

# PSW 8.0 | Aula 3

Acesso o material pelo Notion:

<https://grizzly-amaranthus-f6a.notion.site/PSW-8-0-Aula-3-8325cc9dd6de4a49ab2219be57111572?pvs=4>

## ▼ Gerar acesso médico

Crie a model onde iremos armazenar os links de acessos médicos:

```
class AcessoMedico(models.Model): usuario =  
models.ForeignKey(User, on_delete=models.DO_NOTHING)  
identificacao = models.CharField(max_length=50) tempo_de_acesso  
= models.IntegerField() # Em horas criado_em =  
models.DateTimeField() data_examens_iniciais = models.DateField()  
data_examens_finais = models.DateField() token =  
models.CharField(max_length=20) def __str__(self): return  
self.token def save(self, *args, **kwargs): if not self.token:  
self.token = token_urlsafe(6) super(AcessoMedico,  
self).save(*args, **kwargs)
```

Execute as migrações e cadastre no ADMIN!

Crie a URL onde o paciente irá gerar o link para seu médico:

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

Crie a view:

```
@login_required def gerar_acesso_medico(request): if
request.method == "GET": acessos_medicos =
AcessoMedico.objects.filter(usuario =request. user) return
render(request, 'gerar_acesso_medico.html', {'acessos_medicos':
acessos_medicos}) elif request.method == "POST": identificacao =
request.POST.get('identificacao') tempo_de_acesso =
request.POST.get('tempo_de_acesso') data_exame_inicial =
request.POST.get("data_exame_inicial") data_exame_final =
request.POST.get("data_exame_final") acesso_medico =
AcessoMedico( usuario = request.user, identificacao =
identificacao, tempo_de_acesso = tempo_de_acesso,
data_examenes_iniciais = data_exame_inicial, data_examenes_finais =
data_exame_final, criado_em = datetime.now() )
acesso_medico.save() messages.add_message(request,
constants.SUCCESS, 'Acesso gerado com sucesso') return
redirect('/examenes/gerar_acesso_medico')
```

Crie o HTML gerar\_acesso\_medico:

```
{% extends "bases/base.html" %} {% load static %} {% block
'head' %} <link href="{% static 'exames/css/css.css' %}"
rel="stylesheet"> <link href="{% static
'exames/css/gerenciar_pedidos.css' %}" rel="stylesheet"> {%
endblock 'head' %} {% block 'conteudo' %} <br> <br> <div
class="container"> {% if messages %} <br> {% for message in
messages %} <div class="alert {{ message.tags }}">{{ message }}
</div> {% endfor %} {% endif %} <div class="row"> <div
class="col-md-5"> <h3>Gerar acesso médico</h3> <form action="{%
url "gerar_acesso_medico" %}" method="POST">{% csrf_token %}
<label for="">Identificação</label> <br> <input style="width:
100%" type="text" name="identificacao" id="" class="input-
default"> <br> <br> <label for="">Tempo de acesso (em horas)
</label> <br> <input style="width: 100%" type="number"
name="tempo_de_acesso" id="" class="input-default"> <br> <br>
<div class="row"> <label for="">Acesso a exames realizados
entre:</label> <div class="col-md"> <input style="width: 100%"
type="date" name="data_exame_inicial" id="" class="input-
default"> </div> <div class="col-md"> <input style="width: 100%"
type="date" name="data_exame_final" id="" class="input-default">
<br> <br> </div> </div> <input style="width: 40%" type="submit"
class="btn-default" value="Gerar link"> </form> </div> <div
class="col-md"> <div class="card"> <table style="text-align:
center"> <tr> <th>Id</th> <th>Status</th> <th>Link</th> </tr>
<tr class="linha-tabela"> <td>01</td> <td>Status 1</td> <td>URL
AQUI</td> </tr> </table> </div> </div> </div> </div> {% endblock
'conteudo' %}
```

Exiba os acesso dinamicamente:

```
{% for acesso in acessos_medicos %} <tr class="linha-tabela">
<td>{{ acesso.identificacao }}</td> <td>{{ acesso.status }}</td>
<td><a href="{{ acesso.url }}">{{ acesso.url }}</a></td> </tr> {%
endfor %}
```

Crie os métodos da model:

```
@property def status(self): return 'Expirado' if timezone.now()
> (self.criado_em + timedelta(hours=self.tempo_de_acesso)) else
'Ativo' @property def url(self): #TODO: reverse return
f"http://127.0.0.1:8000/exames/acesso_medico/{self.token}"
```

Crie a URL onde o médico irá acessar:

```
path('acesso_medico/<str:token>', views.acesso_medico,
name="acesso_medico"),
```

