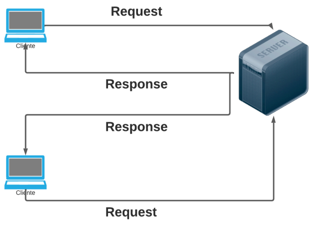
Aula 1

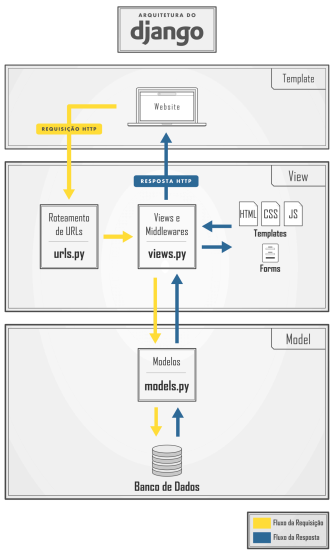
Acesse diretamente pelo Notion:

```
https://grizzly-amaranthus-f6a.notion.site/Aula-1-1c16cf8ea89f80d1989dd93f8197e254?pvs=4
```

▼ Teoria



Fluxo de dados no Django:



▼ 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 core .
```

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 usuarios

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

Em usuarios/urls.py crie a URL para o cadastro

```
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 os arquivos de templates

```
BASE_DIR / 'templates'
```

Configure os arquivos de media

```


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/'
```

Para o HTML, antes vamos configurar um base.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Mentorfy</title>
  <script src="https://unpkg.com/@tailwindcss/browser@4"></script>
  {% block 'head' %}{% endblock 'head' %}
</head>
<body>
  {% block 'body' %}{% endblock 'body' %}
</body>
</html>
```

E crie o cadastro.html

```
{% extends "base.html" %}
{% block 'body' %}

<div class="flex min-h-screen bg-slate-900">
  <div class="flex flex-1 flex-col justify-center px-4 py-12 sm:px-6 lg:flex-none lg:px-20 xl:px-24">
   <div class="mx-auto w-full max-w-sm lg:w-96">
    <div>

     <img class="h-15 w-auto" src="" alt="Logo">

     <h2 class="mt-4 text-2xl/9 font-bold tracking-tight text-gray-100">Crie sua conta</h2>
     <p class="mt-2 text-sm/6 text-gray-500">
      Já tem uma conta?
      <a href="" class="font-semibold text-indigo-300 hover:text-indigo-200">Faça login</a>
     </p>

    </div>

    <div class="mt-8">
     <div>
      <form action="" method="" class="space-y-6">
       <div>
        <label for="email" class="block text-sm/6 font-medium text-gray-200">Username</label>
        <div class="mt-2">
          <input type="text" name="username" required class="block w-full rounded-md bg-white/5 px-3 py-1.5 text-base text-white outline outline-1 -outline-offset-1 outline-white/10 placeholder:text-gray-500 focus:outline focus:outline-2 focus:-outline-offset-2 focus:outline-indigo-500 sm:text-sm/6">
         </div>
       </div>

       <div>
        <label for="password" class="block text-sm/6 font-medium text-gray-200">Senha</label>
        <div class="mt-2">
          <input type="password" name="senha"  required class="block w-full rounded-md bg-white/5 px-3 py-1.5 text-base text-white outline outline-1 -outline-offset-1 outline-white/10 placeholder:text-gray-500 focus:outline focus:outline-2 focus:-outline-offset-2 focus:outline-indigo-500 sm:text-sm/6">
         </div>
       </div>
       <div>
        <label for="password" class="block text-sm/6 font-medium text-gray-200">Confirmar senha</label>
        <div class="mt-2">
          <input type="password" name="confirmar_senha" required class="block w-full rounded-md bg-white/5 px-3 py-1.5 text-base text-white outline outline-1 -outline-offset-1 outline-white/10 placeholder:text-gray-500 focus:outline focus:outline-2 focus:-outline-offset-2 focus:outline-indigo-500 sm:text-sm/6">
         </div>
```

```
          </div>


          <div>
            <button type="submit" class="flex w-full justify-center cursor-pointer rounded-md bg-indigo-600 px-3 py-1.5 text-sm/6 font-semibold text-white shadow-sm hover:bg-indigo-500 focus-visible:outline focus-visible:outline-2 focus-visible:outline-offset-2 focus-visible:outline-indigo-600">Cadastrar</button>
          </div>
        </form>
      </div>


    </div>
   </div>
  </div>
  <div class="relative hidden w-0 flex-1 lg:block">
   <img class="absolute inset-0 size-full object-cover" src="https://images.unsplash.com/photo-1541746972996-4e0b0f43e02a?q=80&w=2670&auto=format&fit=crop&ixlib=rb-4.0.3&ixid=M3wxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8fA%3D%3D" alt="">

