

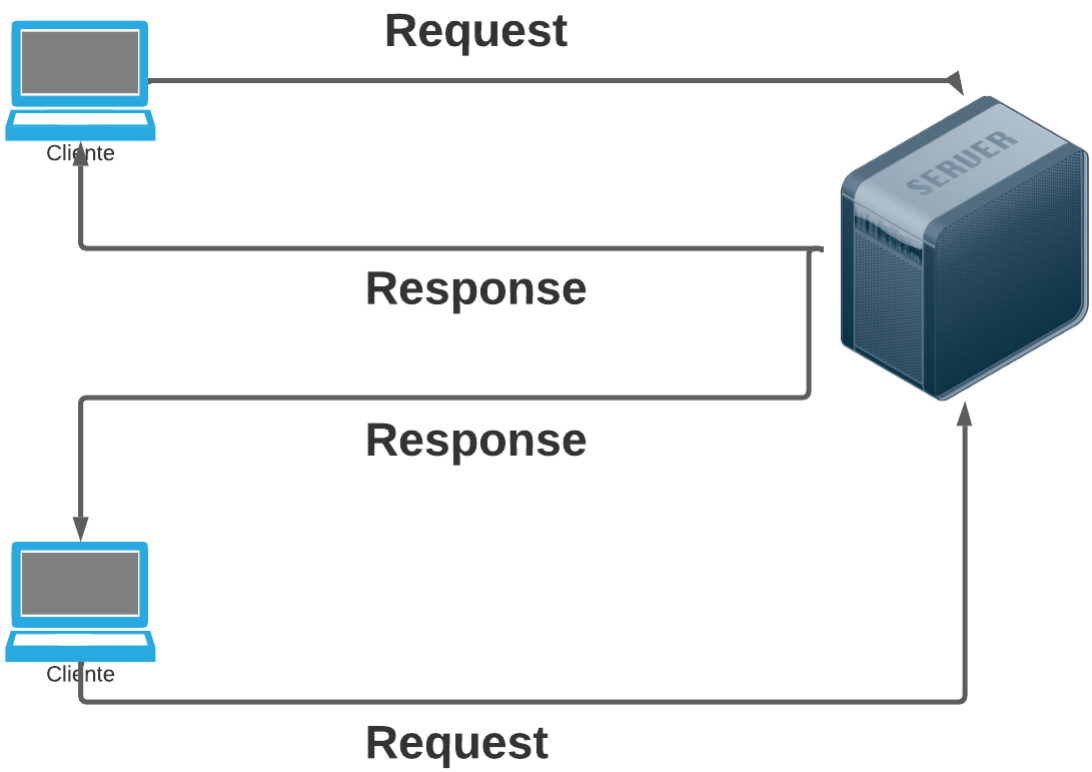


## Aula 1

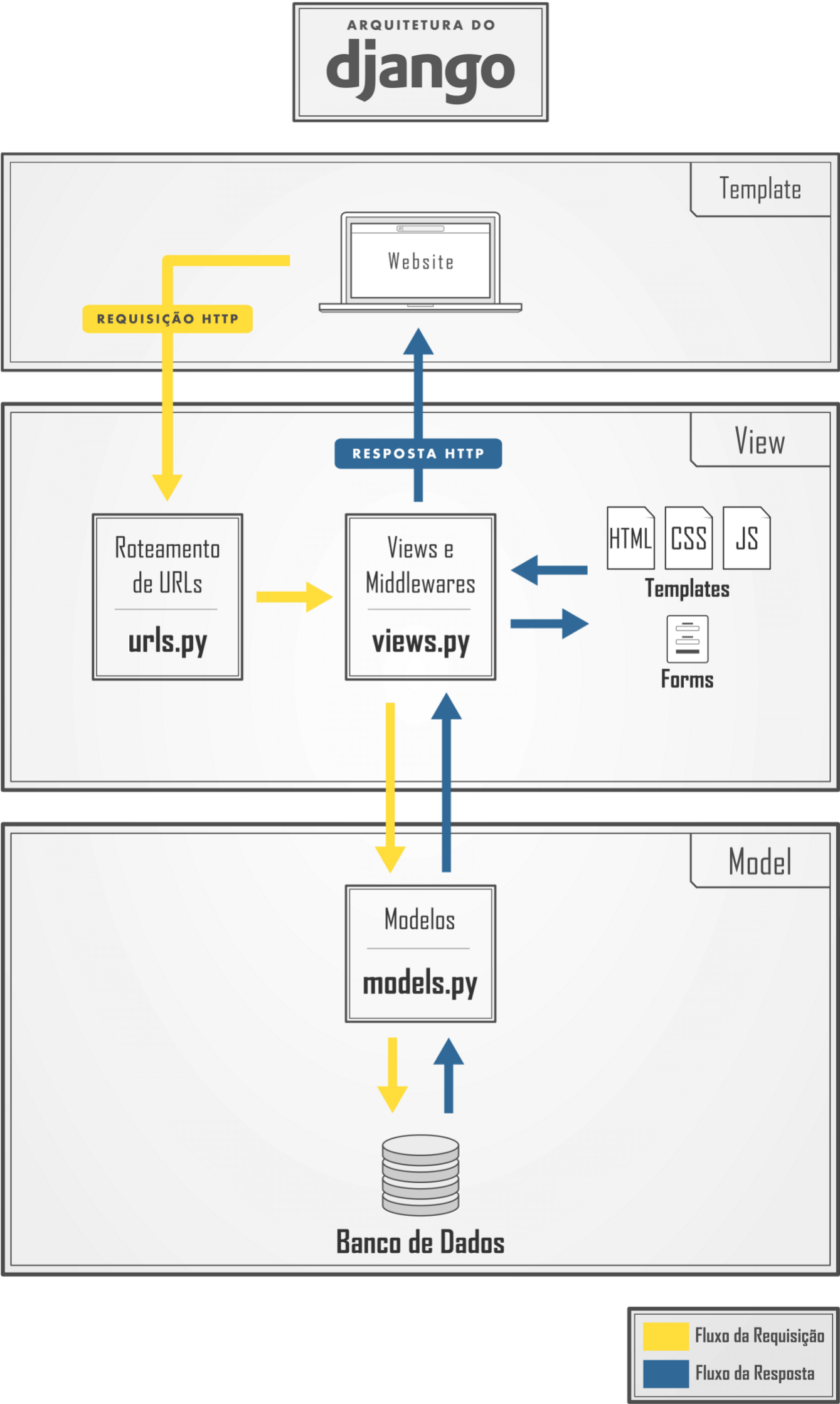
Acesse diretamente pelo Notion:

<https://grizzly-amaranthus-f6a.notion.site/Aula-1-fe55c39497e842ebb855562c572a3feb?pvs=4>

### ▼ Conceitos



Fluxo de dados no Django:



▼ O projeto

<https://www.figma.com/file/BLKelQ7XgXlGnSQnfqGled/Untitled?type=design&node-id=0%3A1&mode=design&t=80WjGb2madRtfrKp-1>

▼ 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 Django:

```
django-admin startproject healing .
```

Rode o servidor para testar:

```
python manage.py runserver
```

Crie o app usuario:

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

Ative o auto-save

INSTALE O APP!

▼ Cadastro

Crie uma URL para o app usuario:

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

No app usuario crie uma URL para cadastro:

```
from django.urls import path
from . import views

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

Em usuarios/views.py crie a view cadastro:

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

Configure onde o Django deve buscar por arquivos de templates:

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

Em templates crie o base.html:

```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" href="">
    <title>{% block 'title' %}{% endblock %}</title>
    <link href="https://cdn.jsdelivrivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet">
    {% block 'head' %}{% endblock 'head' %}
  </head>
  <body>
    {% block 'body' %}{% endblock 'body' %}
