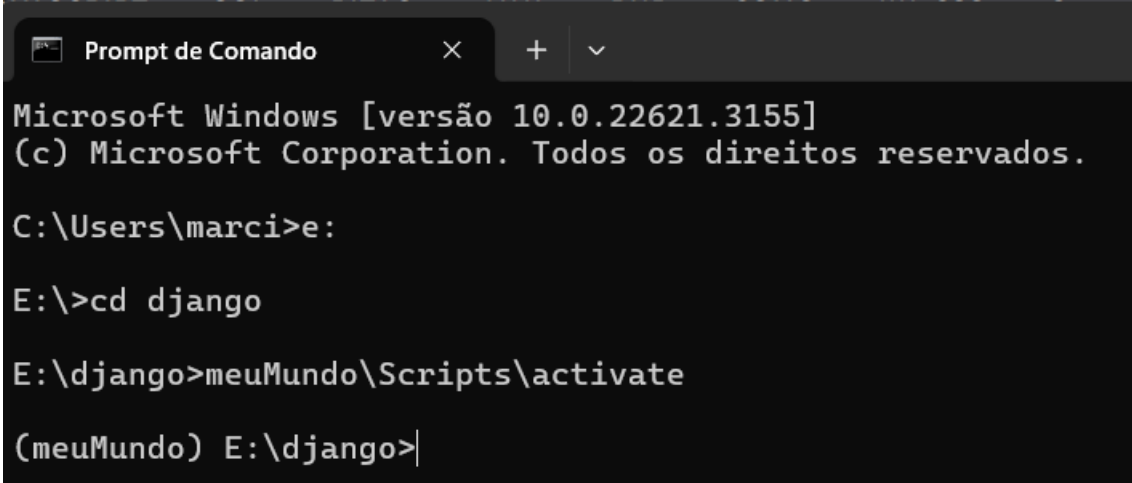
Django-Auth Users

PROGRAMAÇÃO BACK-END

Iniciando

PROGRAMAÇÃO BACK-END

Rodar o script activate.bat



```
Prompt de Comando
Microsoft Windows [versão 10.0.22621.3155]
(c) Microsoft Corporation. Todos os direitos reservados.

C:\Users\marci>e:

E:\>cd django

E:\django>meuMundo\Scripts\activate

(meuMundo) E:\django>
```


Este passo é importante para configurar as variáveis de ambiente.

PROGRAMAÇÃO BACK-END

Iniciar Servidor Postgresql

PROGRAMAÇÃO BACK-END

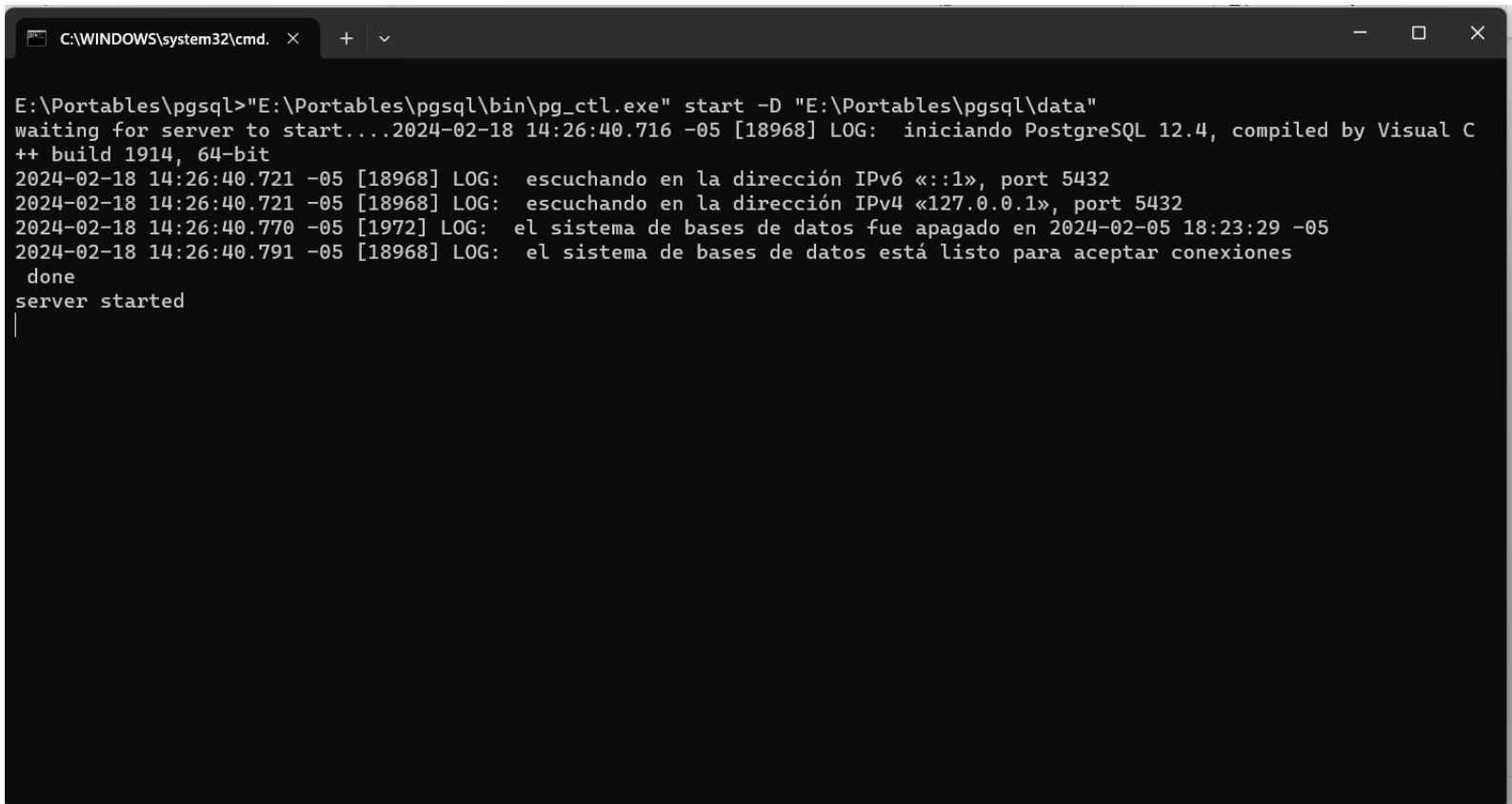
Na pasta `../pgsql` execute o arquivo **PostgreSQL-Start.bat** para iniciar o servidor



