

Relatório do Projeto: BuyPy - Backoffice para Loja Online

1. Introdução e Objetivos

O projeto **BuyPy** tem como objetivo criar uma aplicação de backoffice para uma loja online, permitindo a gestão de produtos, clientes e encomendas. O sistema foi pensado para operadores administrativos que necessitam de uma interface simples e eficaz para interagir com a base de dados da loja.

Foram definidos três principais componentes:

- **Base de dados relacional MySQL**, com estrutura adequada à gestão de um e-commerce.
- **Gestão de acessos e permissões**, com perfis distintos para operadores, administradores e clientes.
- **Interface gráfica em Python (com PyQt6)**, para facilitar a utilização do sistema no dia a dia.

2. Desenho e Estrutura

2.1 Base de Dados

A base de dados criada chama-se **BuyPy** e contém tabelas organizadas por temas:

- **Produto**: inclui livros e produtos eletrónicos, com campos específicos por tipo.
- **Cliente, Encomenda e ItemEncomenda**: representam os dados principais das compras realizadas.
- **Autor, LivroAutor**: definem a estrutura N:N entre livros e autores.
- **Recomendacao**: relaciona produtos sugeridos a clientes.
- **Operador**: responsáveis pela administração da loja.

Foram ainda criados três perfis com permissões específicas:

- **WEB_CLIENT** (apenas leitura + encomendas),
- **BUYDB_OPERATOR** (leitura/escrita geral),
- **BUYDB_ADMIN** (acesso total e gestão de privilégios).

2.2 Interface Gráfica

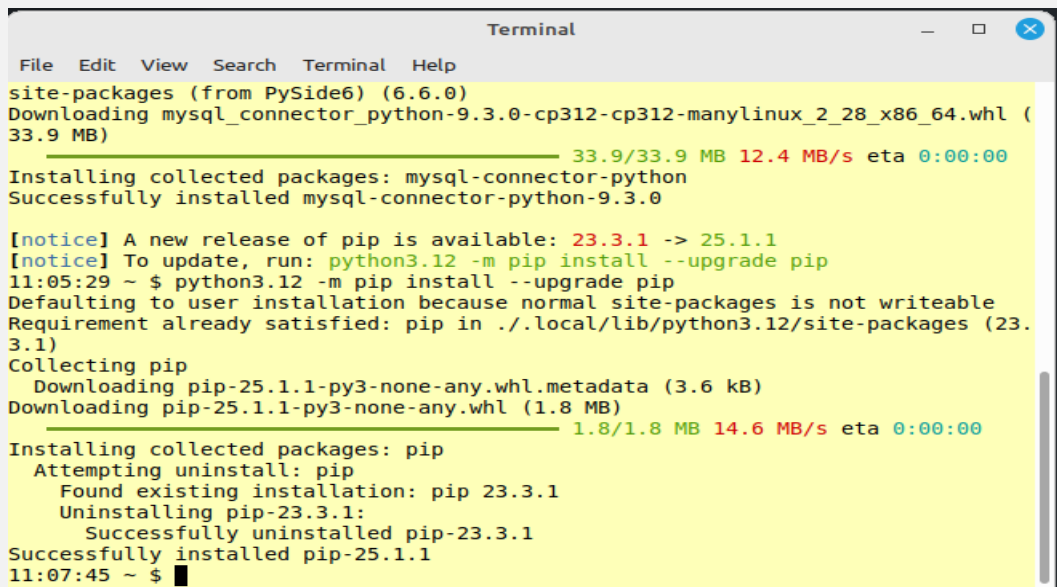
A interface foi desenvolvida em **Python com PyQt6**, com ligação direta à base de dados via `mysql.connector`. Ela permite:

- Visualizar e editar clientes;
- Criar e visualizar encomendas;
- Atualizar a quantidade de produtos (restrito);
- Visualizar listas de produtos e recomendações.

3. Desenvolvimento

O projeto foi dividido em três fases principais:

- **Modelação e criação da base de dados**, com estrutura relacional, dados de teste e stored procedures.
- **Gestão de utilizadores e permissões**, utilizando o mecanismo de *roles* do MySQL para facilitar a manutenção.
- **Implementação da interface gráfica**, que se conecta à base de dados e realiza operações CRUD conforme as permissões do utilizador autenticado.
- Importação de tabela CSV



```
Terminal
File Edit View Search Terminal Help
site-packages (from PySide6) (6.6.0)
Downloading mysql_connector_python-9.3.0-cp312-cp312-manylinux_2_28_x86_64.whl (
33.9 MB)
33.9/33.9 MB 12.4 MB/s eta 0:00:00
Installing collected packages: mysql-connector-python
Successfully installed mysql-connector-python-9.3.0

[notice] A new release of pip is available: 23.3.1 -> 25.1.1
[notice] To update, run: python3.12 -m pip install --upgrade pip
11:05:29 ~ $ python3.12 -m pip install --upgrade pip
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: pip in ./local/lib/python3.12/site-packages (23.
3.1)
Collecting pip
  Downloading pip-25.1.1-py3-none-any.whl.metadata (3.6 kB)
  Downloading pip-25.1.1-py3-none-any.whl (1.8 MB)
  1.8/1.8 MB 14.6 MB/s eta 0:00:00
Installing collected packages: pip
  Attempting uninstall: pip
    Found existing installation: pip 23.3.1
    Uninstalling pip-23.3.1:
      Successfully uninstalled pip-23.3.1
  Successfully installed pip-25.1.1
11:07:45 ~ $
```

Procedimentos criados:

- CreateOrder
- AddProductToOrder
- GetOrderTotal

Estes procedimentos são usados pela interface para gerir o processo de compra.

4. Testes e Validação

Durante o desenvolvimento, foram realizados testes para garantir o bom funcionamento:

- Inserção e atualização de produtos e clientes;
- Criação de encomendas associando produtos e clientes;
- Restrições de acesso funcionando corretamente por perfil;
- Interface gráfica funcionando com ações reais no banco de dados;
- Teste das procedures com diferentes entradas e simulação de casos reais.

Exemplo de teste:

Perfil WEB_CLIENT só consegue editar encomendas e clientes – tentativas de alterar preços ou apagar produtos resultam em erro de permissão, como esperado.

5. Conclusão e Trabalho Futuro

O projeto BuyPy cumpriu todos os objetivos propostos. A base de dados está normalizada, segura e com permissões corretamente atribuídas. A interface facilita a gestão da loja e comunica corretamente com o banco.

Pontos fortes:

- Segurança por perfil.
- Separação clara entre produtos do tipo livro e eletrónicos.
- Integração funcional entre backend (MySQL) e frontend (PyQt6).

Melhorias futuras:

- Criar tela de login com autenticação por tipo de utilizador.
- Implementar notificações para produtos em baixo estoque.
- Migrar para ambiente web com Flask ou Django.