    <script src="https://cdn.jsdelivrivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script>
  </body>
</html>
```

Crie o cadastro.html

```
{% extends "base.html" %}

{% block 'body' %}

  <div class="container-fluid">
    <div class="row">
      <div class="col-md d-flex justify-content-center">
        <div class="cont-cadastro">
          <br>
          <br>
          <section class="cont-logo">
            
            <label class="text-logo">HEALING</label>
          </section>
          <hr>

          <form action="" method="post">
            <h2 class="fonte-destaque1">Cadastre-se</h2>
            <br>
            <label for="">Username</label>
            <input type="text" name="username" id="" class="form-control" placeholder="Username ...">
            <br>
            <label for="">E-mail</label>
            <input type="text" name="email" id="" placeholder="email@email.com" class="form-control">
          </form>
        </div>
      </div>
    </div>
  </div>

{% endblock %}
```

```

        <br>
        <div class="row">
            <div class="col-md">
                <label for="">Senha</label>
                <input type="password" name="senha" class="form-control" placeholder="Digite sua senha ...">
            </div>
            <div class="col-md">
                <label for="">Confirmar senha</label>
                <input type="password" name="confirmar_senha" class="form-control" placeholder="Digite sua senha novamente ..." id="">
            </div>
        </div>
        <br>

