**FACULDADE MARTHA FALCÃO | WYDEN**  
**Curso:** Engenharia de Software – Turno Matutino  
**Disciplina:** Desenvolvimento Rápido de Aplicações em Python  
**Professor:** Ranyere Lima

Desenvolvimento de Diário Digital em Python Aplicando RAD: Manipulação de Arquivos, Banco de Dados e Interface Gráfica\*\*

## ****Aluno:**** Calil Fellipe Holanda Faria ****Período:**** 5º período ****Data de Entrega:**** 28/05/2025

## Sumário

1. Resumo ………………………………………………………. 2
2. Introdução …………………………………………………. 2
3. Fundamentação Teórica ……………………………….. 3 3.1 Python e Manipulação de Arquivos …………… 3 3.2 Python com Banco de Dados …………………… 4 3.3 Interface Gráfica em Python ……………………. 4 3.4 Desenvolvimento Rápido de Aplicações (RAD) ………………………………………………………………. 5
4. Metodologia e Desenvolvimento …………………… 6
5. Resultados (Gráficos e Tabelas) ……………………. 7
6. Conclusão …………………………………………………. 8
7. Referências ……………………………………………….. 9

## 1. Resumo

Este relatório descreve o desenvolvimento de uma aplicação em Python que permite registros pessoais em formato de diário digital. O projeto aborda conceitos fundamentais como manipulação de arquivos, uso de banco de dados SQLite e desenvolvimento de interface gráfica, aplicando a metodologia de Desenvolvimento Rápido de Aplicações (RAD). A aplicação oferece funcionalidades como adicionar registros com título, categoria e conteúdo, visualizar todos os registros, buscar por palavra-chave, filtrar por data, excluir registros e realizar backup dos dados. A interface gráfica foi desenvolvida utilizando a biblioteca Tkinter, proporcionando uma experiência de usuário intuitiva e agradável. O projeto demonstra a aplicação prática dos conceitos de programação em Python, resultando em uma ferramenta funcional e útil para o registro de pensamentos e informações pessoais.

## 2. Introdução

Com o avanço da tecnologia, a criação de aplicações simples, mas úteis, tornou-se uma excelente estratégia para introduzir os conceitos de programação. O Diário Digital desenvolvido neste projeto representa uma ferramenta prática que permite aos usuários registrarem seus pensamentos, anotações e eventos importantes de forma organizada e persistente.

A aplicação foi desenvolvida em Python, uma linguagem de programação conhecida por sua simplicidade e versatilidade, ideal para projetos educacionais e de desenvolvimento rápido. O projeto incorpora diversos conceitos fundamentais da programação moderna, como manipulação de arquivos, banco de dados relacional, interface gráfica e estruturas de controle.

O Diário Digital vai além de um simples registro de texto, oferecendo funcionalidades como categorização de entradas, busca por palavras-chave, filtragem por data, exclusão de registros e backup de dados. Estas características transformam o projeto em uma ferramenta completa para gerenciamento de informações pessoais.

A interface gráfica desenvolvida com Tkinter proporciona uma experiência de usuário intuitiva, com botões para cada funcionalidade e janelas de diálogo para entrada de dados. O design minimalista, com cores escuras, oferece um ambiente agradável para a escrita e leitura dos registros.

Este relatório documenta o processo de desenvolvimento do Diário Digital, desde sua concepção até a implementação final, destacando os conceitos aplicados, as decisões técnicas tomadas e os resultados obtidos. O projeto serve como exemplo prático da aplicação de conhecimentos teóricos em um produto funcional e útil.

## 3. Fundamentação Teórica

### 3.1 Python e Manipulação de Arquivos

Python oferece recursos robustos para manipulação de arquivos, permitindo ler, escrever e atualizar dados armazenados localmente. No contexto do Diário Digital, embora a implementação final utilize banco de dados SQLite, o conceito de manipulação de arquivos permanece fundamental para entender o armazenamento persistente de dados.

A manipulação de arquivos em Python é realizada principalmente através da função open(), que aceita parâmetros como o nome do arquivo e o modo de abertura. Os modos mais comuns incluem ‘r’ (leitura), ‘w’ (escrita, sobrescrevendo conteúdo existente) e ‘a’ (append, adicionando ao final do arquivo). Esta flexibilidade permite diferentes estratégias de armazenamento, dependendo das necessidades da aplicação.

No Diário Digital, além do armazenamento principal em banco de dados, a funcionalidade de backup implementa manipulação direta de arquivos, utilizando os módulos os e shutil para verificar a existência de diretórios e copiar arquivos. O código utiliza os.path.exists() para verificar se o diretório de backup existe e os.makedirs() para criá-lo caso necessário. Em seguida, shutil.copy() é empregado para copiar o arquivo de banco de dados para o diretório de backup, com um nome que inclui a data atual.