Nome	Data de modificação	Tipo	Tamanho
bin	11/08/2020 08:41	Pasta de arquivos	
data	18/02/2024 16:26	Pasta de arquivos	
doc	11/08/2020 08:41	Pasta de arquivos	
include	11/08/2020 08:41	Pasta de arquivos	
lib	11/08/2020 08:41	Pasta de arquivos	
pgAdmin 4	11/08/2020 08:41	Pasta de arquivos	
share	11/08/2020 08:41	Pasta de arquivos	
StackBuilder	11/08/2020 08:41	Pasta de arquivos	
Make Cluster.bat	09/01/2017 14:12	Arquivo em Lotes ...	1 KB
PgAdmin4.bat	28/04/2017 22:29	Arquivo em Lotes ...	1 KB
PostgreSQL-Restart.bat	06/04/2012 09:42	Arquivo em Lotes ...	1 KB
PostgreSQL-Start.bat	28/04/2017 21:46	Arquivo em Lotes ...	1 KB
PostgreSQL-Stop.bat	06/04/2012 09:43	Arquivo em Lotes ...	1 KB
Service - Delete.bat	29/08/2020 23:25	Arquivo em Lotes ...	1 KB
Service - Maker.bat	29/08/2020 23:26	Arquivo em Lotes ...	1 KB

PROGRAMAÇÃO BACK-END

Será aberta uma janela em Texto informando que o servidor iniciou (server started**).**
Não feche esta janela



```
C:\WINDOWS\system32\cmd. x + v

E:\Portables\pgsql>"E:\Portables\pgsql\bin\pg_ctl.exe" start -D "E:\Portables\pgsql\data"
waiting for server to start...2024-02-18 14:26:40.716 -05 [18968] LOG:  iniciando PostgreSQL 12.4, compiled by Visual C
++ build 1914, 64-bit
2024-02-18 14:26:40.721 -05 [18968] LOG:  escuchando en la dirección IPv6 «::1», port 5432
2024-02-18 14:26:40.721 -05 [18968] LOG:  escuchando en la dirección IPv4 «127.0.0.1», port 5432
2024-02-18 14:26:40.770 -05 [1972] LOG:  el sistema de bases de datos fue apagado en 2024-02-05 18:23:29 -05
2024-02-18 14:26:40.791 -05 [18968] LOG:  el sistema de bases de datos está listo para aceptar conexiones
done
server started
|
```

