Universidade Estácio de Sá

curso CIÊNCIA DA COMPUTAÇÂO

UNIDADE NOVA AMERICA

**TRABALHO DE PROJETO EM PYTHON**

**EM DESENVOLVIMENTO RAPIDO DE APLICAÇÔES EM PYTHON**

RIO DE JANEIRO - RJ

MAIO / 2024

202202180769– MIXSODAIME ST NORD

**Trabalho de projeto em Python**

**em Desenvolvimento Rápido de Aplicações em Python**

Trabalho de projeto em python apresentado a Universidade Estácio de Sá, como exigência para avaliação na disciplina desenvolvimento rápido de aplicações em python.

Orientador:

Prof. Ronaldo Candido dos Santos

SUMÁRIO

[1 INTRODUÇÃO 3](#_Toc166009182)

[1.1 DESCRIÇÃO DO PROBLEMA 3](#_Toc166009183)

[1.2 OBJETIVOS 3](#_Toc166009184)

[2 DESENVOLVIMENTO 4](#_Toc166009185)

[2.1 TELA DE LOGIN 4](#_Toc166009186)

[2.2 TELA DE CADASTRO 4](#_Toc166009187)

[3 CONCLUSÃO 5](#_Toc166009188)

[REFERÊNCIAS 6](#_Toc166009189)

# INTRODUÇÃO

(Gostaria de apresentar uma inovação tecnológica que pode transformar a gestão de cadastros de alunos e professores em nossa universidade. Trata-se de um aplicativo desenvolvido com a metodologia RAD (Rápid Application Development) em Python, que oferece uma solução eficiente e moderna em substituição ao uso tradicional de planilhas Excel.)

## DESCRIÇÃO DO PROBLEMA

* Planilhas Excel não oferecem níveis adequados de segurança para proteger informações sensíveis. A ausência de controles de acesso robustos colocava em risco os dados pessoais dos alunos e professores, aumentando a vulnerabilidade a acessos não autorizados

## 

* + O acesso às informações armazenadas em planilhas Excel era limitado e muitas vezes dependia da disponibilidade de arquivos específicos em dispositivos individuais. Isso dificultava o compartilhamento de dados entre departamentos e atrasava a tomada de decisões.

## OBJETIVOS: O aplicativo foi projetado com o intuito de centralizar e otimizar o cadastro de alunos e professores, oferecendo uma interface intuitiva e acessível que facilita a inserção, consulta e atualização de dados. Esta ferramenta visa proporcionar maior segurança das informações, reduzir erros comuns nas planilhas, e melhorar a acessibilidade e usabilidade dos dados.

.MOTIVANDO:

. A criação deste aplicativo foi motivada pela necessidade de solucionar os problemas mencionados acima e proporcionar uma plataforma que: centralize as informa A criação deste aplicativo foi motivada pela necessidade de solucionar os problemas mencionados acima e proporcionar uma plataforma que: Centralize as informações, minimize erros humanos, e facilite o acesso e compartilhamento:

# DESENVOLVIMENTO

(É a parte principal do texto, dividido em seções ou subseções. Contém a descrição pormenorizada do assunto e a fundamentação teórica, podendo conter a metodologia (material e método), os resultados e respectivas discussões (quando previstas atividades experimentais/numéricas no Plano de Trabalho). Devem ser feitas as citações e as notas bibliográficas e/ou explicativas, no texto. Discorrer sobre o tema proposto, fundamentando-se nos textos obtidos de livros e artigos encontrados na literatura, discutindo os principais dados e/ou resultados obtidos, destacando pontos que não estão consolidados na ATUALIDADE).

## TELA DE LOGIN

(para fazer tela de login utilizamos várias pesquisa no livro com ajuda do professor e no site como esses : BACKES, A. Vídeo [ED] Aula 52 - Ordenação - QuickSort. Disponível em: <https://www.youtube.com/watch?v=RZbg5oT5Fgw>. Acesso em: 22 mai. 2021.

BLOODSHED DEV C++: Download do software. Disponível em: <https://sourceforge.net/projects/orwelldevcpp/>. Acesso em: 28 mai. 2021.