Esta abordagem demonstra como a manipulação de arquivos pode ser utilizada não apenas para o armazenamento primário de dados, mas também para funcionalidades complementares como backup e recuperação de informações.

### 3.2 Python com Banco de Dados

O SQLite é um sistema de gerenciamento de banco de dados relacional leve, que não requer um servidor separado e armazena todo o banco de dados em um único arquivo. Esta característica o torna ideal para aplicações desktop como o Diário Digital, onde a simplicidade de implementação e a portabilidade são importantes.

No projeto, o módulo sqlite3 é utilizado para criar e gerenciar o banco de dados. A conexão é estabelecida com sqlite3.connect('diario.db'), criando o arquivo de banco de dados se ele não existir. Em seguida, um cursor é obtido para executar comandos SQL.

A estrutura do banco de dados é definida com a criação da tabela registros, que contém campos para id (chave primária autoincremental), data, título, categoria e conteúdo. Esta estrutura permite organizar as entradas do diário de forma eficiente e facilita operações como busca e filtragem.

As operações CRUD (Create, Read, Update, Delete) são implementadas através de comandos SQL executados pelo cursor. A inserção de novos registros utiliza INSERT INTO, enquanto a leitura emprega SELECT com diferentes condições para recuperar todos os registros, filtrar por data ou buscar por palavra-chave. A exclusão de registros é realizada com o comando DELETE FROM.

O uso de banco de dados proporciona vantagens significativas em relação ao armazenamento em arquivos de texto simples, como estruturação dos dados, indexação para buscas eficientes e integridade referencial.

### 3.3 Interface Gráfica em Python

A biblioteca Tkinter é a interface padrão para o toolkit Tk do Python, fornecendo uma maneira rápida e fácil de criar aplicações gráficas. No Diário Digital, Tkinter é utilizado para desenvolver uma interface de usuário intuitiva e funcional.

A interface é construída a partir de um objeto Tk(), que representa a janela principal da aplicação. Elementos como rótulos (Label), botões (Button) e caixas de diálogo (messagebox, simpledialog) são utilizados para criar a experiência de usuário.

O projeto implementa um design com tema escuro, configurando cores de fundo (background) e texto (foreground) para criar uma estética moderna e agradável. O estilo é definido usando ttk.Style(), que permite personalizar a aparência dos widgets de forma consistente.

As caixas de diálogo são particularmente importantes na aplicação, pois permitem a interação com o usuário para entrada de dados. simpledialog.askstring() é utilizado para solicitar texto, como título, categoria e conteúdo dos registros, enquanto messagebox.showinfo() exibe informações e resultados de operações.

A organização da interface segue um layout vertical simples, com botões empilhados usando o método pack() com espaçamento adequado (pady). Esta abordagem cria uma navegação clara e direta para o usuário.

### 3.4 Desenvolvimento Rápido de Aplicações (RAD)

A metodologia de Desenvolvimento Rápido de Aplicações (RAD) é uma abordagem ágil focada na entrega rápida através de prototipação e ciclos iterativos. No contexto do Diário Digital, esta metodologia permitiu o desenvolvimento eficiente de uma aplicação funcional com recursos completos.

O RAD enfatiza o desenvolvimento incremental, onde funcionalidades são adicionadas progressivamente, testadas e refinadas. Esta abordagem é particularmente adequada para projetos educacionais, onde o foco está no aprendizado de conceitos e na aplicação prática.

No projeto, podemos identificar elementos do RAD na estrutura modular do código, com funções bem definidas para cada operação (adicionar, visualizar, buscar, excluir, backup), facilitando o desenvolvimento e teste de cada componente separadamente.

A escolha de ferramentas como SQLite e Tkinter também reflete princípios do RAD, pois são tecnologias que permitem prototipação rápida e resultados visíveis em curto prazo, sem necessidade de configurações complexas ou infraestrutura adicional.

A aplicação da metodologia RAD no Diário Digital resultou em um produto completo e funcional, que atende aos requisitos educacionais enquanto proporciona uma experiência de usuário satisfatória.

## 4. Metodologia e Desenvolvimento

O desenvolvimento do projeto Diário Digital seguiu etapas de planejamento, codificação, testes e validação. A metodologia adotada foi baseada nos princípios do Desenvolvimento Rápido de Aplicações (RAD), permitindo a criação de um produto funcional em um curto período de tempo.

