
Relatório de Atividades - Trabalho Prático

1. Identificação do Grupo

- **Nome do projeto:** SCFF - Sistema de Controle Financeiro Familiar
- **Integrantes do grupo:**
 - João Martins
 - João Elias
 - Maria Senna
- **Curso:** Sistemas de Informação
- **Semestre:** 5º Semestre
- **Disciplina:** Banco de Dados II
- **Professor responsável:** Rebeca Barros
- **Data de entrega:** 21/04/2025

2. Introdução

Este projeto tem como objetivo o desenvolvimento de um sistema completo que possibilite o controle financeiro de forma familiar, permitindo aos usuários gerenciar receitas, despesas, investimentos, dívidas, patrimônio e orçamentos. O sistema também fornece relatórios para auxiliar na tomada de decisões financeiras com base em dados consolidados.

Inicialmente, o projeto contemplava entidades relacionadas a imposto de renda, grupos familiares e dependentes. Contudo, ao longo do desenvolvimento e conforme os requisitos da atividade foram sendo analisados, optamos por reformular a modelagem, priorizando tabelas mais alinhadas ao escopo da disciplina e ao CRUD básico com recursos avançados em SQL.

3. Modelagem do Banco de Dados

O banco de dados foi modelado para contemplar as seguintes tabelas principais:

- usuario
- meta
- investimento
- patrimonio
- dividas

- movimentacao
- orçamento_mensal
- auditoria

Relações e chaves estrangeiras:

- A tabela movimentacao está relacionada ao usuario.
- meta, investimento, patrimonio, dividas e orcamento também estão relacionadas ao usuario.

Diagrama ER:

Incluído no anexo do projeto (gerado via ferramenta)

4. Implementação do Banco de Dados

O SGBD utilizado foi o **PostgreSQL**. Foi desenvolvido um script completo contendo:

a) DDL

- Criação do banco de dados e tabelas com suas respectivas chaves primárias e estrangeiras.

b) DML

- Inserção de ao menos 2 registros em cada tabela.

c) Aspectos avançados:

- **Trigger de auditoria:** Criada para registrar alterações em uma das tabelas principais (movimentacao). As modificações são registradas na tabela auditoria.
 - **Function:** A função contar_linhas_tabelas soma todas as linhas, itens das tabelas e as retorna para ter uma melhor visão da quantidade de itens em cada tabela
 - **View:** A view relatorio_financeiro_mensal consolida receitas, despesas e saldo para cada usuário, permitindo a visualização mensal das finanças.
 - **View:** A view resumo_financeiro_outros consolida progresso médio das metas, total patrimônio e total investido final e seus respectivos proventos para cada usuário, permitindo a visualização total dessas finanças.
 - **View:** A view gasto_mensal_por_categoria consolida o gasto mensal por categoria baseado na tabela de movimentação para cada usuário, permitindo a visualização mensal dessas finanças.
 - **View:** A view comparativo_orcamento_gasto consolida a comparação entre o orçamento_mensal planejado e o gasto mensal baseado na tabela de movimentação para cada usuário, permitindo a visualização mensal dessas finanças.
-

5. Desenvolvimento da Aplicação

A aplicação foi desenvolvida em **Python**, utilizando os seguintes recursos:

- **FastAPI** para criação da API;
- **Pydantic** para validação dos dados;
- **SQLAlchemy** para conexão com o banco, mas as queries foram escritas manualmente em SQL puro, conforme exigido;

A API realiza todas as operações básicas de CRUD para os recursos principais do sistema, recebendo dados em JSON.

6. Demonstração

Durante a apresentação, as operações CRUD serão demonstradas da seguinte forma:

- **CREATE (Inserção):** Inclusão de uma nova movimentação na aplicação e visualização direta no PgAdmin.
 - **READ (Consulta):** Consulta a relatórios ou registros utilizando as views ou SELECTs.
 - **UPDATE (Atualização):** Atualização de uma meta ou orçamento e conferência do valor atualizado no banco.
 - **DELETE (Remoção):** Exclusão de uma despesa e verificação da ausência do registro no banco.
-

7. Conclusão

Este trabalho proporcionou uma oportunidade prática de aplicar conhecimentos de modelagem, SQL avançado, e integração entre sistemas e banco de dados. Os maiores desafios incluíram a adaptação da modelagem inicial para atender aos critérios da atividade e a construção de triggers e views com foco gerencial.

Futuramente, o sistema pode ser expandido com funcionalidades extras como envio de notificações, gráficos interativos, ou integração com bancos e cartões.

8. Anexos

- Script SQL completo: scff_script_postgresql_usuario_relacionado.sql
 - Diagrama ER
-