

Descrição Geral do Sistema

O objetivo deste sistema é auxiliar na gestão financeira pessoal, permitindo ao usuário cadastrar suas contas, saldos, despesas e receitas.

As tecnologias para o desenvolvimento do sistema são: HTML, CSS, JavaScript, Java, banco de dados relacional PostgreSQL e controle de versionamento (Git e Github).

Funções do Sistema

O sistema deve permitir que o usuário possa inserir, editar e apagar registros.

Regras de negócio

- Requisitos Funcionais

RF001 – O sistema deve possuir uma tela de cadastro de usuário, exigindo nome, e-mail e senha.

RF002 – O sistema deve possuir uma tela de Login.

RF003 – O sistema deve possuir uma aba lateral de navegação.

RF004 – O sistema deve utilizar banco de dados relacional para o armazenamento de registros.

RF005 – O sistema deve permitir que o usuário insira dados de receitas, sendo eles: valor, data de recebimento, data de recebimento esperado, descrição, conta e tipo de receita.

RF006 – A tela de receitas deve possuir as funções de cadastrar, editar, remover.

RF007 – O sistema deve permitir que o usuário insira dados de despesas, sendo eles: valor, data de pagamento, data de pagamento esperado, tipo de despesa e conta.

RF008 – A tela de despesas deve possuir as funções cadastrar, editar e remover.

RF009 – O sistema deve permitir que o usuário insira dados de conta, sendo eles: saldo, tipo de conta e instituição financeira.

RF010 – A tela de conta deve possuir as funções de cadastrar, editar e remover.

- Requisitos Funcionais adiados

RF001 – A tela de conta deve possuir a função de transferir saldo entre contas.

RF002 – A tela de receitas deve permitir o usuário utilizar filtros por período e tipo de receita.

RF003 – A tela de despesas deve permitir o usuário utilizar filtros por período e tipo de receita.

RF004 – A tela de conta deverá listar saldo total

RF005 – O sistema deve realizar o logout do usuário ao clicar no botão de 'sair'

- Requisitos Não-Funcionais

RNF001 – O sistema deve ser desenvolvido no formato de uma REST API.

RNF002 – O sistema só pode ser acessado por um usuário cadastrado.

RNF003 – O sistema deve ter alta disponibilidade, acima de 99% do tempo.

RNF004 – O sistema deve ser desenvolvido em JAVA.

RNF005 – O sistema deve utilizar bando de dados relacional para armazenamento.

RNF006 – O sistema deve se comunicar com o banco de dados.

RNF007 – Deve ser utilizado controle de versionamento para o desenvolvimento do sistema.

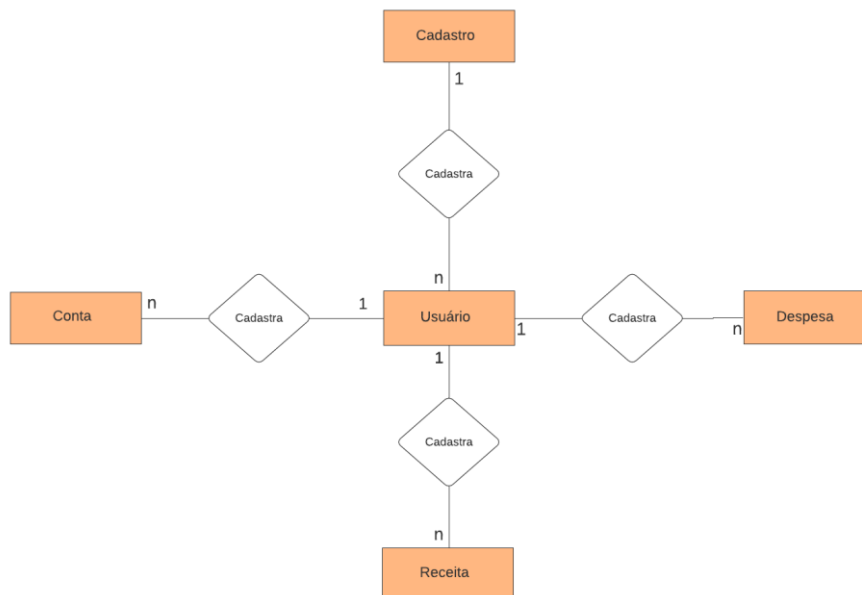
Arquitetura do Sistema

O sistema foi desenvolvido utilizando o padrão MVC de arquitetura, pois desta forma podemos usufruir de seus benefícios como: organização, eficiência e segurança.