Uma imagem contendo Tabela

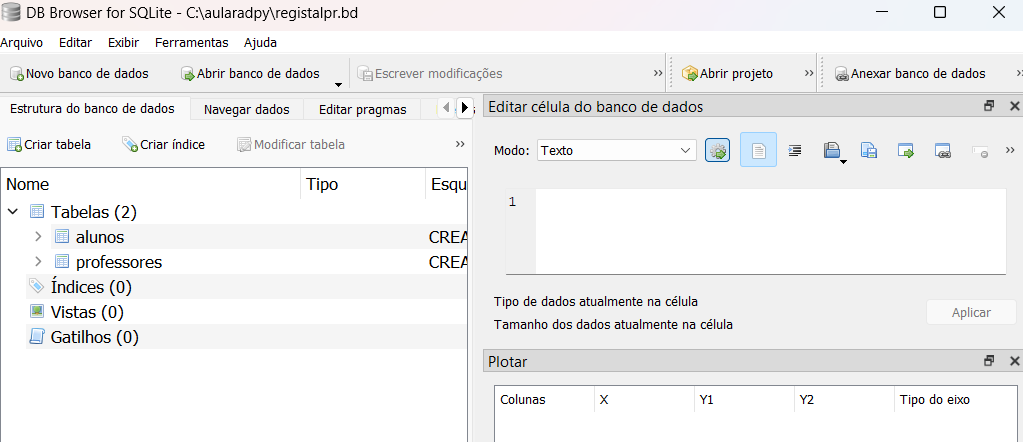
Descrição gerada automaticamente

## TELA DE CADASTRO

## ( A tela de cadastro foi desenvolvido utilizando a biblioteca tkinter em python, que proporciona uma interface gráfica intuitiva e amigável.A tela e dividida em duas seçôes principais: uma para o cadastro alunos e outra para o cadastro de professores. Utilizamos SQLite para armazenas o banco de dados)

## MODELAGE DE DADOS:

## **Coleta de Requisitos**: Envolver stakeholders e usuários finais para identificar necessidades e requisitos do sistema de forma rápida e eficiente. Utilizar workshops, entrevistas e brainstorming.

****

# CONCLUSÃO

(O desenvolvimento do aplicativo seguiu as melhores práticas da metodologia RAD, focando na rápida criação e iteração do software para atender às necessidades da universidade. Abaixo estão descritos os principais estágios do desenvolvimento:

1. **Análise de Requisitos:** Reuniões com stakeholders para identificar problemas e necessidades específicas, documentando os requisitos funcionais e não funcionais do sistema.
2. **Prototipagem:** Desenvolvimento de protótipos iniciais para visualização de funcionalidades e interfaces, com revisões e ajustes baseados em feedback dos stakeholders.
3. **Desenvolvimento Incremental:** Implementação do aplicativo em ciclos iterativos, permitindo a adição gradual de funcionalidades e constante validação com os usuários finais.
4. **Testes e Validação:** Realização de testes exaustivos para garantir funcionalidade, desempenho e segurança do aplicativo.
5. **Implementação e Treinamento:** Implementação do aplicativo nos sistemas da universidade, acompanhada de sessões de treinamento para os usuários finais.
6. **Manutenção e Suporte:** Disponibilização de suporte técnico contínuo e realização de melhorias contínuas com base no feedback dos usuários.

Acreditamos que a implementação deste aplicativo trará grandes benefícios para a universidade, tornando o processo de gerenciamento de dados mais eficiente e seguro. Estamos à disposição para agendar uma reunião e apresentar o aplicativo em detalhes, bem como para esclarecer quaisquer dúvidas que possam surgir.

Agradecemos a atenção e aguardamos a oportunidade de contribuir para a modernização dos processos administrativos da universidade.

Atenciosamente,

.)

# REFERÊNCIAS

(Elemento obrigatório constituído por uma lista ordenada dos documentos efetivamente citados no texto. Não devem ser referenciadas fontes bibliográficas que não foram citadas no texto. Indicar todos os artigos e livros consultados, e utilizados para o desenvolvimento deste trabalho. Exemplos:)

ASCENCIO, A. F. G.; ARAÚJO, G. S. de. Estrutura de Dados: Algoritmos, Análise da Complexidade e implementações em Java e C/C++. São Paulo: Pearson Prentice Hall, 2010. Capítulo 2: Algoritmos de ordenação e busca. Páginas 21-102.

BACKES, A. Vídeo [ED] Aula 52 - Ordenação - QuickSort. Disponível em: <https://www.youtube.com/watch?v=RZbg5oT5Fgw>. Acesso em: 22 mai. 2021.

BLOODSHED DEV C++: Download do software. Disponível em: <https://sourceforge.net/projects/orwelldevcpp/>. Acesso em: 28 mai. 2021.

CORMEN, T. Desmistificando algoritmos. Rio de Janeiro: Elsevier, 2014. Capítulo 3: Algoritmos para ordenar e buscar. Páginas 20-49.

KOFFMAN, E. B.; WOLFGANG, P. A. T. Abstração, Estruturas de Dados e Projeto Usando C++. Rio de Janeiro: LTC, 2008. Capítulo 10: Ordenação.

NORMAS ABNT. Normas para elaboração de trabalhos acadêmicos. Disponível em: <https://www.normasabnt.org/>. Acesso em: 28 mai. 2021.