**Projeto de Programação**

UFCD 10790 - U26

|  |
| --- |
| Data: 2024/12/13 |
| Nome do Formador: João David |
| ID do Formando: 7407712  Nome do Formando: Elisa Coutinho |

**1. Introdução**

**Âmbito do Projeto**

**1.1.1. Descrição**

O projeto consiste em uma análise de dados para extrair insights e criar visualizações gráficas com base em uma base de dados contendo postings de empregos no LinkedIn dos Estados Unidos referentes a um único dia.

**1.1.2. Objetivo**

O objetivo principal é realizar uma análise exploratória dos dados, identificando padrões e gerando visualizações claras que apresentem os insights obtidos. Este projeto também visa aprimorar habilidades práticas em análise de dados e composição de relatórios visuais, usando ferramentas e frameworks amplamente utilizados na área.

**1.1.3. Público-Alvo**

• **Recrutadores** e **líderes de equipe** interessados em explorar dados de empregos nos Estados Unidos para obter insights rápidos sobre o mercado de trabalho.

• **Comitês de avaliação ou recrutadores técnicos**, para demonstrar as habilidades do desenvolvedor em análise e apresentação de dados.

**1.1.4. Limitações**

• **Escopo temporal limitado:** A análise é baseada em dados de um único dia de postings de empregos no LinkedIn, o que restringe a abrangência temporal das conclusões.

• **Região específica:** Os insights são centrados no mercado de trabalho dos Estados Unidos, sem considerar variações globais.

• Recursos individuais: O projeto é desenvolvido individualmente, o que limita o tempo e a possibilidade de incluir recursos mais complexos.

**Levantamento de Requisitos**

**1.1.5. Funcionais**

• Coletar e processar dados de postings de empregos no LinkedIn.

• Realizar análises exploratórias para identificar padrões e insights relevantes.

• Criar visualizações gráficas utilizando Matplotlib para apresentar os resultados.

• Documentar o processo de análise e as observações obtidas.

**1.1.6. Não Funcionais**

• O projeto será desenvolvido em **Python** através do **Jupyter Notebook**, utilizando principalmente os frameworks **pandas** e **Matplotlib**.

• Garantir que o código seja bem documentado e legível.

• As visualizações devem ser criadas com foco na clareza e impacto visual, facilitando a interpretação dos dados.

• Utilizar Trello para organizar as tarefas e prazos.

**2. Desenvolvimento**

**2.1 Arquitetura**

O projeto foi estruturado para permitir tanto a análise sistemática dos dados quanto a apresentação visual e interativa dos resultados. A arquitetura inclui:

• Base de Dados: Um arquivo CSV contendo postings de empregos no LinkedIn dos Estados Unidos em um único dia.

• Jupyter Notebook: Ambiente principal para execução de código e análise dos dados.

• Website: Desenvolvido com Flask para apresentar o projeto e as visualizações de forma interativa.

• Estrutura de Código: Dividida em:

• Scripts para análise de dados e criação de gráficos no Jupyter Notebook.

• Scripts do website em Flask, organizados com templates HTML e diretórios de arquivos estáticos.

• Visualizações: Integradas ao site para permitir que os usuários explorem insights de forma dinâmica.

**2.2 Tecnologias Utilizadas**

Python: Linguagem principal usada tanto para análise de dados quanto para desenvolvimento web.

• Flask: Framework leve para criar o website do projeto.

• pandas: Para manipulação e análise de dados.

• Matplotlib: Para geração dos gráficos usados no site e relatório.

• HTML/CSS: Para o design e apresentação do site.

• Jupyter Notebook: Plataforma usada para o desenvolvimento inicial da análise de dados.

• Visual Studio Code: Ambiente de desenvolvimento para os scripts do Flask e o website.

• Trello: Gerenciamento de tarefas e prazos.

**2.3 Implementação**

**2.3.1 Análise de Dados:**

• O arquivo CSV foi carregado para um DataFrame pandas, permitindo a limpeza, organização e análise exploratória dos dados.

• Foram criados gráficos detalhados para destacar os principais insights sobre níveis de experiência, tipos de contrato e localizações.

**2.3.2 Desenvolvimento do Website:**

• Utilizando Flask, foi criado um website com:

- Home Page: Apresenta o objetivo e a motivação do projeto.

- Seções Interativas: Cada análise possui uma seção com gráficos e textos explicativos.

- Download dos Dados: Um botão permite que os visitantes baixem a base de dados original para explorar.

- Design Responsivo: Utilizando CSS para uma boa apresentação em diferentes dispositivos.

• Arquivos estáticos (CSS, imagens e gráficos gerados) foram organizados em pastas específicas para facilitar o desenvolvimento.

**2.3 Integração:**

• Os gráficos gerados no Jupyter Notebook foram exportados e incorporados ao website.

• Rotas em Flask foram configuradas para exibir as visualizações e oferecer uma navegação fluida.

**3. Desempenho do Projeto**

**3.1 Executar o Projeto**

Para executar o projeto, siga as etapas:

1. Instale as dependências listadas no requirements.txt:

pip install -r requirements.txt

1. Execute o servidor Flask:

python app.py

1. Acesse o website pelo navegador no endereço <http://127.0.0.1:5000>.
2. Para analisar o código de dados, abra o arquivo no Jupyter Notebook e execute as células.

**3.2 Testes Realizados**

Os testes incluíram:

• **Validação do Website**:

• Testes de navegação e exibição de gráficos para garantir que as rotas Flask funcionam corretamente.

• Verificação do download da base de dados pela interface do site.

• **Validação do Código**:

• Testes com diferentes tamanhos de dados para garantir a eficiência dos scripts.

• Conferência das visualizações geradas no Matplotlib para consistência com os dados.

**4. Conclusão**

**4.1 Dificuldades**

As principais dificuldades enfrentadas incluem:

• A integração entre os gráficos gerados no Jupyter Notebook e o website.

• Ajustar o design do website para ser responsivo e visualmente agradável.

**4.2 Reflexão**

Este projeto não só aprimorou habilidades em análise de dados e visualização, mas também expandiu conhecimentos em desenvolvimento web com Flask. A criação do website adicionou uma camada prática ao projeto, permitindo a apresentação de resultados de forma interativa.

**4.3 Trabalhos Futuros**

Como próximos passos, pretende-se:

• Analisar outros datasets, incorporando séries temporais para capturar tendências mais amplas.

• Explorar frameworks de visualização mais interativos, como Plotly ou Dash.

• Melhorar o website para incluir gráficos dinâmicos gerados diretamente a partir do Flask.