Na fase inicial de planejamento, foram definidos os requisitos básicos da aplicação: permitir ao usuário registrar entradas com data e hora, visualizar registros anteriores e navegar por um menu interativo. Estes requisitos foram posteriormente expandidos para incluir funcionalidades adicionais como categorização, busca por palavra-chave, filtragem por data, exclusão de registros e backup de dados.

A codificação foi realizada em Python, aproveitando suas características de linguagem de alto nível e rica biblioteca padrão. O desenvolvimento seguiu uma abordagem modular, com funções específicas para cada operação do diário. Esta estrutura facilita a manutenção e expansão futura do código.

A implementação começou com a configuração do banco de dados SQLite, definindo a estrutura da tabela para armazenar os registros. Em seguida, foram desenvolvidas as funções para manipulação dos dados: adicionar, visualizar, buscar, filtrar, excluir e fazer backup. Cada função foi implementada de forma independente, permitindo testes isolados.

A interface gráfica foi desenvolvida utilizando Tkinter, com atenção especial à usabilidade e estética. O design adotou um tema escuro, com contraste adequado para facilitar a leitura e interação. Os elementos da interface foram organizados de forma lógica, com botões para cada funcionalidade principal.

Os testes foram realizados de forma incremental, verificando cada funcionalidade à medida que era implementada. Foram testados cenários como: - Adição de novos registros com diferentes títulos, categorias e conteúdos - Visualização de todos os registros armazenados - Busca por palavras-chave específicas no conteúdo - Filtragem de registros por data - Exclusão de registros específicos - Realização de backup e verificação da integridade dos dados

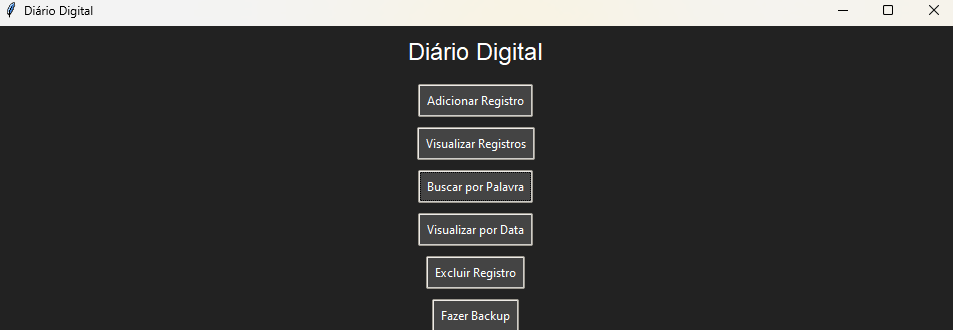
A validação final envolveu a execução de um fluxo completo de uso, simulando a experiência real do usuário. Foram identificados e corrigidos pequenos problemas de interface e comportamento, garantindo uma experiência fluida e intuitiva.

O resultado foi uma aplicação completa e funcional, que atende aos requisitos educacionais do projeto enquanto oferece uma ferramenta prática para o registro de pensamentos e informações pessoais.

## 5. Resultados (Gráficos e Tabelas)

A implementação do Diário Digital resultou em uma aplicação funcional com interface gráfica intuitiva e recursos completos para gerenciamento de registros pessoais. As imagens a seguir demonstram as principais funcionalidades e a aparência da aplicação.

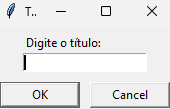
A tela principal do Diário Digital apresenta um design minimalista com fundo escuro e botões claramente identificados para cada funcionalidade. Esta interface foi desenvolvida utilizando Tkinter com personalização de estilo para criar uma experiência visual agradável e moderna.



Tela Principal do Diário Digital

A interface principal oferece seis opções principais: - Adicionar Registro: permite inserir uma nova entrada no diário - Visualizar Registros: exibe todas as entradas armazenadas - Buscar por Palavra: permite encontrar registros que contenham uma palavra específica - Visualizar por Data: filtra registros por uma data específica - Excluir Registro: remove uma entrada específica do diário - Fazer Backup: cria uma cópia de segurança do banco de dados

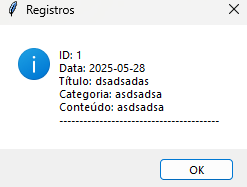
Ao selecionar a opção “Adicionar Registro”, o usuário é apresentado a uma série de caixas de diálogo para inserir as informações. A primeira solicita o título do registro, como mostrado na imagem abaixo:



Diálogo para Inserção de Título

Após inserir todas as informações necessárias (título, categoria e conteúdo), o registro é salvo no banco de dados com a data atual. A aplicação confirma o salvamento com uma mensagem de sucesso.