        <input type="submit" value="Cadastrar" class="btn btn-success btn-dark-color">
        <a href="#" class="btn btn-dark-color-outline">Já possui uma conta</a>
    </form>
</div>

</div>
<div class="col-md bg-main d-flex justify-content-center align-items-center">
    

</div>
</div>
</div>

{% endblock 'body' %}
```

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 base.css

```

:root{

    --main-color: #00CCBE;
    --dark-color: #09A6A3;
    --contrast-color: #FFD686;

}

.bg-color-dark{

    background-color: var(--main-color);

}

.p-bold{

    font-weight: bold;

}

.color-dark{

    color: var(--dark-color);

}
```

Importe o base.css em base.html

```

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

Agora em usuarios/css crie o usuarios.css

```

.bg-main{
    background-color: var(--main-color);
    height: 100vh;
}

.cont-cadastro{
    width: 60%;
}

.cont-logo{
    text-align: center;
}

.text-logo{
    font-size: 30px;
    font-family: system-ui, -apple-system, BlinkMacSystemFont, 'Segoe UI', Roboto, Oxygen, Ubuntu, Cantarell, 'Open Sans', 'Helvetica Neue', sans-serif;
}

.fonte-destaque1{
    color: var(--dark-color)
}

.btn-dark-color{
    background-color: var(--dark-color);
    color: white;
}

.btn-dark-color-outline{

    border: 1px solid var(--dark-color);
    color: var(--dark-color)

}

.btn-dark-color-outline:hover{
```

```
background-color: var(--dark-color);
color: white;
border: 1px solid var(--dark-color);
}
```

Em cadastro.html importe o usuarios.css

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

Adicione a logo da plataforma em /templates/static/geral/img

[https://drive.google.com/drive/folders/1crt\\_JSI8RGWiwjylukwV8EkdoPFLquSM?usp=sharing](https://drive.google.com/drive/folders/1crt_JSI8RGWiwjylukwV8EkdoPFLquSM?usp=sharing)

Adicione a logo no HTML:

```

```

Adicione a ilustração em /templates/usuarios/img

```

```

Execute as migrações:

```
python manage.py makemigrations
python manage.py migrate
```

Configure sua tag form para enviar os dados par nosso Back-end:

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

Na view cadastro:

```
def cadastro(request):
    if request.method == "GET":
        return render(request, 'cadastro.html')
    elif request.method == "POST":
        username = request.POST.get('username')
        email = request.POST.get("email")
        senha = request.POST.get("senha")
        confirmar_senha = request.POST.get('confirmar_senha')

        users = User.objects.filter(username=username)

        if users.exists():
            print('Erro 1')
            return redirect('/usuarios/cadastro')

        if senha != confirmar_senha:
            print('Erro ')
            return redirect('/usuarios/cadastro')

        if len(senha) < 6:
            print('Erro 3')
            return redirect('/usuarios/cadastro')

        try:
            User.objects.create_user(
                username=username,
                email=email,
                password=senha
            )
            return redirect('/usuarios/login')
        except:
            print('Erro 4')
            return redirect('/usuarios/cadastro')
```

Configure as mensagens:

```
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',
}
```

Adicione as messages, exemplo:

```
messages.add_message(request, constants.ERROR, 'A senha deve possuir pelo menos 6 caracteres')
```

Renderize as mensagens no HTML:

```
{% if messages %}
<br>
{% for message in messages %}
<section class="alert {{message.tags}}">
    {{message}}
</section>
{% endfor %}
{% endif %}
```

▼ Login

Crie uma URL para a view login:

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

Crie a view login\_view:

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

Crie o HTML login:

```
{% extends "base.html" %}
{% load static %}

{% block 'head' %}
    <link rel="stylesheet" href="{% static 'usuarios/css/usuarios.css' %}">
{% endblock 'head' %}

{% block 'body' %}

    <div class="container-fluid">
        <div class="row">
            <div class="col-md d-flex justify-content-center">
                <div class="cont-cadastro">
                    <br>
                    <br>
                    <section class="cont-logo">
                        
                        <label class="text-logo">HEALING</label>
                    </section>
                    <hr>

                    <form action="#" method="POST">
                        <h2 class="fonte-destaque1">Logar</h2>
                        {% if messages %}
                            <br>
                            {% for message in messages %}
                                <section class="alert {{message.tags}}">
                                    {{message}}
                                </section>
                            {% endfor %}
                        {% endif %}
                        <br>
                        <label for="">Username</label>
                        <input type="text" name="username" id="" class="form-control" placeholder="Username ...">
                        <br>

                        <label for="">Senha</label>
                        <input type="password" name="senha" class="form-control" placeholder="Digite sua senha ...">
                        <br>

                        <input type="submit" value="Logar" class="btn btn-success btn-dark-color">
                        <a href="{% url 'cadastro' %}" class="btn btn-dark-color-outline">Não possuo uma conta</a>
                    </form>
                </div>

            </div>
            <div class="col-md bg-main d-flex justify-content-center align-items-center">
                

            </div>
        </div>
    </div>

{% endblock 'body' %}
```

