Universidade Estácio de Sá

curso Desenvolvimento rápido de aplicações em python

UNIDADE NOVA AMÉRICA

**TRABALHO DE CADASTRO DE ALUNO**

**EM DESENVOLVIMENTO E APLICAÇÕES RÁPIDAS EM PYTHON**

Rio de Janeiro - RJ

06 / 2025

202308290444 – Allan Reis

202103060201 – Anderson Glay Gonçalves Marques

202402796143 – Alex Machado Sá

202403235293 – Fábio Souza

**Trabalho de cadastro de aluno**

**em Desenvolvimento e Aplicação rápida em python**

Trabalho de cadastro de alunos apresentado a Universidade Estáciode Sá, como exigência para avaliação na disciplina desenvolvimento de aplicação rápida em python

Orientador:

Prof. Ronaldo Candido dos Santos

SUMÁRIO

[1 INTRODUÇÃO 3](#_Toc84406832)

[1.1 DESCRIÇÃO DO PROBLEMA 3](#_Toc84406833)

[1.2 OBJETIVOS 3](#_Toc84406834)

[2 DESENVOLVIMENTO 4](#_Toc84406835)

[2.1 XXXXXXXXXXXXX 4](#_Toc84406836)

[3 CONCLUSÃO 5](#_Toc84406837)

[REFERÊNCIAS 6](#_Toc84406838)

# INTRODUÇÃO

O objetivo dessa pesquisa é armazenar informações cadastrais dos alunos de uma rede privada de ensino com o objetivo de facilitar, automatizar e atualizar os dados armazenados entre outros que está descrito no item 1.2 em objetivos.

## DESCRIÇÃO DO PROBLEMA

Atualmente, esse processo é feito manualmente em papéis ou planilhas, o que pode levar a erros, perdas de informações e dificuldade na busca por dados. É essencial ter um sistema que permita o cadastro, consulta, atualização e exclusão de informações dos alunos de forma fácil e intuitiva.

## OBJETIVOS

(Definir os resultados a obter na realização do trabalho.)

* **Gerenciamento Eficiente de Dados:** Criar um sistema que facilite o gerenciamento eficiente das informações dos alunos, como dados pessoais, histórico acadêmico e contatos de emergência.
* **Facilitar o Acesso à Informação:** Desenvolver uma plataforma que permita que os professores e administradores acessem rapidamente informações relevantes sobre alunos, auxiliando na tomada de decições pedagógicas.
* **Integrar Dados Acadêmicos:** Permitir a integração com outras bases de dados, como notas, frequências e atividades extracurriculares, proporcionando uma visão mais completa do desempenho do aluno.
* **Melhorar a Comunicação:** Estabelecer um canal de comunicação efeicaz entre a escola, os alunos e os responsáveis, promovendo o engajamento e a colaboração em relação ao processo eduacativo.
* **Atender as normas Regulatórias:** Garantir que o cadastro siga as normas e regulamentos de proteção de dados pessoais, como a LGPD (Lei Geral de Proteção de Dados), respeitando a privacidade dos alunos.
* **Automaziar Processos:** Automatizar processos como matrículas, atualizações de dados e emissão de documentos, reduzindo erros e economizando tempo.
* **Facilitar Relatórios e Análises:** Permitir a geração de relatórios analíticos sobre a população estudantil, facilitando o planejamento estratégico e a identificação de áreas que precisam de melhorias.
* **Promoção a inclusão:** Garantir que o sistema atenda a todas as necessidades dos alunos, incluindo aqueles com deficiências, promovendo a inclusão e a equidade na educação.

# DESENVOLVIMENTO

Este projeto tem como objetivo o desenvolvimento de um sistema de **cadastro de alunos** para uma instituição de ensino, utilizando a metodologia **Rapid Application Development (RAD)** com a linguagem de programação **Python**. A escolha pela abordagem RAD se deve à necessidade de entrega rápida, iterações constantes com o cliente e flexibilidade para mudanças ao longo do processo.

## Fases aplicadas no projeto

## *Planejamento dos requisitos*: Coleta rápida de necessidades com os usuários.

## *Design do Sistema*: Criação de wireframes e definição da estrutura de dados.

## *Construção:* Desenvolvimento incremental com python e SQLite.

## *Entrega:* Testes com os usuários e ajustes finais

## 3.0 – Tecnologias Utilizadas

## Visual Studio Code para desenvolver o código

## Python 3.10+

## SQLite(Para um banco de dados leve e embutido)

## 4.0 – Funcionalidades Implementadas

## Cadastro de novos alunos.

## Listagem de alunos com filtros por nome, série ou situação.

## Edição e exclusão de registros.

## Validação de dados de entrada.

## Interface responsiva para uso em diferentes dispositivos.

## 5 – Conclusão O uso da metodologia RAD permitiu o denvolvimento rápido e iterativo da aplicação, com validação constante dos requitos pelos usuários finais. A escolha do Python e do SQLite se mostrou eficaz para entregar uma solução funcional, escalável e de fácil manutenção.

## 5 – Conclusão O uso da metodologia RAD permitiu o denvolvimento rápido e iterativo da aplicação, com validação constante dos requitos pelos usuários finais. A escolha do Python e do SQLite se mostrou eficaz para entregar uma solução funcional, escalável e de fácil manutenção.

# REFERÊNCIAS

(Elemento obrigatório constituído por uma lista ordenada dosdocumentos efetivamente citados no texto. Não devem serreferenciadas fontes bibliográficas que não foram citadas no texto. Indicar todos osartigos, livros, sites 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. 2024.

BLOODSHED DEV C++:Download do software. Disponível em: <https://sourceforge.net/projects/orwelldevcpp/>. Acesso em: 28 mai. 2024.

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. 2024.