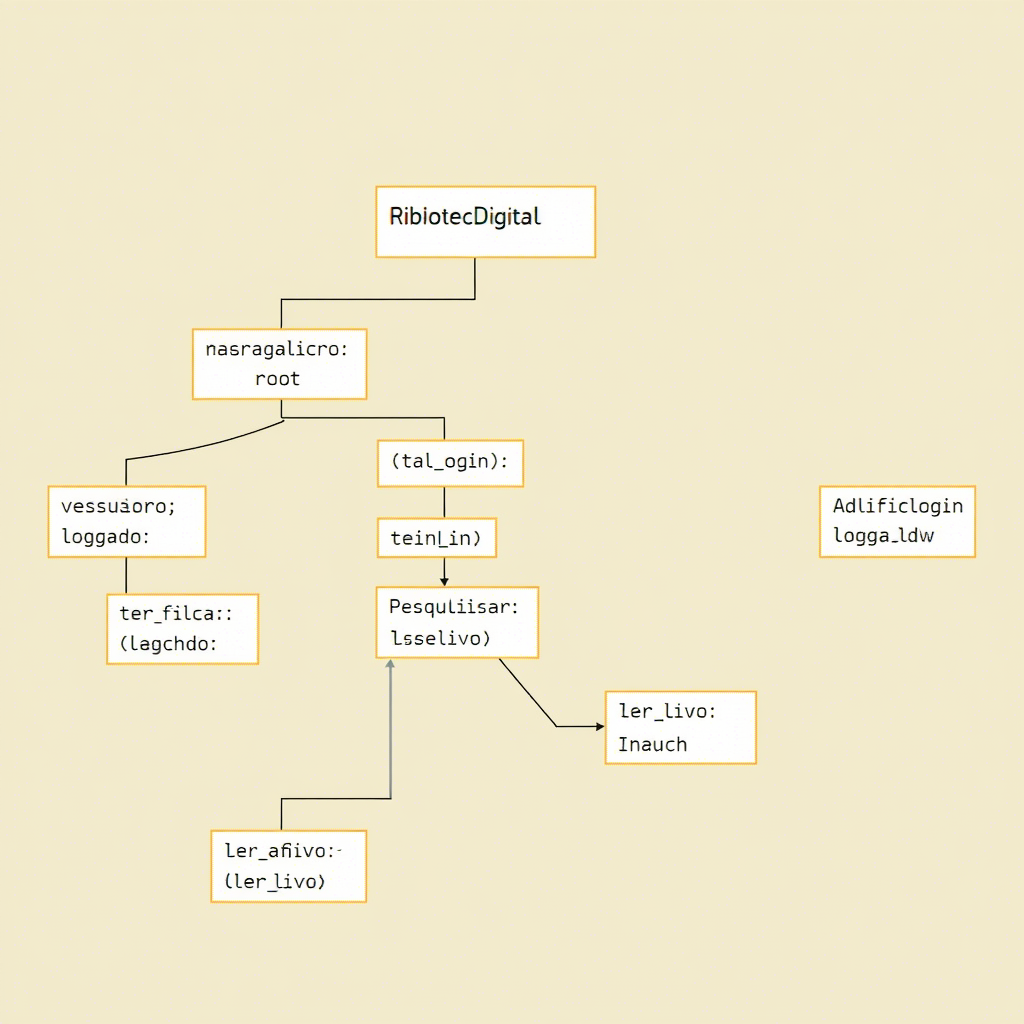
1. **Diagrama de Classe**

O diagrama de classe representa a estrutura do sistema em termos de classes, atributos e métodos. Abaixo está o diagrama simplificado da aplicação:



**Explicação:**

* Classe**BibliotecaDigital** : Representa a aplicação principal.
  + Atributos :
    - **root**: A janela principal da interface gráfica.
    - **arquivo\_livros**: O caminho do arquivo JSON onde os livros são salvos.
    - **livros**: Um dicionário que armazena os livros (nome: caminho).
    - **usuario\_logado**: Indica se o usuário está logado.
  + Métodos :
    - Métodos como , , e implementam as funcionalidades do sistema.**tela\_loginadicionar\_livropesquisar\_livro**

1. **Trechos de Código Importantes**

**Tela de Login**

def tela\_login(self):

self.limpar\_tela()

tk.Label(self.root, text="Login", font=("Arial", 24), bg="#FFD7B5").pack(pady=20)

tk.Label(self.root, text="Usuário:", bg="#FFD7B5").pack()

self.usuario\_entry = tk.Entry(self.root, width=30)

self.usuario\_entry.pack(pady=5)

tk.Label(self.root, text="Senha:", bg="#FFD7B5").pack()

self.senha\_entry = tk.Entry(self.root, width=30, show="\*")

self.senha\_entry.pack(pady=5)

tk.Button(self.root, text="Entrar", command=self.verificar\_login, bg="#FFA500", fg="white").pack(pady=10)

**Adicionar Livro**

def adicionar\_livro(self):

arquivo = filedialog.askopenfilename(title="Selecionar Livro", filetypes=[("Arquivos PDF", "\*.pdf")])

if arquivo:

nome\_livro = os.path.basename(arquivo)

self.livros[nome\_livro] = arquivo

self.salvar\_livros()

self.atualizar\_lista\_livros()

messagebox.showinfo("Sucesso", f"Livro '{nome\_livro}' adicionado à biblioteca.")