Crie a view:

```
def acesso_medico(request, token): acesso_medico =
AcessoMedico.objects.get(token = token) if acesso_medico.status ==
'Expirado': messages.add_message(request, constants.WARNING, 'Esse
já se expirou!') return redirect('/usuarios/login') pedidos =
PedidosExames.objects.filter(data__gte =
acesso_medico.data_exames_iniciais).filter(data__lte =
acesso_medico.data_exames_finais).filter(usuario=acesso_medico.usua
return render(request, 'acesso_medico.html', {'pedidos': pedidos})
```

Por fim, crie o HTML do médico:

```
{% extends "bases/base.html" %} {% load static %} {% block
'head' %} <link href="{% static 'exames/css/css.css' %}"
rel="stylesheet"> <link href="{% static
'exames/css/gerenciar_pedidos.css' %}" rel="stylesheet"> {%
endblock 'head' %} {% block 'conteudo' %} <br> <br> <div
class="container"> <table style=""> <tr> <th>Id</th>
<th>Exame</th> </tr> <tr class="linha-tabela"> <td>Nome</td>
<td>Em análise</td> </tr> </table> </div> {% endblock 'conteudo'
%}
```

Liste dinamicamente os exames:

```
{% for pedido in pedidos %} {% for exame in pedido.exames.all %}
<tr class="linha-tabela"> <td>{{exame.exame.nome}}</td> {% if
exame.resultado %} <td><a style="text-decoration: none;"
class="btn-secundario" href="{{exame.resultado.url}}">Ver
exame</a></td> {% else %} <td>Em análise</td> {% endif %} </tr>
{% endfor %} {% endfor %}
```

## ▼ Gerenciar clientes

Crie o app empresarial, onde os funcionários da empresa irão trabalhar:

```
python manage.py startapp empresarial
```

### INSTALE O APP!

Crie a URL empresarial:

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

Em empresarial/urls.py crie:

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

Crie a view gerenciar\_clientes:

```

from django.shortcuts import render from
django.contrib.auth.models import User from django.db.models
import Value from django.db.models.functions import Concat from
django.contrib.admin.views.decorators import
staff_member_required @staff_member_required def
gerenciar_clientes(request): clientes =
User.objects.filter(is_staff=False) nome_completo =
request.GET.get('nome') email = request.GET.get('email') if
email: clientes = clientes.filter(email__contains = email) if
nome_completo: clientes = clientes.annotate(
full_name=Concat('first_name', Value(' '), 'last_name')
).filter(full_name__contains=nome_completo) return
render(request, 'gerenciar_clientes.html', {'clientes':
clientes, 'nome_completo': nome_completo, 'email': email})

```

Desenvolva o HTML gerenciar\_clientes:

```

{% extends "bases/base.html" %} {% load static %} {% block
'head' %} <link href="{% static 'exames/css/css.css' %}"
rel="stylesheet"> <link href="{% static
'exames/css/gerenciar_pedidos.css' %}" rel="stylesheet"> {%
endblock 'head' %} {% block 'conteudo' %} <br> <br> <div
class="container"> {% if messages %} <br> {% for message in
messages %} <div class="alert {{ message.tags }}">{{ message }}
</div> {% endfor %} {% endif %} <div class="row"> <div
class="col-md"> <form action="{% url "gerenciar_clientes" %}"
method="GET"> <label for="">Nome</label> <input type="text"
class="form-control" name="nome" value="{{ nome_completo }}">
</div> <div class="col-md"> <label for="">E-mail</label> <input
type="text" class="form-control" name="email" value="{{ email }}">
</div> <div class="col-md"> <br> <input type="submit"
class="btn-default"> </form> </div> </div> <br> <div class="card
card-view"> <table> <tr> <th>Nome</th> <th>E-mail</th>
<th>CPF</th> </tr> {% for cliente in clientes %} <tr
class="linha-tabela"> <td><a href="#">{{ cliente.get_full_name }}
</a></td> <td>{{ cliente.email }}</td> <td> </td> </tr> {% endfor
%} </table> </div> </div> {% endblock 'conteudo' %}

```

Crie a URL onde o admin irá acessar um paciente em específico:



```
path('cliente/<int:cliente_id>', views.cliente, name="cliente"),
```

Desenvolva a view:

```
@staff_member_required def cliente(request, cliente_id): cliente
= User.objects.get(id=cliente_id) exames =
SolicitacaoExame.objects.filter(usuario=cliente) return
render(request, 'cliente.html', {'cliente': cliente, 'exames':
exames})
```