**Alterando o
Cadastro de usuários**

Templates

PROGRAMAÇÃO BACK-END

Utilizando a App **dvdrentalapp**. Na pasta **templates** altere o arquivo **cadastro.html** digite o seguinte conteúdo:

```
dvdrentalapp > templates > <> cadastro.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <title>Cadastro de usuários</title>
6  </head>
7  <body>
8      <h1>Cadastro de Usuário</h1>
9      <form method="post">
10         {% csrf_token %}
11         {{ form_usuario.as_p }}
12         <div class="form-group">
13             <input type="submit" value="Cadastrar" class="btn btn-primary">
14         </div>
15     </form>
16 </body>
17 </html>
```

PROGRAMAÇÃO BACK-END

Arquivos views.py e urls.py

PROGRAMAÇÃO BACK-END

Na pasta **dvdrentalapp** adicione o seguinte conteúdo no arquivo **views.py**

```
12 from .forms import CustomUserCreationForm
```

```
53 def cadastrar_usuario(request):
54     if request.method == "POST":
55         form_usuario = CustomUserCreationForm(request.POST)
56         if form_usuario.is_valid():
57             form_usuario.save()
58             return redirect('salva')
59     else:
60         form_usuario = CustomUserCreationForm()
61     return render(request, 'cadastro.html', {'form_usuario': form_usuario})
62
```


PROGRAMAÇÃO BACK-END

Na pasta **dvdrentalapp** adicione o seguinte conteúdo no arquivo **forms.py**

```
dvdrentalapp > forms.py > ...
1  from django.forms import ModelForm
2  from django import forms
3  from dvdrentalapp.models import *
4  from .models import Customer
5  from django.contrib.auth.forms import UserCreationForm
6
7  class CustomUserCreationForm(UserCreationForm):
8      class Meta(UserCreationForm.Meta):
9          model = CustomUser
10         fields = UserCreationForm.Meta.fields + ('endereco', 'cidade', 'estado', 'cep')
11
12     class CustomerForm(ModelForm):
13         class Meta:
14             model = Customer
15             #fields = '__all__'
16             fields = ['first_name', 'last_name', 'email', 'address', 'activebool', 'store_id', 'create_date']
```

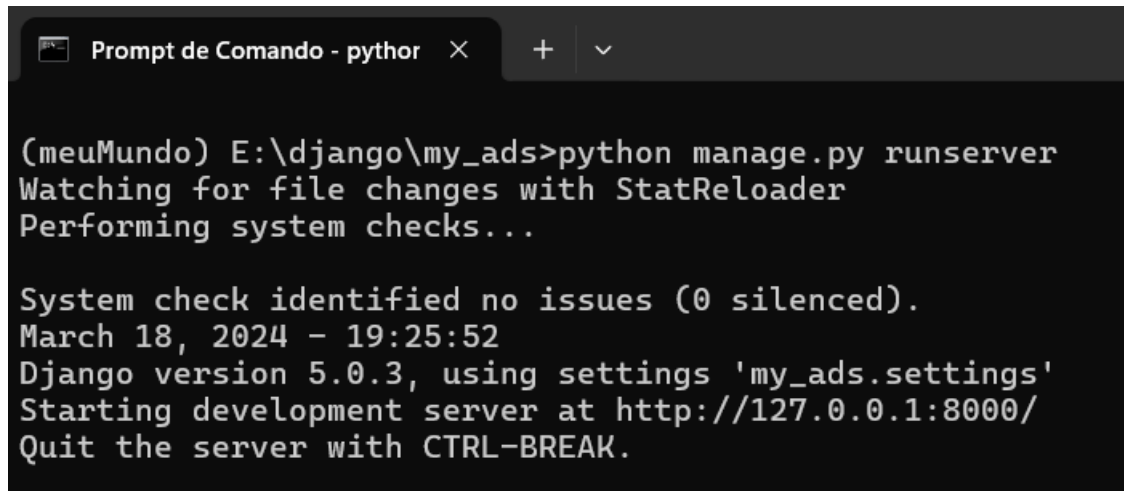
A **class CustomerForm** não é necessário digitar

**Iniciando o servidor
e testando**

PROGRAMAÇÃO BACK-END

Execute na pasta **my_ads** o comando:

