

Gestão financeira pessoal

Responsáveis: Gustavo Larsen e Luis Felipe Mondini

[Link Figma](#)

Objetivo

Auxiliar o usuário a manter o controle de suas finanças pessoais, possibilitando-o a registrar seus créditos/débitos, visualizar o histórico e evolução da balança, através de números e gráficos pré-montados.

Funcionalidades Principais

Registro de Despesas e Receitas: Os usuários poderão inserir suas transações financeiras, especificando se são despesas ou receitas.

Categorização: As transações serão categorizadas para facilitar a análise e o acompanhamento.

Visualização de Fluxo de Caixa: A aplicação exibirá gráficos e relatórios para mostrar o fluxo de caixa ao longo do tempo.

Definição de Metas Financeiras: Os usuários poderão estabelecer metas de economia ou de gastos e acompanhar seu progresso.

Tecnologias Utilizadas

Frontend:

HTML, CSS e JavaScript para a interface do usuário.

React para a criação de componentes.

Backend:

PHP para o servidor.

Banco de dados Apache MySQL para armazenamento de dados.

Versionamento de código:

GitHub.

Arquitetura:

Monolítica.

Passos do Desenvolvimento

1. Planejamento e Análise

Definir os requisitos detalhados da aplicação.

Criar protótipos da interface do usuário usando o Figma.

2. Configuração do Ambiente de Desenvolvimento

Instalar e configurar PHP e MySQL no ambiente de desenvolvimento.

Configurar um projeto usando React para o frontend.

Configurar um projeto usando PHP para o backend.

3. Desenvolvimento do Backend

Desenvolver funções para autenticação de usuários.

Criar modelos de dados para usuários, transações financeiras e categorias.

Implementar as rotas RESTful para manipulação de usuários e transações.

4. Desenvolvimento do Frontend

Criar componentes React para formulários de registro e login, entrada de transações e visualização de relatórios.

Implementar a integração com as APIs do backend para autenticação e manipulação de dados.

Estilizar a interface do usuário usando CSS.

5. Integração e Testes

Integrar o frontend com o backend e garantir a comunicação adequada entre as partes.

Realizar testes de unidade e integração para garantir o bom funcionamento de cada componente e funcionalidade.

6. Implementação de CI/CD

Implementação de CI/CD usando o GitHub Actions