Configure o form:

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

Em cadastro.html altere o href do botão já possuo uma conta:

```
<a href="{% url 'login' %}" class="btn btn-dark-color-outline">Já possuo uma conta</a>
```

Crie a lógica do login na View:

```
def login_view(request):
    if request.method == "GET":
        return render(request, 'login.html')
    elif request.method == "POST":
        username = request.POST.get('username')
        senha = request.POST.get("senha")

        user = auth.authenticate(request, username=username, password=senha)

        if user:
            auth.login(request, user)
            return redirect('/pacientes/home')

        messages.add_message(request, constants.ERROR, 'Usuário ou senha incorretos')
        return redirect('/usuarios/login')
```

▼ Logout

Crie uma URL para deslogar:

```
path('sair/', views.sair, name="sair")
```

Crie a view sair:

```
def sair(request):
    auth.logout(request)
    return redirect('/usuarios/login')
```

▼ Cadastro médico

Crie um novo app chamado medico

```
python manage.py startapp medico
```

Instale o app!!

Em models.py, crie a tabela no banco para salvar as informações de médico:

```
class Especialidades(models.Model):
    especialidade = models.CharField(max_length=100)
    icone = models.ImageField(upload_to="icones", null=True, blank=True)

    def __str__(self):
        return self.especialidade

class DadosMedico(models.Model):
    crm = models.CharField(max_length=30)
    nome = models.CharField(max_length=100)
    cep = models.CharField(max_length=15)
    rua = models.CharField(max_length=100)
    bairro = models.CharField(max_length=100)
    numero = models.IntegerField()
    rg = models.ImageField(upload_to="rgs")
    cedula_identidade_medica = models.ImageField(upload_to='cim')
    foto = models.ImageField(upload_to="fotos_perfil")
    user = models.ForeignKey(User, on_delete=models.DO_NOTHING)
    descricao = models.TextField(null=True, blank=True)
    especialidade = models.ForeignKey(Especialidades, on_delete=models.DO_NOTHING, null=True, blank=True)
    valor_consulta = models.FloatField(default=100)

    def __str__(self):
        return self.user.username
```

Adicione no admin.py:

```
from .models import Especialidades, DadosMedico

# Register your models here.
admin.site.register(Especialidades)
admin.site.register(DadosMedico)
```

Crie um superusuário:

```
python3 manage.py createsuperuser
```

No core do projeto, crie uma URL para o app medico:

```
path('medicos/', include('medico.urls'))
```

Agora em medico crie uma URL para uma pessoa se tornar médico.

```
from django.urls import path
from . import views

urlpatterns = [
    path('cadastro_medico/', views.cadastro_medico, name="cadastro_medico"),
]
```

Crie a view cadastr\_medico:

```
from django.contrib.auth.decorators import login_required

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

Crie o HTML cadastro\_medico.html:

```
{% extends "base.html" %}
{% load static %}

{% block 'head' %}
    <link rel="stylesheet" href="{% static 'usuarios/css/usuarios.css' %}">
{% endblock 'head' %}

{% block 'body' %}

<div class="container">
    <br>
    <br>
    <div class="row">
        <div class="col-md-8">

            <p class="p-bold">Olá, <span class="color-dark"></span></p>
            <p class="p-bold">Vamos realizar seu cadastro médico legal.</p>
            {% if messages %}
                <br>
                {% for message in messages %}
                    <section class="alert {{message.tags}}">
                        {{message}}
                    </section>
                {% endfor %}
            {% endif %}
            <br>
            <form action="#" method="post" enctype='multipart/form-data'>
                <div class="row">
                    <div class="col-md">
                        <label for="">CRM:</label>
                        <input type="text" class="form-control shadow-main-color" name="crm" placeholder="CRM...">
                    </div>
                    <div class="col-md">
                        <label for="">Cédula de identidade médica:</label>
                        <input type="file" name="cim" id="" class="form-control shadow-main-color">
                    </div>
                </div>
            </div>
        </div>
    </div>
```

