

Repositorio GitHub: <https://github.com/ThiagoYuri/Projeto-Padaria-IESB/tree/main>

CMD Exec: python3.11 manage.py runserver

Introdução: Esse projeto foi feito para agendamento de pedidos e dos clientes para a padaria. Utilizando Django e outros frameworks.

1. Banco de dados db.sqlite:

1.1. Criar models: Cliente, Pedidos, Produto, ItensPedido

```
app_padaria > models.py > ItensPedido
4
5 # Create your models here.
6 class Cliente(models.Model):
7     id_cliente = models.AutoField(primary_key=True)
8     nome = models.TextField()
9     telefone = models.IntegerField()
10
11
12 # Create your models here.
13 class Pedidos(models.Model):
14     id_pedido = models.AutoField(primary_key=True)
15     nome = models.TextField()
16     id_cliente = models.ForeignKey(Cliente, on_delete=models.PROTECT) ## No cascade
17
18 class Produto(models.Model):
19     id_produto = models.AutoField(primary_key=True)
20     nome = models.TextField()
21     preco = models.FloatField()
22
23
24 class ItensPedido(models.Model):
25     pedido = models.ForeignKey(Pedidos, on_delete=models.PROTECT)
26     produto = models.ForeignKey(Produto, on_delete=models.PROTECT)
27     quantidade = models.PositiveIntegerField()
28
29
30
```

1.2. Config setting

```
DATABASES = {
    "default": {
        "ENGINE": "django.db.backends.sqlite3",
        "NAME": BASE_DIR / "db.sqlite3",
    }
}
```

1.3. Criando migration:

CMD:

python manage.py makemigrations <name>

python manage.py migrate

1.4. Todas as tabelas geradas pelo migration:

Name	Type	Schema
▼ Tables (15)		
> app_padaria_cliente		CREATE TABLE "app_padaria_cliente" ("id_cliente" integer NOT NULL PRIMARY KEY AI
> app_padaria_itenspedido		CREATE TABLE "app_padaria_itenspedido" ("id" integer NOT NULL PRIMARY KEY AUTI
> app_padaria_pedidos		CREATE TABLE "app_padaria_pedidos" ("id_pedido" integer NOT NULL PRIMARY KEY ,
> app_padaria_produto		CREATE TABLE "app_padaria_produto" ("id_produto" integer NOT NULL PRIMARY KEY
> auth_group		CREATE TABLE "auth_group" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT,
> auth_group_permissions		CREATE TABLE "auth_group_permissions" ("id" integer NOT NULL PRIMARY KEY AUTC
> auth_permission		CREATE TABLE "auth_permission" ("id" integer NOT NULL PRIMARY KEY AUTOINCREM
> auth_user		CREATE TABLE "auth_user" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, '
> auth_user_groups		CREATE TABLE "auth_user_groups" ("id" integer NOT NULL PRIMARY KEY AUTOINCRE
> auth_user_user_permissions		CREATE TABLE "auth_user_user_permissions" ("id" integer NOT NULL PRIMARY KEY A
> django_admin_log		CREATE TABLE "django_admin_log" ("id" integer NOT NULL PRIMARY KEY AUTOINCR
> django_content_type		CREATE TABLE "django_content_type" ("id" integer NOT NULL PRIMARY KEY AUTOINC
> django_migrations		CREATE TABLE "django_migrations" ("id" integer NOT NULL PRIMARY KEY AUTOINCR
> django_session		CREATE TABLE "django_session" ("session_key" varchar(40) NOT NULL PRIMARY KEY
> sqlite_sequence		CREATE TABLE sqlite_sequence(name,seq)
▼ Indices (18)		

