

Guia de Desenvolvimento do Projeto: Gerador de Documentos com IA

Para: Patrick (Desenvolvedor)

De: Neuton (Cliente)

Assunto: Resumo e Escopo Atualizado do Projeto de Automação de Documentos

Olá, Patrick.

Este documento formaliza o escopo e os requisitos do nosso projeto, conforme discutido em nossa reunião e nas conversas subsequentes. O objetivo é criar uma aplicação web que auxilie usuários a gerar documentos complexos (DFD e ETP) de forma automatizada, utilizando Inteligência Artificial.

1. Resumo do Projeto

O projeto consiste em desenvolver uma plataforma web onde um usuário pode fazer login, preencher um formulário com informações específicas sobre uma demanda de contratação e, com base nessas respostas, o sistema utilizará a API da OpenAI (GPT) para gerar o texto descritivo e estruturado de dois documentos principais:

- **DFD** (Documento de Formalização de Demanda)
- **ETP** (Estudo Técnico Preliminar)

A aplicação deve guiar o usuário na inserção dos dados necessários e, em seguida, orquestrar a chamada à IA para redigir as seções mais elaboradas dos documentos, combinando as respostas do usuário com o texto gerado pela IA para produzir um documento final pronto para uso.

2. Escopo Atualizado e Funcionalidades Principais

A visão inicial de usar n8n para preencher templates foi superada. O novo escopo é centrado em uma aplicação customizada com as seguintes funcionalidades:

Sistema de Autenticação de Usuários:

- O usuário precisa criar uma conta e fazer login para acessar o sistema.
- Isso é fundamental para salvar o progresso, armazenar documentos gerados e garantir a privacidade dos dados.

Interface de Coleta de Dados (Formulários Dinâmicos):

- Após o login, o usuário selecionará qual documento deseja criar (DFD ou ETP).
- O sistema apresentará um formulário com todas as perguntas necessárias para gerar o documento escolhido.

- **Fonte de dados para os formulários:** Os arquivos `Documento 1 - DFD.pdf` e `Documento 2 - ETP.pdf` são a base para todos os campos que devem ser solicitados ao usuário.

Núcleo de Geração com IA (O "Cérebro" do Sistema):

Este é o coração do projeto. O back-end da aplicação será responsável por:

- Coletar todas as respostas do formulário preenchido pelo usuário.
- Construir um prompt detalhado para a API da OpenAI. Este prompt deve conter as respostas do usuário e instruções claras para a IA sobre como estruturar o texto (similar ao exemplo fornecido por Neuton: "I - DESCRIÇÃO DA NECESSIDADE...").
- Enviar a requisição para a API da OpenAI usando a chave fornecida: `sk-proj-o01o1...`
- Receber e tratar a resposta de texto gerada pela IA.

Montagem e Geração do Documento Final:

- O sistema deve pegar o texto gerado pela IA e combiná-lo com os dados estruturados fornecidos pelo usuário (ex: nomes, números, códigos, datas) em um layout final.
- O documento final deve ser gerado em um formato prático, como `.docx` ou `.pdf`, para que o usuário possa baixar e editar, se necessário.

Dashboard do Usuário (Área Logada):

- Dentro da sua área logada, o usuário deve ter um histórico dos documentos que já gerou, com a possibilidade de visualizá-los, baixá-los novamente ou excluí-los.

3. Fluxo de Trabalho do Usuário

1. **Acesso:** Usuário acessa o site e faz login ou se cadastra.
2. **Seleção:** No painel principal, escolhe "Gerar Novo Documento" e seleciona "DFD" ou "ETP".
3. **Preenchimento:** O usuário é levado a um formulário passo a passo, onde responde a todas as perguntas listadas nos documentos de referência.
4. **Geração:** Ao finalizar o preenchimento, o usuário clica em "Gerar Documento". O sistema envia os dados para o back-end, que consulta a IA e monta o arquivo.
5. **Download:** Após alguns instantes, o documento final é disponibilizado para download. Ele também fica salvo no histórico do usuário.

4. Detalhes Técnicos e Recomendações

Front-end:

- Deve ser uma aplicação web moderna e responsiva. Frameworks como React, Vue ou Angular são recomendados para criar uma experiência de usuário fluida, especialmente para os formulários.

Back-end:

- Será necessário um servidor para gerenciar a autenticação, salvar os dados dos formulários e, crucialmente, fazer as chamadas para a API da OpenAI de forma segura.
- **Atenção:** A chave da API da OpenAI ((sk-...)) NUNCA deve ser exposta no código do front-end. Todas as chamadas para a IA devem ser intermediadas pelo seu back-end.
- Tecnologias como Node.js (com Express), Python (com Django/Flask) ou outras são adequadas.

Engenharia de Prompt:

- A qualidade do documento final dependerá diretamente da qualidade do prompt enviado à IA. Será necessário experimentar para criar um "mega-prompt" que instrua a IA a usar as respostas do usuário para escrever cada seção do documento no tom e formato corretos. Os exemplos de documentos ((DFD Piscina.docx), (Estudo Técnico.docx)) podem servir de inspiração para o estilo de escrita esperado.

Geração de Arquivos:

- Pesquisar e utilizar bibliotecas para a geração de arquivos (.docx) ou (.pdf) no lado do servidor (ex: docx ou pdf-lib para Node.js).

5. Próximos Passos Sugeridos

1. **Arquitetura:** Definir a stack de tecnologias (front-end, back-end, banco de dados).
2. **Modelagem de Dados:** Desenhar o esquema do banco de dados para armazenar usuários e seus documentos.
3. **Módulo de Autenticação:** Implementar o sistema de login/cadastro como primeira funcionalidade.
4. **Interface dos Formulários:** Construir os formulários para DFD e ETP com base nos PDFs fornecidos.
5. **Integração com IA:** Criar o serviço no back-end que monta o prompt, chama a API da OpenAI e processa a resposta.
6. **Gerador de Documentos:** Implementar a funcionalidade que insere os dados e o texto da IA em um template de documento e o disponibiliza para download.