**Pesquisar Livro**

def pesquisar\_livro(self):

termo = self.pesquisa\_entry.get().strip().lower()

if not termo:

self.atualizar\_lista\_livros()

return

livros\_filtrados = {nome: path for nome, path in self.livros.items() if termo in nome.lower()}

self.lista\_livros\_text.delete(1.0, tk.END)

if not livros\_filtrados:

self.lista\_livros\_text.insert(tk.END, "Nenhum livro encontrado.\n")

else:

for idx, livro in enumerate(livros\_filtrados.keys(), start=1):

self.lista\_livros\_text.insert(tk.END, f"{idx}. {livro}\n")

**3. Funcionalidades Implementadas**

1. Tela de login :
   * Autenticação com credenciais fixas ( , ).**usuário: 20212025**, **senha: 0123456789fn**
2. Adicionar Livros :
   * Permitir ao usuário adicionar livros em formato PDF ao sistema.
   * Os livros são salvos no dicionário e persistidos no arquivo JSON.**self.livros**
3. Barra de Pesquisa :
   * Filtre os livros pelo nome digitado na barra de pesquisa.
4. Leitura de Livros :
   * Extrai o texto de um livro PDF selecionado e exibido em uma nova janela.
5. Download de Livros :
   * Permite baixar o livro selecionado para o computador.
6. Persistência de Dados :
   * Os livros adicionados são salvos no arquivo e carregados automaticamente ao iniciar o programa.**livros.json**

**4. Ferramentas Usadas**

1. Linguagem de Programação :
   * Python (versão 3.x).
2. Bibliotecas :
   * **tkinter**: Para criar uma interface gráfica.
   * **PyPDF2**: Para manipular arquivos PDF (ler e extrair texto).
   * **json**: Para salvar e carregar dados no formato JSON.
3. Ambiente de Desenvolvimento :
   * Visual Studio Code ou PyCharm.
4. Formato de Arquivos :
   * Arquivos PDF para os livros.
   * Arquivo JSON( ) para persistir os dados.**livros.json**

**5. Requisitos Funcionais**

1. RF1 - Tela de Login :
   * O sistema deve permitir o login do usuário com credenciais específicas.
2. RF2 - Adicionar Livros :
   * O sistema deve permitir ao usuário adicionar livros em formato PDF.
3. RF3 - Pesquisar Livros :
   * O sistema deve permitir ao usuário pesquisar livros pelo nome.
4. RF4 - Ler Livros :
   * O sistema deve permitir ao usuário visualizar o conteúdo de um livro PDF.
5. RF5 - Baixar Livros :
   * O sistema deve permitir ao usuário baixar um livro para o computador.
6. RF6 - Persistência de Dados :
   * O sistema deve salvar os livros adicionais para que permaneçam disponíveis após o fechamento do programa.

**6. Requisitos Não Funcionais**

1. RNF1 - Interface Amigável :
   * A interface deve ser intuitiva e fácil de usar, com núcleos claros e botões bem posicionados.
2. RNF2 - Desempenho :
   * O sistema deve carregar rapidamente os livros e processar as transações sem lentidão.
3. RNF3 - Segurança :
   * As credenciais de login devem ser verificadas corretamente.
4. RNF4 - Portabilidade :
   * O sistema deve funcionar em diferentes sistemas operacionais (Windows, Linux, macOS).
5. RNF5 - Escalabilidade :
   * O sistema deve suportar a adição de vários livros sem perda de desempenho.

**7. Casos de Uso**

**Caso de Uso 1: Login**

* Ator : Usuário.
* Pré-condição : O sistema está iniciado.
* Fluxo Principal :
  1. O usuário insira o nome de usuário e senha.
  2. O sistema verifica as credenciais.
  3. Se as credenciais forem corretas, o usuário será redirecionado para a tela principal.
* Fluxo Alternativo :
  1. Se as credenciais estiverem incorretas, o sistema exibirá uma mensagem de erro.

**Caso de Uso 2: Adicionar Livro**

* Ator : Usuário.
* Pré-condição : O usuário está logado.
* Fluxo Principal :
  1. O usuário clicou no botão "Adicionar Livro".
  2. O sistema abre uma janela para selecionar um arquivo PDF.
  3. O sistema adiciona o livro à lista e salva no arquivo JSON.
* Pós-condição : O livro é exibido na lista de livros.

**Caso de Uso 3: Pesquisar Livro**

* Ator : Usuário.
* Pré-condição : O usuário está logado e há livros adicionados.
* Fluxo Principal :
  1. O usuário digita um termo na barra de pesquisa.
  2. O sistema filtra os livros pelo termo.
  3. O sistema exibe os resultados na lista.
* Pós-condição : Apenas os livros correspondentes ao termo são propostos.

**Caso de Uso 4: Ler Livro**

* Ator : Usuário.
* Pré-condição : O usuário está logado e há livros adicionados.
* Fluxo Principal :
  1. O usuário seleciona um livro e clica em "Ler Livro Selecionado".
  2. O sistema extrai o texto do PDF e exibe em uma nova janela.
* Pós-condição : O conteúdo do livro é listado para leitura.

**Caso de Uso 5: Baixar Livro**

* Ator : Usuário.
* Pré-condição : O usuário está logado e há livros adicionados.
* Fluxo Principal :
  1. O usuário seleciona um livro e clica em "Baixar Livro Selecionado".
  2. O sistema abre uma janela para escolher o destino do arquivo.
  3. O sistema copia o arquivo PDF para o destino escolhido.
* Pós-condição : O livro foi baixado para o computador.