python manage.py runserver



```
Prompt de Comando - pythor  X  +  v

(meuMundo) E:\django\my_ads>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
March 18, 2024 - 19:25:52
Django version 5.0.3, using settings 'my_ads.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Vá ao navegador e digite:

<http://127.0.0.1:8000/home/>

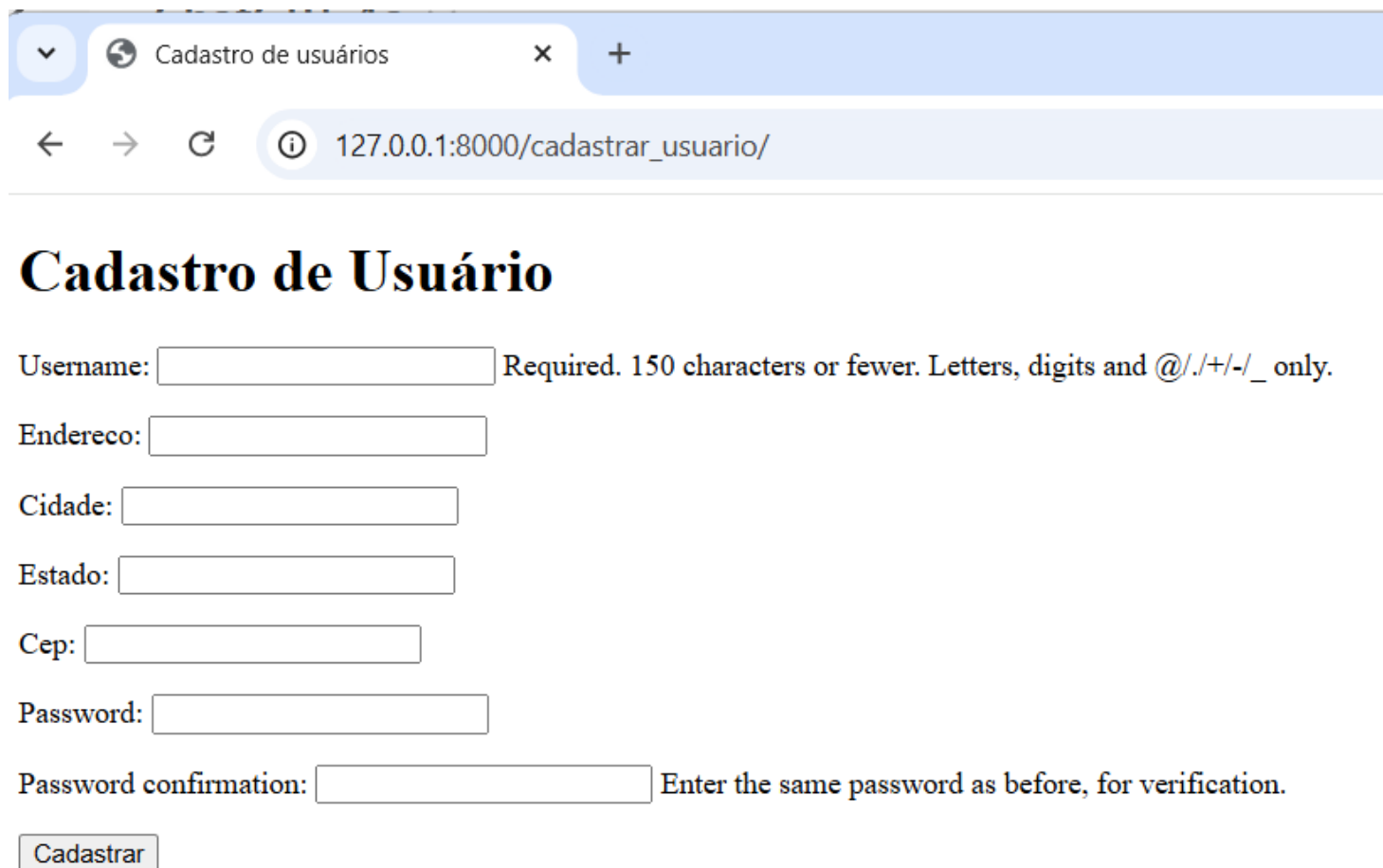
PROGRAMAÇÃO BACK-END

O resultado da página será o formulário home.



PROGRAMAÇÃO BACK-END

O Cadastro de Usuário contém novos campos conforme a figura.



The image shows a web browser window with a single tab titled "Cadastro de usuários". The address bar displays the URL "127.0.0.1:8000/cadastrar_usuario/". The page content features a heading "Cadastro de Usuário" followed by a series of input fields for user registration. The fields are labeled "Username:", "Endereco:", "Cidade:", "Estado:", "Cep:", "Password:", and "Password confirmation:". The "Username:" field has a hint text: "Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only." The "Password confirmation:" field has a hint text: "Enter the same password as before, for verification." At the bottom left, there is a button labeled "Cadastrar".

Cadastro de usuários

127.0.0.1:8000/cadastrar_usuario/

Cadastro de Usuário

Username: Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Endereco:

Cidade:

Estado:

Cep:

Password:

Password confirmation: Enter the same password as before, for verification.

Cadastrar