  </div>
 </div>

{% endblock 'body' %}
```

https://drive.google.com/file/d/1WXMGD0rEJ0dd3H88894Ox0S9SIxtnRPh/view?usp=sharing

Adicione a imagem da logo:

```
{% static 'logo.png' %}
```

Execute as migrações para criar as tabelas no banco de dados.

Envie os dados do form para o back-end

```
<form action="{% url 'cadastro' %}" method="POST" class="space-y-6">{% csrf_token %}
```

No back-end receba e processe os dados:

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

        if not senha == confirmar_senha:
            return redirect('/usuarios/cadastro')

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

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

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

        User.objects.create_user(
            username=username,
            password=senha
        )


        return redirect('/usuarios/login')
```

Configure as messages em settings.py

```
from django.contrib.messages import constants

MESSAGE_TAGS = {
    constants.SUCCESS: 'bg-green-50 text-green-700',
    constants.ERROR: 'bg-red-50 text-red-700'
}
```

Liste as mensagens no HTML

```
{% if messages %}
    {% for message in messages %}
        <div class="rounded-md {{message.tags}} mt-4">
            <div class="flex">
                <div class="ml-3 py-4">
                    {{message}}

                </div>

            </div>
        </div>
    {% endfor %}
{% endif %}
```

Login

Crie uma URL para o login

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

Desenvolva a view login

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

Crie o HTML

```
{% extends "base.html" %}
{% load static %}
{% block 'body' %}
<div class="flex min-h-screen bg-slate-900">
    <div class="flex flex-1 flex-col justify-center px-4 py-12 sm:px-6 lg:flex-none lg:px-20 xl:px-24">
      <div class="mx-auto w-full max-w-sm lg:w-96">
        <div>
          <img class="h-15 w-auto" src="{% static 'logo.png' %}" alt="Your Company">
          <h2 class="mt-4 text-2xl/9 font-bold tracking-tight text-gray-100">Entre em sua conta</h2>
          <p class="mt-2 text-sm/6 text-gray-500">
            Não tem conta?
            <a href="{% url 'cadastro' %}" class="font-semibold text-indigo-300 hover:text-indigo-200">Crie sua conta aqui</a>
          </p>

          {% if messages %}
              {% for message in messages %}
                  <div class="rounded-md {{message.tags}} mt-4">
                      <div class="flex">
                          <div class="ml-3 py-4">
                              {{message}}

                          </div>

                      </div>
                  </div>
              {% endfor %}
          {% endif %}
        </div>

        <div class="mt-8">
          <div>
            <form action="{% url 'login' %}" method="POST" class="space-y-6">{% csrf_token %}
              <div>
                <label class="block text-sm/6 font-medium text-gray-200">Username</label>
                <div class="mt-2">
                    <input type="text" name="username" required class="block w-full rounded-md bg-white/5 px-3 py-1.5 text-base text-white outline outline-1 -outline-offset-1 outline-white/10 placeholder:text-gray-500 focus:outline focus:outline-2 focus:-outline-offset-2 focus:outline-indigo-500 sm:text-sm/6">
                  </div>
              </div>

              <div>
                <label for="password" class="block text-sm/6 font-medium text-gray-200">Senha</label>
                <div class="mt-2">
                    <input type="password" name="senha" required class="block w-full rounded-md bg-white/5 px-3 py-1.5 text-base text-white outline outline-1 -outline-offset-1 outline-white/10 placeholder:text-gray-500 focus:outline focus:outline-2 focus:-outline-offset-2 focus:outline-indigo-500 sm:text-sm/6">
                  </div>
              </div>


              <div>
                <button type="submit" class="flex w-full justify-center cursor-pointer rounded-md bg-indigo-600 px-3 py-1.5 text-sm/6 font-semibold text-white shadow-sm hover:bg-indigo-500 focus-visible:outline focus-visible:outline-2 focus-visible:outline-offset-2 focus-visible:outline-indigo-600">Entrar</button>
              </div>
            </form>
          </div>


        </div>
      </div>
    </div>
    <div class="relative hidden w-0 flex-1 lg:block">
      <img class="absolute inset-0 size-full object-cover" src="https://images.unsplash.com/photo-1541746972996-4e0b0f43e02a?q=80&w=2670&auto=format&fit=crop&ixlib=rb-4.0.3&ixid=M3wxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8fA%3D%3D" alt="">

    </div>
  </div>

{% endblock 'body' %}
```

