REVISÕES

|  |  |  |  |
| --- | --- | --- | --- |
| Versão | Data | Autor | Notas |
| 0.1 | 06/06/2023 | Leonardo Graças | Base geral do projeto |
| 0.2 | 07/06/2023 | Leonardo Graças | Inserção das diagramações |

Banco digital em Python

# Entrevista (fictícia):

* O cliente precisa de uma aplicação web de gerenciamento de banco.
* Dentre as operações, existirão: cadastro de usuário, remoção de usuário, edição de dados do usuário.
* Além dessas, existirão operações específicas para cada usuário: saque, depósito, transferência, pegar empréstimos e pagar dívidas.
* Haverá um botão para passar o mês. Esse botão irá:
  + Somar a renda de cada usuário ao seu saldo.
  + Fazer render o saldo corrente que sobrar e distribuir entre o banco e os usuários.
  + Aumentar os juros das dívidas dos usuários.
  + Cobrar a taxa de anuidade do banco
* Se o usuário ficar muito tempo com o saldo negativo, sua conta é automaticamente deletada.
* Se o banco tiver o saldo negativo, ele irá a falência.

# Doc. Casos de Uso:

Diagrama

Descrição gerada automaticamente

Figura -Visão geral do projeto

## Inicialização e atualização:

* Atualiza as informações do banco dos clientes cadastrados na View com os dados atuais.
* Retorna 0 ou null se os dados forem vazios.
* **Exceção:** *Banco de dados não encontrado*
  + Para o caso de haver algum problema com o endereço do banco de dados
* **Exceção:** *Banco de dados inacessível*
  + Para o caso de haver algum problema com o banco de dados em si
* **Estado final:** todas as informações requisitadas ao banco de dados devem ser formatadas e exibidas na tela.

Diagrama

Descrição gerada automaticamente

## Passar mês:

* Clientes negativados recebem faltas. Ao atingir o limite de faltas, o banco toma ações judiciais e a conta é excluída.
* Soma o valor da renda mensal ao saldo do cliente
* Cobra os juros dos débitos pendentes dos clientes
* Rende o saldo corrente e divide o lucro entre o banco e os clientes
* Cobra a taxa de anuidade dos clientes
* **Estado final:** Todos os valores, exceto a renda fixa, devem sofrer algum tipo de alteração dependendo de que tipo de operação os modificou.

Diagrama

Descrição gerada automaticamente

## Gerar id:

* Gera um id único
* **Estado final:** Devolve o id gerado.

Diagrama

Descrição gerada automaticamente

## Cadastro de cliente:

* Recebe o nome e renda do novo usuário
* Gera um id único para o usuário
* Instancia um cliente
* Cadastra o cliente
* Atualiza a lista de clientes
* **Exceção:** *Problema no cadastro do cliente*
  + Para o caso de as informações fornecidas pelo usuário não serem do tipo correto
* **Exceção:** *Nome inválido*
  + Para o caso de haver alguma irregularidade no nome do usuário, como algo impróprio ou algum número.
* **Estado final:** A lista de clientes no banco de dados deve ser atualizada com o novo cliente

Diagrama

Descrição gerada automaticamente

## Selecionar cliente:

* Mostrar todos os dados do cliente
* Mostrar todas as operações possíveis para o cliente
* **Exceção:** *Cliente inexistente*
  + Para o caso de o cliente referenciado não estar mais cadastrado no banco e dados
* **Estado final:** A aba de operações retorna todos os dados do cliente referenciado

Uma imagem contendo Diagrama

Descrição gerada automaticamente

## Deletar cliente:

* Pré-requisito: usuário existe no sistema
* Deleta o cliente referenciado
* Atualiza a lista de clientes
* Atualiza os dados do banco
* Verifica se o cliente não está negativado
* **Exceção:** *Débitos pendentes*
* **Estado final:** a lista de clientes no banco de dados deve ser atualizada e o cliente referenciado deve ser removido

Diagrama

Descrição gerada automaticamente

## Saque:

* Retira a quantia informada do saldo do cliente caso o valor esteja disponível
* **Exceção:** *Saldo insuficiente*
  + Para o caso de o valor informado ser maior do que o saldo
* **Exceção:** *Valor inválido*
  + Para o caso de o valor informado ser negativo ou não ser um número real.
* **Estado final:** *Saldo* deve ser igual ao *Saldo inicial* com o valor informado decrescido.

**Diagrama

Descrição gerada automaticamente**

## Depósito:

* Deposita a quantia informada no saldo do cliente.
* **Exceção:** *Valor inválido*
  + Para o caso de o valor informado ser negativo ou não ser um número real.
* **Estado final:** *Saldo* deve ser igual ao *Saldo inicial* com o valor informado acrescentado

**Diagrama

Descrição gerada automaticamente**

## Empréstimo:

* Caso o valor informado esteja dentro do limite disponível para o usuário, deposita a quantia.
* Cria uma dívida com o banco baseada no valor sacado.
* **Exceção:** *Limite atingido*
  + Para o caso de o valor solicitado ser superior ao limite disponível
* **Estado final:** O valor informado deve ser depositado no saldo do cliente referenciado, que deve também criar um débito com o banco.

**Diagrama

Descrição gerada automaticamente**

## Transferência:

* Caso a quantia informada esteja disponível e a conta exista ou não seja a mesma referenciada, realiza a operação.
* Retira a quantia do saldo da conta referenciada e deposita no saldo da conta de destino.
* **Exceção:** *Saldo insuficiente*
  + Para o caso de o valor informado ser maior do que o saldo disponível
* **Exceção:** *Usuário não encontrado*
  + Para o caso de o usuário de destino não existir no sistema.

**Diagrama

Descrição gerada automaticamente**

## Pagar débitos:

* Tenta quitar o maior valor possível baseado no saldo disponível na conta.
* **Estado final:** O saldo da conta e o débito irão se subtrair.

**Diagrama

Descrição gerada automaticamente**