

CURSO: Talento Tech

DISCIPLINA: Fundamentos de engenharia de software

PROFESSOR FORMADOR: Dr. Mauricio Zadra Pacheco

ALUNO: Maria Julia de Souza das Neves

ATIVIDADE 5: Projeto Individual com Git

Conversor de Moedas - Documentação do Desenvolvimento

Objetivo do Projeto

O objetivo deste projeto é desenvolver um aplicativo simples de conversão de moedas que permite ao usuário converter entre diferentes moedas, como o **Dólar (USD)**, o **Real (BRL)** e o **Euro (EUR)**. O sistema utiliza uma **API** para obter taxas de câmbio atualizadas e permite ao usuário selecionar as moedas de origem e destino, além de fornecer o valor convertido.

Funcionalidades Implementadas

- **Seleção de Moeda:** O usuário pode escolher a moeda de origem e a moeda de destino (USD, BRL, EUR).
- **Entrada de Valor:** O usuário insere um valor que deseja converter.
- **Conversão em Tempo Real:** O valor inserido é convertido utilizando as taxas de câmbio obtidas através de uma API externa.
- **Interface Simples e Responsiva:** A interface foi projetada para ser simples e fácil de usar, com um layout que se adapta a diferentes tamanhos de tela.

Desafios Enfrentados

1. Integração com a API de Câmbio

A maior dificuldade foi integrar a API externa para buscar as taxas de câmbio atualizadas. Inicialmente, encontrei dificuldades com a autenticação e a forma de consumir os dados da API.

Solução: Após pesquisar sobre diferentes APIs de câmbio, optei por utilizar a API gratuita do [exchangerate-api](#), que fornecia dados JSON com as taxas de câmbio. Realizei a integração com o JavaScript utilizando a função `fetch()`.

2. Manipulação de Erros

Outro desafio foi garantir que o aplicativo tratasse adequadamente possíveis falhas, como erros na API, entradas inválidas ou valores não numéricos.

Solução: Implementei um sistema de verificação de erros para checar entradas inválidas do usuário, como valores não numéricos ou vazios. Além disso, adicionei um tratamento de erros para o caso de falha ao acessar a API de câmbio.

3. Design Responsivo

Como o projeto precisava ser acessível em diferentes dispositivos (desktop e dispositivos móveis), a implementação de uma interface responsiva foi um desafio.

Solução: Utilizei **CSS Flexbox** para garantir que os elementos se ajustassem dinamicamente de acordo com o tamanho da tela, proporcionando uma experiência de usuário otimizada.

Como o Git Ajudou no Processo

Git foi fundamental no gerenciamento do projeto e controle de versão. Ele permitiu que eu organizasse o desenvolvimento e mantivesse o código facilmente acessível. As funcionalidades foram divididas em **branches**, o que facilitou o desenvolvimento paralelo e o controle de alterações.

1. Uso de Branches

Para organizar o desenvolvimento, criei branches específicas para cada tarefa:

- **ui-development:** para o desenvolvimento da interface do usuário.
- **api-integration:** para integrar a API de câmbio ao projeto.
- **error-handling:** para adicionar verificações e mensagens de erro.

2. Commits e Merges

Durante o desenvolvimento, realizei commits frequentes para registrar as mudanças feitas no código e facilitar o controle de versões:

- **Exemplo de Commit:** "Adicionada lógica de conversão de moedas com API externa."

Após concluir o trabalho em cada branch, fiz o merge para a branch principal (main) para consolidar as funcionalidades desenvolvidas.

3. Versionamento e Rastreabilidade

Com os commits frequentes, foi possível rastrear facilmente qualquer erro ou alteração indesejada. Caso algo tivesse dado errado, poderia reverter para versões anteriores rapidamente utilizando o Git.

Conclusão

O uso do Git ajudou a manter o desenvolvimento organizado, especialmente em um projeto que envolvia integração com uma API externa. A utilização de branches foi essencial para o desenvolvimento paralelo e o gerenciamento de diferentes funcionalidades. O projeto foi concluído com sucesso, e a implementação da API de câmbio permitiu que a conversão de moedas fosse feita em tempo real.