Receba e processe todos os dados

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

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

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

        messages.add_message(request, constants.ERROR, 'Username ou senha inválidos')
        return redirect('login')
```

Mentorados

Crie um novo app para gerenciar os mentorados

```
python manage.py startapp mentorados
```

Instale o APP!

Aponte uma URL para mentorados

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

Crie a URL para a view

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

urlpatterns = [
    path('', views.mentorados, name='mentorados'),
]
```

Crie o HTML

```
{% extends "base.html" %}
{% load static %}
{% block 'body' %}
<header class="bg-slate-900">
    <nav class="flex items-center justify-between p-4 lg:px-8" aria-label="Global">
      <div class="flex lg:flex-1">
```

```
     <a href="#" class="-m-1.5 p-1.5">
      <span class="sr-only">Your Company</span>
      <img class="h-8 w-auto" src="{% static 'logo.png' %}" alt="">
     </a>
    </div>
    <div class="flex lg:hidden">
     <button type="button" class="-m-2.5 inline-flex items-center justify-center rounded-md p-2.5 text-gray-700">
      <span class="sr-only">Open main menu</span>
      <svg class="size-6" fill="none" viewBox="0 0 24 24" stroke-width="1.5" stroke="currentColor" aria-hidden="true" data-slot="icon">
       <path stroke-linecap="round" stroke-linejoin="round" d="M3.75 6.75h16.5M3.75 12h16.5m-16.5 5.25h16.5" />
      </svg>
     </button>
    </div>
    <div class="hidden lg:flex lg:gap-x-12">
     <a href="#" class="text-sm/6 font-semibold text-gray-100">Mentorados</a>
     <a href="#" class="text-sm/6 font-semibold text-gray-100">Tarefas</a>
     <a href="#" class="text-sm/6 font-semibold text-gray-100">Progressos</a>
     <a href="#" class="text-sm/6 font-semibold text-gray-100">Navigators</a>
    </div>
    <div class="hidden lg:flex lg:flex-1 lg:justify-end">

    </div>
   </nav>

   </div>
  </header>


  <div class="bg-[#040e1b] min-h-screen">
   <div class="max-w-7xl mx-auto py-12">
     <div class="grid grid-cols-2 gap-4">
       <div>
        <form action="" method="">
         <h2 class="mt-4 text-2xl/9 font-bold tracking-tight text-gray-100">Cadastre seus mentorados</h2>
         {% if messages %}
           {% for message in messages %}
             <div class="rounded-md {{message.tags}} mt-4">
               <div class="flex">
                 <div class="ml-3 py-4">
                   {{message}}

                 </div>

               </div>
             </div>
           {% endfor %}
         {% endif %}
         <div>
           <label for="email" class="block text-sm/6 font-medium text-gray-200">Nome</label>
           <div class="mt-2">
             <input type="text" name="nome" required class="block w-full rounded-md bg-white/5 px-3 py-1.5 text-base text-white outline outline-1 -outline-offset-1 outline-white/10 placeholder:text-gray-500 focus:outline focus:outline-2 focus:-outline-offset-2 focus:outline-indigo-500 sm:text-sm/6">
            </div>
         </div>
         <div class="grid grid-cols-2 gap-4 mt-4">
           <div>
             <label class="block text-sm/6 font-medium text-gray-200">Foto</label>
             <div class="mt-2">
               <input type="file" name="foto" id="foto" autocomplete="foto" required class="block w-full rounded-md bg-white/5 px-3 py-1.5 text-base text-white outline outline-1 -outline-offset-1 outline-white/10 placeholder:text-gray-500 focus:outline focus:outline-2 focus:-outline-offset-2 focus:outline-indigo-500 sm:text-sm/6">
             </div>
           </div>
           <div>
             <label class="block text-sm/6 font-medium text-gray-200">Estágio</label>
             <div class="mt-2">