Além disso, a programação foi realizada no paradigma orientada à objetos.

Diagrama de Entidade Relacionamento

Diagrama de Entidade e Relacionamento



- Um usuário pode ser cadastrado somente uma vez, e o cadastro pode cadastrar vários usuários
- Um usuário pode cadastrar várias receitas e uma receita só pode ser cadastrada por um usuário
- Um usuário pode cadastrar várias despesas e uma despesa só pode ser cadastrada por um usuário
- Um usuário pode cadastrar várias contas e uma conta só pode ser cadastrada por um usuário

Diagrama de Navegação

Diagrama de navegação

Luiz Maciel | January 15, 2022

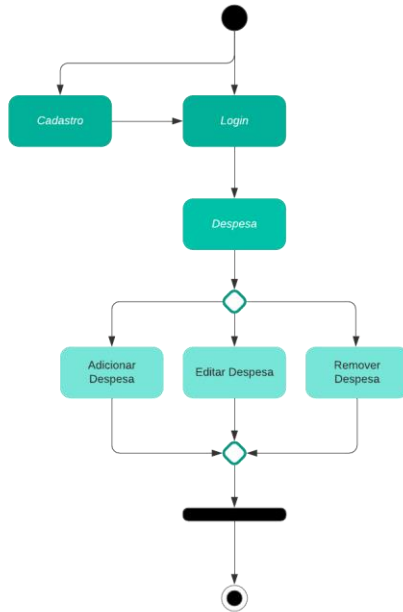


Diagrama de navegação

Luiz Maciel | January 15, 2022

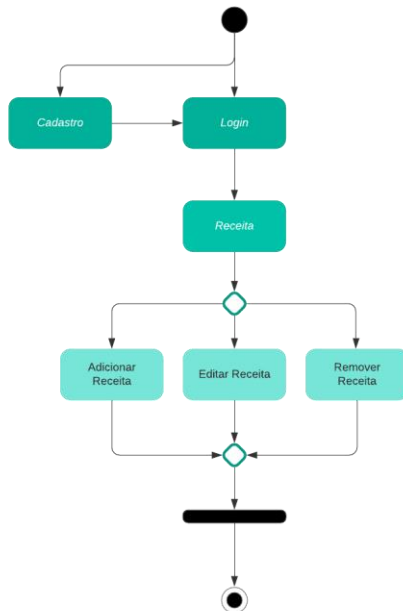
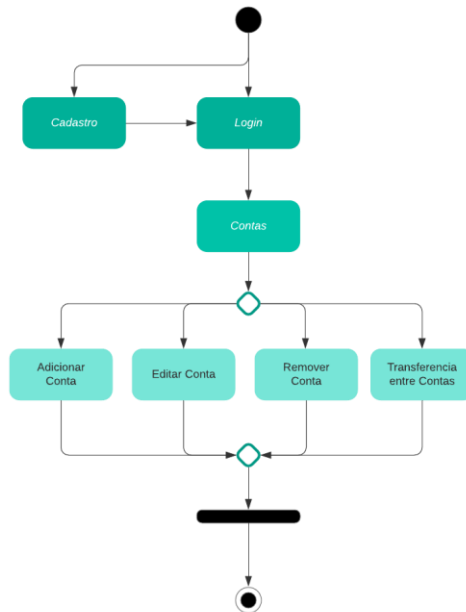


Diagrama de navegação

Luiz Maciel | January 15, 2022



Observação

Infelizmente algumas partes do código poderão estar presentes em outros projetos, como pode ser observado nos insights abaixo:



