

Relatório 2 - Linguagem de Programação Python

Edryck Freitas Nascimento

1. Introdução

O objetivo desta aula foi aprender o básico sobre a linguagem de programação Python. A tarefa consistia em fazer um relatório sobre o conteúdo aprendido no minicurso solicitado no *card*, apresentar o código gerado na aula e um código aplicando o conteúdo aprendido no minicurso.

2. Desenvolvimento

O minicurso foi separado em duas partes:

- **Primeira parte:** Tipos de variáveis, sintaxe, operadores, tipos de condições e laços de repetição.
- **Segunda parte:** Funções e orientação à objetos em Python.

Após a conclusão do mini-curso, desenvolvi um projeto básico em Python para auxiliar na fixação do conteúdo aprendido. O projeto consistiu em um sistema básico de gerenciamento de uma biblioteca.

- **Classes e Funcionalidades:** Adicionei um sistema de login bem simples, apenas para diferenciar o leitor do bibliotecário. O usuário pode pesquisar livros e realizar login ou cadastro caso não esteja cadastrado no sistema. Usei herança para criar o usuário, tendo como classes filhos:
 - O **bibliotecário** tem as seguintes permissões: cadastrar, editar e remover livros, editar e deletar usuários.
 - O **leitor** tem as seguintes permissões: pegar e devolver livro.

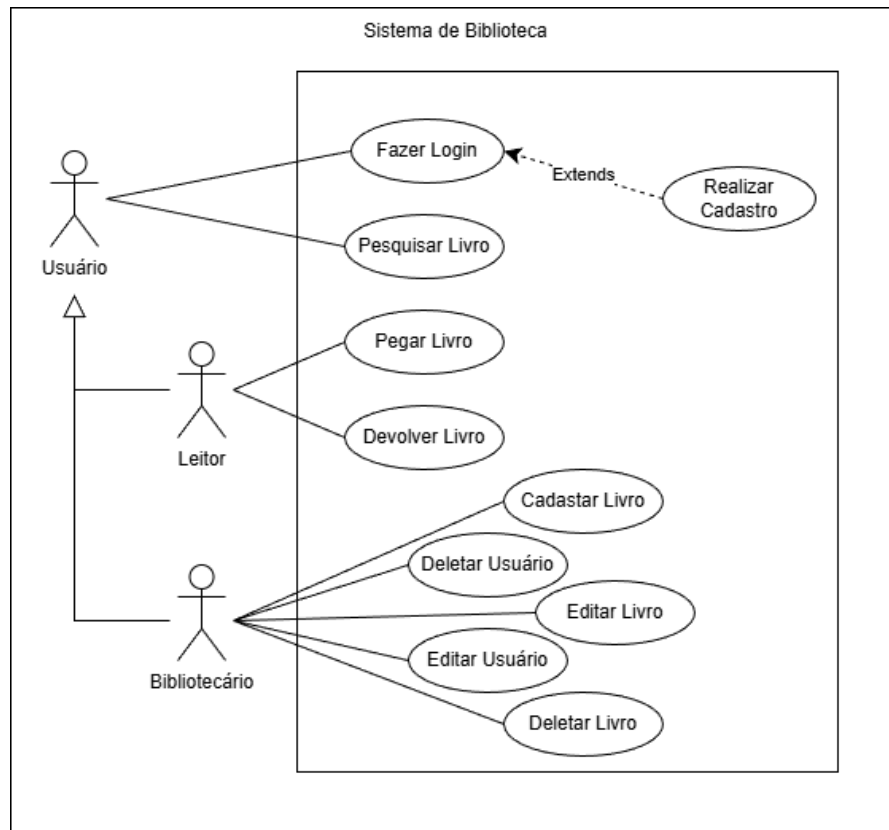


Diagrama 1: Casos de Usos do sistema (autoria própria)

- **Arquitetura e recursos:**

- **Arquitetura:** Model-View-Controller (MVC) básica.
- **Linguagem:** Python 3.13.
- **Interface:** Tkinter que é uma biblioteca padrão e integrada do Python para criar as interfaces gráficas, aliás, para usei o código do site Devmedia¹ como base para as primeiras telas.
- **Banco de Dados:** SQLite. Inicialmente pensei em usar um arquivo JSON para armazenar as informações, mas troque para o SQLite, pois no site também mostrava como usar.

- **Estrutura de pastas:** Tudo se inicia na main.py, ela apenas chama a tela de login, após o usuário logar, vai mostrar a tela do admin ou do leitor, dependendo de quem logar. Segue o seguinte fluxo: usuário logado interage com a tela, chama a função do controller (onde está as regras de negócio), se passar das validações, chama a função na pasta dao, que faz as interações com o banco de dados, e retorna até a camada view. Caso algo dê errado, apenas irá mostrar ao usuário mensagem de erro, caso contrário mostra sucesso (não implementei um feedback bom ao usuário para casos de erros). Os diagramas de pacote e sequência:

¹ <https://www.devmedia.com.br/tkinter-interfaces-graficas-em-python/33956>

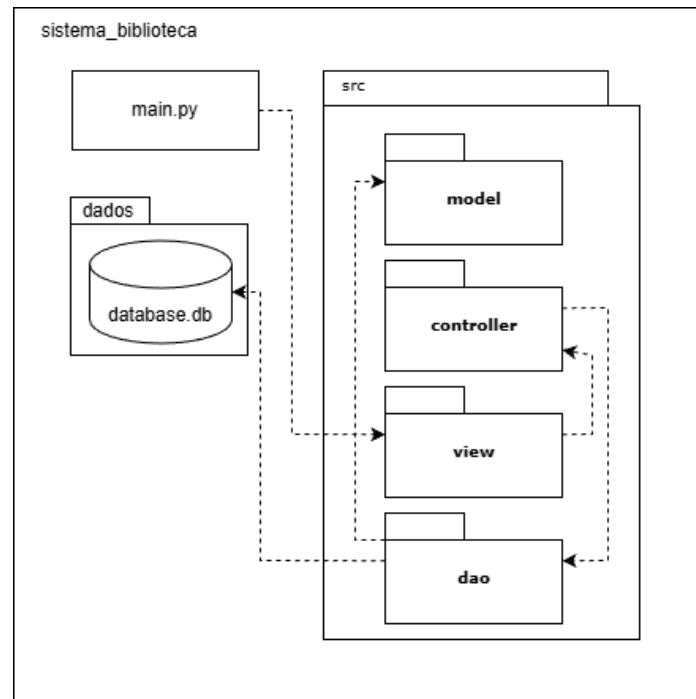


Diagrama 2: Estruturas de pastas e suas interações (autoria própria)

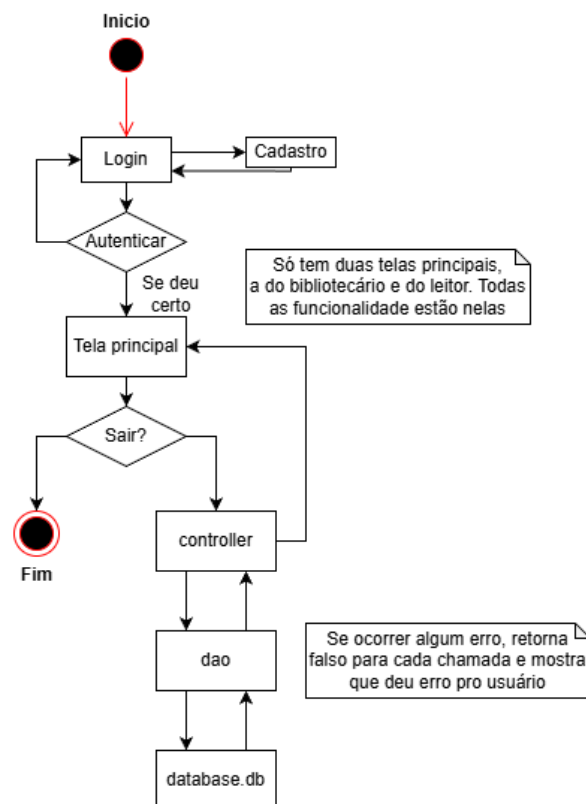


Diagrama 3: Diagrama simplificado de sequência (autoria própria)

- **Desenvolvimento da interface e banco de dados:** Como dito anteriormente, usei a biblioteca Tkinter, que é nativa do Python, e o SQLite para o banco de dados. Ouve um certo desafio em fazer a interface, muitas horas lendo a

documentação² dele, já o banco de dados foi mais fácil, pois já tinha uma certa experiência em projetos anteriores.

3. Conclusão

O desenvolvimento desse projeto ajudou praticar os conceitos de Python e Orientação a Objetos que foram abordados no minicurso. A aplicação da arquitetura MVC com o banco de dados SQLite e à interface gráfica com Tkinter, tornou-se em um sistema que organiza as permissões em Bibliotecário e Leitores. O projeto cumpriu o objetivo de demonstrar o conhecimento adquirido no minicurso e da pesquisa feita para fazer uma interface gráfica.

Código do Projeto:

<https://github.com/Edryck/Sistema-de-Biblioteca>

Referências:

Devmedia, “Tkinter: Interfaces gráficas em Python”, <https://www.devmedia.com.br/tkinter-interfaces-graficas-em-python/33956>

Python, “tkinter — Interface Python para Tcl/Tk”, <https://docs.python.org/pt-br/3.14/library/tkinter.html>

Cod3r Cursos, “PYTHON 3 Curso Rápido 🐍 Parte #1 2020 - 100% Prático!”, <https://www.youtube.com/watch?v=oUrBHiT-lzo>

Cod3r Cursos, “PYTHON 3 Curso Rápido 🐍 Parte #2 2020 - 100% Prático!”, <https://www.youtube.com/watch?v=iq7JLIH-sV0>

² <https://docs.python.org/pt-br/3.14/library/tkinter.html>