```
<br>
<div class="row">
  <div class="col-md">
    <label for="">Nome completo:</label>
    <input type="text" class="form-control shadow-main-color" name="nome" placeholder="Digite seu nome ...">
  </div>
  <div class="col-md">
    <label for="">CEP</label>
    <input type="text" name="cep" id="" class="form-control shadow-main-color">

  </div>
</div>
<br>
<label for="">Rua</label>
<input type="text" name="rua" id="" class="form-control shadow-main-color" placeholder="Endereço ...">
<br>
<div class="row">
  <div class="col-md">
    <label for="">Bairro:</label>
    <input type="text" class="form-control shadow-main-color" name="bairro" placeholder="Bairro ...">
  </div>
  <div class="col-md">
    <label for="">Número:</label>
    <input type="number" name="numero" id="" class="form-control shadow-main-color">

  </div>
</div>
<br>
<label for="">RG:</label>
<input type="file" name="rg" id="" class="form-control shadow-main-color">
<br>
<label for="">Foto de perfil:</label>
<input type="file" name="foto" id="" class="form-control shadow-main-color">
<br>
<label for="">Especialidade médica</label>
<select name="especialidade" class="form-select" id="">

  <option value="">X</option>

</select>
<br>
<label for="">Descrição:</label>
<textarea name="descricao" class="form-control shadow-main-color"></textarea>
<br>
<label for="">Valor consulta:</label>
<input type="number" name="valor_consulta" class="form-control shadow-main-color">
<br>
<input type="submit" value="Cadastre-se" class="btn btn-success btn-dark-color">
</form>
</div>
<div class="col-md-4"></div>
</div>

{% endblock 'body' %}
```

Em templates/static/medicos/css crie o cadastro\_medico.css:

```
.shadow-main-color{
  box-shadow: 1px 1px 5px 1px rgba(0, 204, 190,.4);
}
```

Importe o CSS:

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

Adicione o nome do usuário:

```
<span class="color-dark">{{request.user.username}}</span>
```

Busque as especialidades cadastradas e envie para o HTML:

```
especialidades = Especialidades.objects.all()
return render(request, 'cadastro_medico.html', {'especialidades': especialidades})
```

Liste as especialidades no HTML:

```
{% for especialidade in especialidades %}
  <option value="{{especialidade.id}}">{{especialidade}}</option>
{% endfor %}
```

Configure o form:

```
<form action="{% url 'cadastro_medico' %}" method="post" enctype='multipart/form-data'{% csrf_token %}
```

Realize o cadastro médico em cadastro\_medico:

```
@login_required
def cadastro_medico(request):
    if request.method == "GET":
        especialidades = Especialidades.objects.all()
        return render(request, 'cadastro_medico.html', {'especialidades': especialidades})
    elif request.method == "POST":
        crm = request.POST.get('crm')
        nome = request.POST.get('nome')
        cep = request.POST.get('cep')
        rua = request.POST.get('rua')
        bairro = request.POST.get('bairro')
        numero = request.POST.get('numero')
        cim = request.FILES.get('cim')
        rg = request.FILES.get('rg')
        foto = request.FILES.get('foto')
        especialidade = request.POST.get('especialidade')
        descricao = request.POST.get('descricao')
        valor_consulta = request.POST.get('valor_consulta')
```



```
#TODO: Validar todos os campos

dados_medico = DadosMedico(
    crm=crm,
    nome=nome,
    cep=cep,
    rua=rua,
    bairro=bairro,
    numero=numero,
    rg=rg,
    cedula_identidade_medica=cim,
    foto=foto,
    user=request.user,
    descricao=descricao,
    especialidade_id=especialidade,
    valor_consulta=valor_consulta
)
dados_medico.save()

messages.add_message(request, constants.SUCCESS, 'Cadastro médico realizado com sucesso.')

return redirect('/medicos/abrir_horario')
```

Crie uma função que verifica se um determinado usuário é médico ou não:

```
def is_medico(user):
    return DadosMedico.objects.filter(user=user).exists()
```

Se um usuário já for médico não o deixe realizar o cadastro médico novamente:

```
if is_medico(request.user):
    messages.add_message(request, constants.WARNING, 'Você já está cadastrado como médico.')
    return redirect('/medicos/abrir_horario')
```