2. Views

Home = NavBar

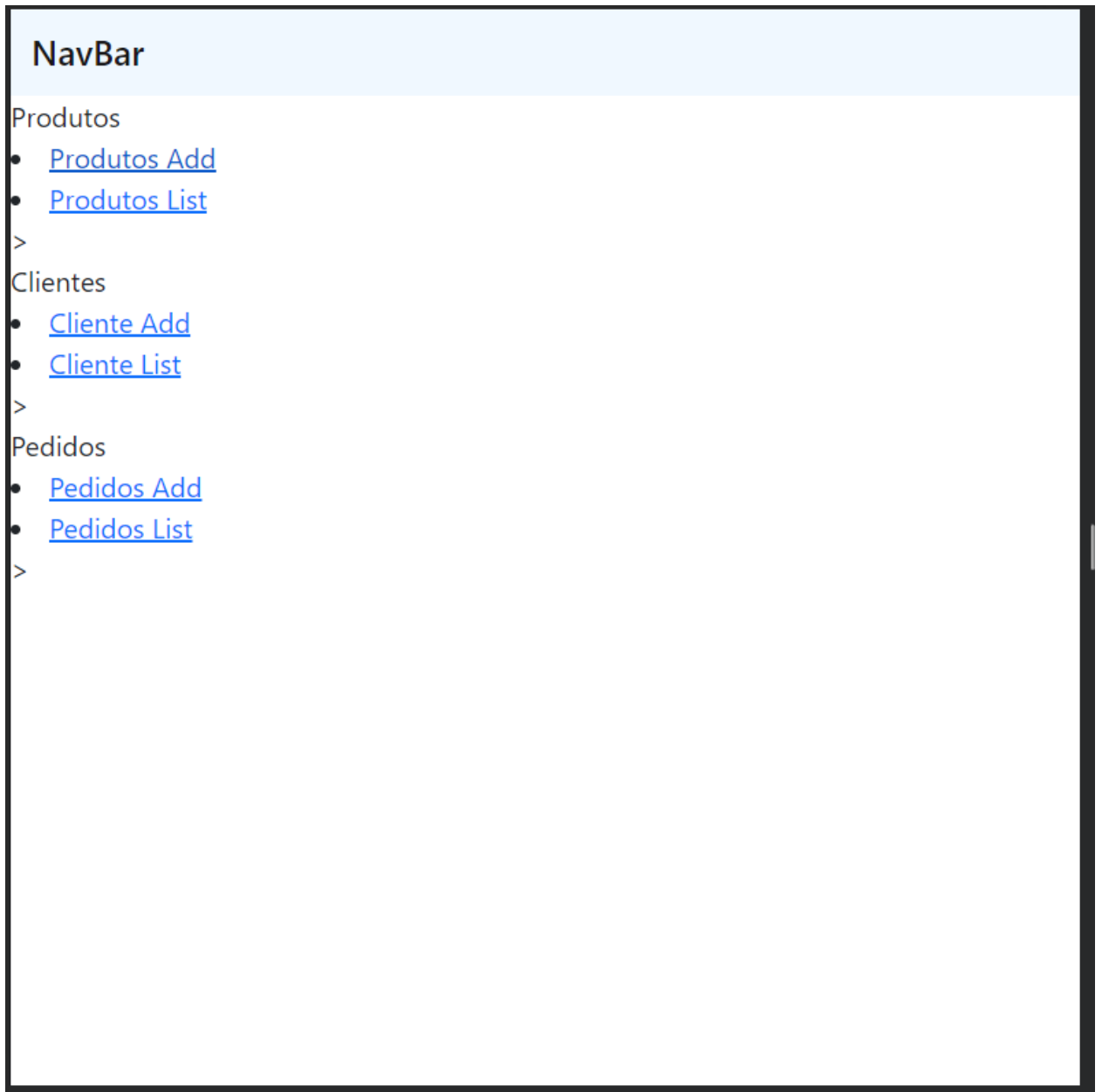
```
▼ templates \ padaria
  ▼ cliente
    <> clienteAdd.html
    <> clienteList.html
  ▼ pedidos
    <> pedidoAdd.html
    <> pedidoList.html
  > produtos
  <> home.html
```

3. URL's

```
0
1 urlpatterns = [
2     # rota, view responsavel. nome de referência
3     path("", views.home, name="home"),
4     # Clientes
5     path("clientesAdd/", views.clientesAdd, name="add_clientes"),
6     path("clientesAddPost/", views.clientesAddPost, name="clientesAddPost"),
7     path("clientes/", views.clientes, name="listagem_clientes"),
8     # Pedidos
9     path("pedidosAdd/", views.pedidosAdd, name="add_pedidos"),
10    path("pedidosAddPost/", views.pedidosAddPost, name="pedidosAddPost"),
11    path("pedidos/", views.pedidos, name="listagem_pedidos"),
12
13    # Produtos
14    path("produtosAdd/", views.produtosAdd, name="add_produtos"),
15    path("produtosAddPost/", views.produtosAddPost, name="produtosAddPost"),
16    path("produtos/", views.produtos, name="listagem_produtos"),
17 ]
18
```

4. Projeto Final

4.1. Home NavBar:



4.2. Clientes:

Add

Padaria

Back Home

Cadastro de Clientes

Nome: Telefone:

Enviar

List

Back Home

id	nome	telefone
40	Thiago Yuri	6474744757

4.3. Produtos:

Add

Padaria

Back Home

Cadastro de Produto

Nome: Preço:

Enviar

Back Home

id_produto nome preco

List

Clientes:

4.4. Pedidos:

Add

Padaria

Back Home

Cadastro de Clientes

id_pedido:

status_pedidos:

id_cliente:

Enviar

Back Home

id_pedido Status Pedido id_cliente

List

Features faltando:

- Adicionar no pedido o campo de data de resgate do pedido.
- Finalizar o Crud do pedido e do produto.
- Criar testes para o sistema Web.
- Ajustar desing.