



UNIVERSIDADE FEDERAL DO CEARÁ

Tecnologias do Projeto

Versão: 2.0

Equipe: Erick Gabriel Ferreira Gaspar - 536261

Ítalo Kauã Vitor Fernandes - 537595

José Vinicius Evangelista Dias de Souza - 537071

Pedro Henrique Santos Moreira - 536925

Professor: Camilo Camilo Almendra

Disciplina: Projeto Integrado de Software 1

SUMÁRIO

Introdução.....	3
1. Linguagens de programação.....	3
2. Frameworks.....	3
3. Tecnologias.....	4
4. Controle e versionamento:.....	4
5. Bibliotecas.....	4

Introdução

Este documento tem como objetivo definir as tecnologias usadas no projeto, bem como os critérios de seleção e avaliação das mesmas. A descrição detalha as tecnologias escolhidas para cada componente do sistema, explicando as vantagens e desvantagens de cada uma.

1. Linguagens de programação

- [Typescript](#)/Javascript: Usadas na construção do *front-end* juntamente com o electron e o node.js .
 - A ideia é construir uma interface moderna, responsiva e intuitiva, logo, a escolha dessa linguagem juntamente com html e CSS se torna uma alternativa superior, levando em conta que é a abstração mais usada no mercado atualmente para fazer as interfaces.
- [Python](#) : Usada no *back-end* e na construção das *APIs* do sistema e acesso ao banco de dados.
 - Para o back-end python é uma escolha sólida, justamente por ser usada amplamente com análise de dados, ela tem bibliotecas fundamentais para toda a tratativa e análise de dados do cliente.
- [SQL](#): Usada na manipulação do banco de dados.

2. Frameworks

- [Electron](#) : Utilizado para renderizar uma página web como uma aplicação desktop multiplataforma.
 - A ideia é projetar uma interface de ponta usando as melhores ferramentas para web, justamente por ter um processo de abstração e implementação mais ágeis, e usar numa janela desktop.
 - Facilita o porte da aplicação em diferentes plataformas.
- [Vite- React](#) : Como o electron é projetado para um *SPA*, Esse framework é uma escolha ideal, além de ter uma vasta biblioteca de componentes feita pela comunidade.
- [FastApi](#): Utilizado no *back-end* para acelerar e facilitar a abstração e implementação do servidor web da *API*.

3. Tecnologias

- [LLama2](#) : LLM para ser usada na parte educacional do projeto e será usada no *back-end*.
 - Escolhida por ser uma tecnologia em alta, recente, open-source e gratuita e desempenho semelhante ao dos concorrentes.
- [Docker](#) : Usado para containerizar o *back-end* e individualizar os processos.
 - A principal motivação do uso é poder garantir mais segurança na implementação das APIs no *back-end* justamente por emular as configurações do ambiente de desenvolvimento da aplicação independente do computador e também por permitir a comunicação entre diferentes containers.
- [Node.JS](#) : Interpretador de JavaScript local, usado com electron e JS.
- [Postgres](#): Banco de dados usado na aplicação.
 - Sua escolha se deve ao fato de ser gratuito e open-source e ser uma das principais opções do mercado, têm bom desempenho e ótimas ferramentas de gerenciamento que facilitam a implementação e manutenção.

4. Controle e versionamento:

- [Git](#) : Ferramenta para gerenciar o repositório local da equipe.
- [Github](#) : Plataforma de hospedagem do repositório online.

5. Bibliotecas

- [react -icons](#): lib de ícones da comunidade para o *front-end*.
- [react-router-dom](#): lib para mapear e organizar as páginas no *front-end*.
- [react-magic-motion](#): lib utilizada para animar de forma rápida elementos e componentes no *front-end*.
- [psycopg2](#): lib de python para manipular o banco de dados usado na *API* no *back-end*.
- [sqlalchemy](#): lib disponibiliza uma engine para gerar *queries* para manipular o banco de dados no *back-end*.