               <select name="estagio" id="" class="block w-full rounded-md bg-white/5 px-3 py-2 text-base text-white outline outline-1 -outline-offset-1 outline-white/10 placeholder:text-gray-500 focus:outline focus:outline-2 focus:-outline-offset-2 focus:outline-indigo-500 sm:text-sm/6">

                 <option class="text-slate-900" value="">10-1000k</option>

               </select>

             </div>
           </div>

         </div>

         <div class="mt-4">
             <label class="block text-sm/6 font-medium text-gray-200">Navigator</label>
             <div class="mt-2">
               <select name="navigator" id="" class="block w-full rounded-md bg-white/5 px-3 py-2 text-base text-white outline outline-1 -outline-offset-1 outline-white/10 placeholder:text-gray-500 focus:outline focus:outline-2 focus:-outline-offset-2 focus:outline-indigo-500 sm:text-sm/6">

                 <option value="">Caio</option>

               </select>

             </div>
             <br>
             <button type="submit" class="flex w-full justify-center cursor-pointer rounded-md bg-indigo-600 px-3 py-1.5 text-sm/6 font-semibold text-white shadow-sm hover:bg-indigo-500 focus-visible:outline focus-visible:outline-2 focus-visible:outline-offset-2 focus-visible:outline-indigo-600">Cadastrar</button>
         </div>
        </form>
       </div>

       <div class="flex justify-center items-center w-1/2 mx-auto">
         <canvas id="myChart" class="w-24 h-24"></canvas>
       </div>


     </div>
     <div class="mt-6">
       <hr class="border-gray-600">
       <br>
       <h2 class="text-base/7 font-semibold text-white ">Seus mentorados</h2>
       <table class="mt-6 w-full whitespace-nowrap text-left">
        <colgroup>
         <col class="w-full sm:w-4/12">
         <col class="lg:w-4/12">
         <col class="lg:w-2/12">
         <col class="lg:w-1/12">
         <col class="lg:w-1/12">
        </colgroup>
        <thead class="border-b border-white/10 text-sm/6 text-white">
         <tr>
          <th scope="col" class="py-2 pl-4 pr-8 font-semibold sm:pl-6 lg:pl-8">Usuário</th>
          <th scope="col" class="hidden py-2 pl-0 pr-8 font-semibold sm:table-cell">Estágio</th>
          <th scope="col" class="py-2 pl-0 pr-4 text-right font-semibold sm:pr-8 sm:text-left lg:pr-20">Status</th>
          <th scope="col" class="hidden py-2 pl-0 pr-8 font-semibold md:table-cell lg:pr-20">Data entrada</th>
          <th scope="col" class="hidden py-2 pl-0 pr-8 font-semibold md:table-cell lg:pr-20">Navigator</th>
         </tr>
        </thead>
        <tbody class="divide-y divide-white/5">


          <tr>
           <td class="py-4 pl-4 pr-8 sm:pl-6 lg:pl-8">
            <div class="flex items-center gap-x-4">
             <img src="#" alt="" class="size-8 rounded-full bg-gray-800">
             <div class="truncate text-sm/6 font-medium text-white">Caio Sampaio</div>
            </div>
           </td>
           <td class="hidden py-4 pl-0 pr-4 sm:table-cell sm:pr-8">
            <div class="flex gap-x-3">
             <div class="rounded-md bg-gray-700/40 px-2 py-1 text-xs font-medium text-gray-400 ring-1 ring-inset ring-white/10">10-100k</div>
            </div>
           </td>
           <td class="py-4 pl-0 pr-4 text-sm/6 sm:pr-8 lg:pr-20">
            <div class="flex items-center justify-end gap-x-2 sm:justify-start">
             <div class="flex-none rounded-full bg-green-400/10 p-1 text-green-400">
              <div class="size-1.5 rounded-full bg-current"></div>
             </div>
             <div class="hidden text-white sm:block">Ativo</div>
            </div>
           </td>
           <td class="hidden py-4 pl-0 pr-8 text-sm/6 text-gray-400 md:table-cell lg:pr-20">26/03/2025</td>
           <td class="hidden py-4 pl-0 pr-8 text-sm/6 text-gray-400 md:table-cell lg:pr-20">

             Caio Sampaio
           </td>
          </tr>

        </tbody>
       </table>
      </div>

   </div>

  </div>


 {% endblock 'body' %}
```