Quando o usuário seleciona “Visualizar Registros”, todas as entradas armazenadas são exibidas em uma janela de mensagem, como demonstrado na imagem a seguir:



Visualização de Registros

Cada registro é apresentado com seu ID, data, título, categoria e conteúdo, separados por uma linha de traços para facilitar a leitura. Esta visualização permite ao usuário revisar todas as entradas feitas anteriormente.

As funcionalidades de busca e filtragem seguem um padrão similar, solicitando a palavra-chave ou data através de caixas de diálogo e exibindo os resultados em uma janela de mensagem. A exclusão de registros requer o ID do registro a ser removido, garantindo que apenas a entrada desejada seja afetada.

O crescimento das funcionalidades implementadas ao longo do desenvolvimento pode ser observado na tabela a seguir:

|  |  |
| --- | --- |
| Fase | Funcionalidades Implementadas |
| Inicial | Adicionar registros básicos com data e texto |
| Intermediária | Visualização de registros, estrutura de banco de dados |
| Avançada | Categorização, busca por palavra, filtragem por data |
| Final | Exclusão de registros, backup de dados, interface gráfica completa |

Esta evolução demonstra a aplicação da metodologia RAD, com incrementos graduais de funcionalidade que resultaram em um produto completo e útil. O uso de SQLite para armazenamento permitiu implementar todas estas funcionalidades de forma eficiente, enquanto o Tkinter proporcionou uma interface gráfica acessível e agradável.

Os testes realizados confirmaram o funcionamento correto de todas as funcionalidades, com tempos de resposta adequados mesmo com múltiplos registros no banco de dados. A aplicação mantém-se estável durante uso prolongado e gerencia corretamente situações como tentativas de busca sem resultados ou visualização com banco de dados vazio.

## 6. Conclusão

O projeto proporcionou aplicação prática dos conceitos de programação, banco de dados e interface gráfica, utilizando RAD. O Diário Digital desenvolvido representa uma ferramenta funcional que permite aos usuários registrar e gerenciar informações pessoais de forma eficiente e organizada.

A implementação do banco de dados SQLite demonstrou-se adequada para o armazenamento estruturado dos registros, permitindo operações de busca, filtragem e exclusão com desempenho satisfatório. A escolha do SQLite também facilitou a implementação da funcionalidade de backup, essencial para a segurança dos dados do usuário.

A interface gráfica desenvolvida com Tkinter proporcionou uma experiência de usuário intuitiva e agradável, com design moderno e controles facilmente identificáveis. A abordagem de diálogos para entrada de dados mostrou-se eficaz, guiando o usuário através do processo de criação e gerenciamento de registros.

A metodologia RAD permitiu o desenvolvimento incremental da aplicação, com adição gradual de funcionalidades que enriqueceram a experiência do usuário. Esta abordagem foi particularmente adequada ao contexto educacional do projeto, permitindo a aplicação prática de diversos conceitos de programação.

Os objetivos educacionais do projeto foram plenamente alcançados, com a aplicação demonstrando o uso de manipulação de arquivos, banco de dados, interface gráfica e estruturas de controle em Python. O resultado final é uma aplicação que não apenas atende aos requisitos acadêmicos, mas também oferece utilidade prática como ferramenta de registro pessoal.

Como possíveis melhorias futuras, poderiam ser implementadas funcionalidades como edição de registros existentes, exportação para diferentes formatos (PDF, texto), sincronização com serviços de nuvem e personalização de temas da interface. Estas expansões demonstrariam ainda mais o potencial do Python como linguagem para desenvolvimento de aplicações desktop completas.

## 7. Referências

ZELEN, P. Python para Desenvolvedores. Novatec, 2020.

* MARTIN, Robert C. Código Limpo: Habilidades Práticas do Agile Software. Alta Books, 2020.
* SUMMERFIELD, Mark. Programming in Python 3: A Complete Introduction to the Python Language. Addison-Wesley, 2019.
* LUTZ, Mark. Learning Python. O’Reilly Media, 2021.
* GADDIS, Tony. Starting Out with Python. Pearson, 2020.
* Pressman, Roger S. Engenharia de Software: uma abordagem profissional. McGraw Hill Brasil, 2016.
* Sommerville, Ian. Engenharia de Software. 10. ed. Pearson, 2019.
* Documentação Oficial do Python. Disponível em: <https://docs.python.org/3/>. Acesso em: maio 2025.
* Documentação Oficial do Tkinter. Disponível em: <https://docs.python.org/3/library/tkinter.html>. Acesso em: maio 2025.
* Documentação Oficial do SQLite. Disponível em: <https://www.sqlite.org/docs.html>. Acesso em: maio 2025.
* FACULDADE MARTHA FALCÃO | WYDEN. Portal Acadêmico.