Crie o HTML:

```
{% extends "bases/base.html" %} {% load static %} {% block
'head' %} <link href="{% static 'exames/css/css.css' %}"
rel="stylesheet"> <link href="{% static
'exames/css/gerenciar_pedidos.css' %}" rel="stylesheet"> <link
href="{% static 'exames/css/gerenciar_exames.css' %}"
rel="stylesheet"> {% endblock 'head' %} {% block 'conteudo' %}
<br> <br> <div class="container"> {% if messages %} <br> {% for
message in messages %} <div class="alert {{ message.tags }}">{{
message }}</div> {% endfor %} {% endif %} <h3 class="font-
destaque">{{cliente.get_full_name}}</h3> <p>{{cliente.email}}
</p> <div class="card card-view"> <div class="sub-card">
<h3>Exames de sangue</h3> <hr> <br> {% for exame in exames %} {%
if exame.exame.tipo == 'S'%} <div class="row"> <div class="col-
md"><h3>{{exame.exame.nome}}</h3></div> <div class="col-md">
{{exame.badge_template}}</div> <div class="col-md"> <a href="#"
class="btn btn-light">Abrir</a> </div> </div> <br> {% endif %}
{% endfor %} </div> <br> <br> <div class="sub-card"> <h3>Exames
de imagem</h3> <hr> <br> {% for exame in exames %} {% if
exame.exame.tipo == 'I'%} <div class="row"> <div class="col-md">
<h3>{{exame.exame.nome}}</h3></div> <div class="col-md">
{{exame.badge_template}}</div> <div class="col-md"><a href="#"
class="btn btn-light">Abrir</a></div> </div> <br> {% endif %} {%
endfor %} </div> </div> </div> {% endblock 'conteudo' %}
```

Em gerenciar\_clientes redirecione para tela do cliente:

```
<td><a href="{% url 'cliente' cliente.id %}">
{{cliente.get_full_name}}</a></td>
```

## ▼ Gerenciar exames dos clientes

Crie a URL:

```
path('exame_cliente/<int:exame_id>', views.exame_cliente,  
name="exame_cliente"),
```

Crie a view `exame_cliente`:

```
@staff_member_required def exame_cliente(request, exame_id):  
    exame = SolicitacaoExame.objects.get(id=exame_id) return  
    render(request, 'exame_cliente.html', {'exame': exame})
```

Crie o HTML `exame_cliente.html`

```
{% extends "bases/base.html" %} {% load static %} {% block
'head' %} <link href="{% static 'exames/css/css.css' %}"
rel="stylesheet"> <link href="{% static
'exames/css/gerenciar_pedidos.css' %}" rel="stylesheet"> <link
href="{% static 'exames/css/gerenciar_exames.css' %}"
rel="stylesheet"> {% endblock 'head' %} {% block 'conteudo' %}
<br> <div class="container"> {% if messages %} <br> {% for
message in messages %} <div class="alert {{ message.tags }}">{{
message }}</div> {% endfor %} {% endif %} <div class="row"> <h3
class="font-destaque">{{exame.usuario.get_full_name}}</h3> <p>
{{exame.usuario.email}}</p> <br> <div class="row"> <div
class="col-md-6"> <form action="" method="POST"> <input
type="FILE" value="Alterar PDF" class="btn-secundario"
name="resultado"> <br> <br> {% if exame.resultado %} <a
style="width: 100%;" href="" target="__blank"><div style="width:
100%;" id="pdf-container">Seu exame aqui</div></a> {% endif %}
</div> <div class="col-md-6"> <h3>Status</h3> <select
name="status" id="" class="form-select"> <option style="color:
black;" value="E">Em análise</option> <option style="color:
black;" value="F">Finalizado</option> </select> <br> <input
type="checkbox" name="requer_senha" id=""><label for="">Requer
senha para acessar ?</label> <br> <br> <input type="submit"
value="Salvar" class="btn-default"> </form> <a style="text-
decoration: none;" href="" class="btn-secundario">Gerar
senha</a> </div> </div> </div> {% endblock 'conteudo' %}
```

Em cliente.html redirecione para a nova URL:

```
<a href="{% url "exame_cliente" exame.id %}" class="btn btn-
light">Abrir</a>
```

Para exibir o PDF do exame valor utilizar um JavaScript:

```
<script
src="https://cdnjs.cloudflare.com/ajax/libs/pdf.js/2.11.338/pdf.mir
</script> <script> const container = document.getElementById('pdf-
container'); pdfjsLib.getDocument('').promise.then(pdf => {
pdf.getPage(1).then(page => { const canvas =
document.createElement('canvas'); const context =
canvas.getContext('2d'); const viewport = page.getViewport({ scale:
}); canvas.width = viewport.width; canvas.height = viewport.height;
page.render({ canvasContext: context, viewport }).promise.then(() =
container.appendChild(canvas); }); }); }); </script>
```

Para funcionar crie a proxy\_pdf:

```
path('proxy_pdf/<int:exame_id>', views.proxy_pdf,
name="proxy_pdf"),
```

E sua respectiva VIEW:

```
@staff_member_required def proxy_pdf(request, exame_id): exame =
SolicitacaoExame.objects.get(id=exame_id) response =
exame.resultado.open() return FileResponse(response)
```

✓ **Correção**