Vamos para a etapa do banco de dados, primeiro crie as models

```
from django.db import models
from django.contrib.auth.models import User

class Navigators(models.Model):
    nome = models.CharField(max_length=255)
    user = models.ForeignKey(User, on_delete=models.CASCADE)

    def __str__(self):
        return self.nome

class Mentorados(models.Model):
    estagio_choices = (
        ('E1', '10-100k'),
        ('E2', '100-1KK')
    )
    nome = models.CharField(max_length=255)
    foto = models.ImageField(upload_to='fotos', null=True, blank=True)
    estagio = models.CharField(max_length=2, choices=estagio_choices)
    navigator = models.ForeignKey(Navigators, null=True, blank=True, on_delete=models.SET_NULL)
    user = models.ForeignKey(User, on_delete=models.CASCADE)
    criado_em = models.DateField(auto_now_add=True)

    def __str__(self):
        return self.nome
```

Execute as migrações!

Envie dados dos navigators e dos mentorados para o HTML

```
def mentorados(request):
    if request.method == 'GET':
        navigators = Navigators.objects.filter(user=request.user)
        return render(request, 'mentorados.html', {'estagios': Mentorados.estagio_choices, 'navigators': navigators})
```

Adicione no HTML

```
{% for estagio in estagios  %}
  <option class="text-slate-900" value="{{estagio.0}}">{{estagio.1}}</option>
{% endfor %}

{% for navigator in navigators  %}
  <option value="{{navigator.id}}">{{navigator.nome}}</option>
{% endfor %}
```

Envie os dados do forms para o back-end

```
<form action="{% url 'mentorados' %}" method="POST" enctype='multipart/form-data'>{% csrf_token %}
```

Processe os dados no back-end

```
def mentorados(request):
    if request.method == 'GET':
        navigators = Navigators.objects.filter(user=request.user)
        return render(request, 'mentorados.html', {'estagios': Mentorados.estagio_choices, 'navigators': navigators})
    else:
        nome = request.POST.get('nome')
        foto = request.FILES.get('foto')
        estagio = request.POST.get("estagio")
        navigator = request.POST.get('navigator')

        mentorado = Mentorados(
            nome=nome,
            foto=foto,
            estagio=estagio,
            navigator_id=navigator,
            user=request.user
        )

        mentorado.save()

        messages.add_message(request, constants.SUCCESS, 'Mentorado cadastrado com sucesso.')
        return redirect('mentorados')
```

Busque no banco todos os mentorados

```
mentorados = Mentorados.objects.filter(user=request.user)
```

Liste todos os mentorados

```
{% for mentorado in mentorados %}
  <tr>
    <td class="py-4 pl-4 pr-8 sm:pl-6 lg:pl-8">
      <div class="flex items-center gap-x-4">
        <img src="{{mentorado.foto.url}}" alt="" class="size-8 rounded-full bg-gray-800">
        <div class="truncate text-sm/6 font-medium text-white">{{mentorado.nome}}</div>
      </div>
    </td>
    <td class="hidden py-4 pl-0 pr-4 sm:table-cell sm:pr-8">
      <div class="flex gap-x-3">
        <div class="rounded-md bg-gray-700/40 px-2 py-1 text-xs font-medium text-gray-400 ring-1 ring-inset ring-white/10">{{mentorado.estagio}}</div>
      </div>
    </td>
    <td class="py-4 pl-0 pr-4 text-sm/6 sm:pr-8 lg:pr-20">
      <div class="flex items-center justify-end gap-x-2 sm:justify-start">
        <div class="flex-none rounded-full bg-green-400/10 p-1 text-green-400">
          <div class="size-1.5 rounded-full bg-current"></div>
        </div>
        <div class="hidden text-white sm:block">Ativo</div>
      </div>
    </td>
    <td class="hidden py-4 pl-0 pr-8 text-sm/6 text-gray-400 md:table-cell lg:pr-20">{{mentorado.criado_em}}</td>
    <td class="hidden py-4 pl-0 pr-8 text-sm/6 text-gray-400 md:table-cell lg:pr-20">

      {{mentorado.navigator.nome}}

    </td>
  </tr>
{% endfor %}
```

Configure as URL’s para media

```
from django.contrib import admin
from django.urls import path, include
from django.conf import settings
from django.conf.urls.static import static

urlpatterns = [
    path('admin/', admin.site.urls),
    path('usuarios/', include('usuarios.urls')),
    path('mentorados/', include('mentorados.urls'))
] + static(settings.MEDIA_URL, document_root=settings.MEDIA_ROOT)
```

▼ Gráficos

Adicione nas views os cálculos para os gráficos

```
def mentorados(request):
    if request.method == 'GET':
        navigators = Navigators.objects.filter(user=request.user)
        mentorados = Mentorados.objects.filter(user=request.user)

        estagios_flat = [i[1] for i in Mentorados.estagio_choices]
        qtd_estagios = []

        for i, j in Mentorados.estagio_choices:
            qtd_estagios.append(Mentorados.objects.filter(estagio=i).count())


        return render(request, 'mentorados.html', {'estagios': Mentorados.estagio_choices, 'navigators': navigators, 'mentorados': mentorados, 'estagios_flat': estagios_flat, 'qtd_estagios': qtd_estagios})
```

Adicione o JavaScript

```
<script src="https://cdn.jsdelivr.net/npm/chart.js"></script>

<script>
  const ctx = document.getElementById('myChart');

  new Chart(ctx, {
    type: 'pie',
    data: {
      labels: {{estagios_flat|safe}},
      datasets: [{
        label: '',
        data: {{qtd_estagios|safe}},
        borderWidth: 1
      }]
    },

  });
</script>
```