

## PROJETO INTERDISCIPLINAR

# FINUP

## Banco Digital

### Alunos:

RGM	Nome
31077978	Raphael Matoso Lima Pinto
30948363	Renato Lopes Ghirelli
31861270	Rogério Silva Cordeiro
30965420	José Lucas de Jesus Souza
34016881	Paulo Roberto Lima Chaves
31685382	André Januário Junior

São Paulo

2023

UNIVERSIDADE CRUZEIRO DO SUL

PROJETO INTERDISCIPLINAR

# FINUP

## Banco Digital

Trabalho apresentado como parte do requisito para aprovação na Disciplina de Projeto Interdisciplinar do curso de Tecnologia em Análise e Desenvolvimento de Sistemas da Universidade Cruzeiro do Sul.

**Orientadores:** Prof. Agnaldo e Prof. Fabio

São Paulo

2023

## Sumário

1. Apresentação:.....	4
1.1 Justificativa e Motivação .....	4
1.2 Dados do Sistema.....	4
2 Requisitos .....	5
3 Regras de Negócio .....	7
4 Diagramas de Caso de Uso .....	8
5 Diagrama de Classe .....	12
6 Interface .....	13
7 Consideração finais .....	18
8 Bibliografia .....	18

# **1. APRESENTAÇÃO:**

## **1.1 Justificativa e Motivação**

Este projeto tem como finalidade compor os conceitos estudados nas disciplinas de Engenharia de Software e Análise e Projeto de Sistemas, visando colocar em prática todo o conhecimento adquirido durante as aulas e com os exercícios propostos pelos professores das respectivas disciplinas.

Destacamos também a oportunidade de trabalhar em grupo e desenvolvemos as nossas soft skills que são importantes para a carreira de um profissional da área de tecnologia.

Este projeto oferece os desafios à altura de um curso de Análise e Desenvolvimento de Sistemas, pois reúne, de uma só vez, habilidades adquiridas em diversas áreas da computação, em especial àquelas voltadas ao Análise e Projetos de Sistemas e Engenharia de Software, além de incentivar a pesquisa.

## **1.2 Dados do Sistema.**

O sistema foi criado para atender as necessidades básicas de um sistema bancário, tem como suas principais funções a realização de saques, transferências do tipo PIX, DOC, TED, comprovantes de transferências, extratos bancários entre outros.

## 2 REQUISITOS

2.1 Requisitos Funcionais	
Requisito	Comportamento
<b>Login</b>	Permitir que os clientes acessem o sistema utilizando o número de CPF e senha;
<b>Realização de saques.</b>	Permitir que os clientes do banco façam saques em caixas eletrônicos ou estabelecimentos conveniados;
<b>Transferências via DOC/TED</b>	Permitir que os clientes possam realizar transferências de valores maiores para outras contas de bancos diferentes, utilizando os sistemas de DOC ou TED.
<b>Realização de PIX</b>	Permitir que os clientes possam realizar transferências de valores de forma instantânea utilizando O sistema PIX.
<b>Geração de comprovantes</b>	Gerar comprovantes das operações realizadas pelos clientes, incluindo saques, transferências, DOCs, TEDs e PIX.
<b>Manutenção de um repositório de clientes</b>	Manter um cadastro atualizado dos clientes que utilizam O sistema, incluindo informações como nome, CPF, endereço e telefone.
<b>Manutenção de um repositório de operações bancárias</b>	Manter um registro de todas as operações realizadas pelos clientes, incluindo saques, transferências, DOCs, TEDs e PIX.
<b>Oferta de histórico de todas as transações aos clientes</b>	Permitir que os clientes possam consultar o histórico de todas as operações bancárias realizadas por eles através do sistema.
<b>Oferta de informes dos valores aplicados em conta aos clientes</b>	Permitir que os clientes possam consultar informações sobre os valores aplicados em conta, incluindo rendimentos e taxas
<b>Capacidade de integração com outros sistemas bancários para operações</b>	Permitir que o sistema possa se comunicar com outros sistemas bancários para realizar operações como transferências interbancárias
<b>Capacidade de interação instantânea para o envio e</b>	Oferecer aos clientes canais de atendimento instantâneos para tirar dúvidas e solucionar problemas

recebimento de mensagens  
através de canais de  
atendimento

## 2.2 Requisitos não funcionais

Requisito	Função
<b>Segurança</b>	O sistema deve garantir um alto nível de segurança, para evitar fraudes e proteger as informações dos clientes.
<b>Escalabilidade</b>	O sistema deve ser escalável, capaz de suportar um grande número de clientes e operações, sem comprometer a performance.
<b>Disponibilidade</b>	O sistema deve estar sempre disponível para os clientes, sem interrupções ou quedas
<b>Confiabilidade</b>	O sistema deve ser confiável, garantindo a integridade e a consistência das operações realizadas pelos clientes.
<b>Usabilidade</b>	O sistema deve ser fácil de usar, com uma interface amigável e intuitiva, que permita aos clientes realizar operações de forma simples e rápida.
<b>Performance</b>	O sistema deve ter um alto desempenho, processando as operações bancárias de forma rápida e eficiente.
<b>Integração</b>	O sistema deve ser capaz de se integrar com outros sistemas bancários, permitindo a realização de operações interbancárias de forma transparente para os clientes.
<b>Manutenção</b>	O sistema deve ser fácil de manter, permitindo a realização de manutenções e atualizações sem afetar a disponibilidade ou a performance.

### 3 REGRAS DE NEGÓCIO

#### Regras do negócio

##### Abertura de conta

- Para que o cliente possa abrir uma conta é necessário primeiro fazer um cadastro com seus documentos pessoais (Documento de identificação com foto, CPF, Comprovante de residência atualizado até 90 dias) e uma foto de seu rosto para a verificação em relação ao seu documento. Assim que feito esse cadastro, o cliente terá de esperar até dois dias úteis para a verificação dos dados fornecidos, essa etapa tem como objetivo evitar fraudes;
- Caso a abertura da conta seja aprovada o cliente receberá um cartão de débito do banco, para que possa fazer saques em caixas 24h e compras.

##### Transferências

- Para que possa ser feita uma transferência é necessário que o cliente tenha esse saldo positivo em conta;
- Quando feita uma transferência TED, o dinheiro cai na conta no mesmo dia se realizada até as 17h;
- Quando feita uma transferência DOC, o dinheiro cai no dia seguinte, porém podendo demorar até dois dias se feita após as 22h. E essa tem o limite de valor, máximo de R\$ 4.999,99;
- Quando feita uma transferência PIX, o cliente poderá limitar o valor de transferência total por dia, ou por horário. O valor é transferido instantaneamente;
- Sempre que feita uma transferência o aplicativo disponibilizara o comprovante em formato PDF, com a opção de download.

##### Saques

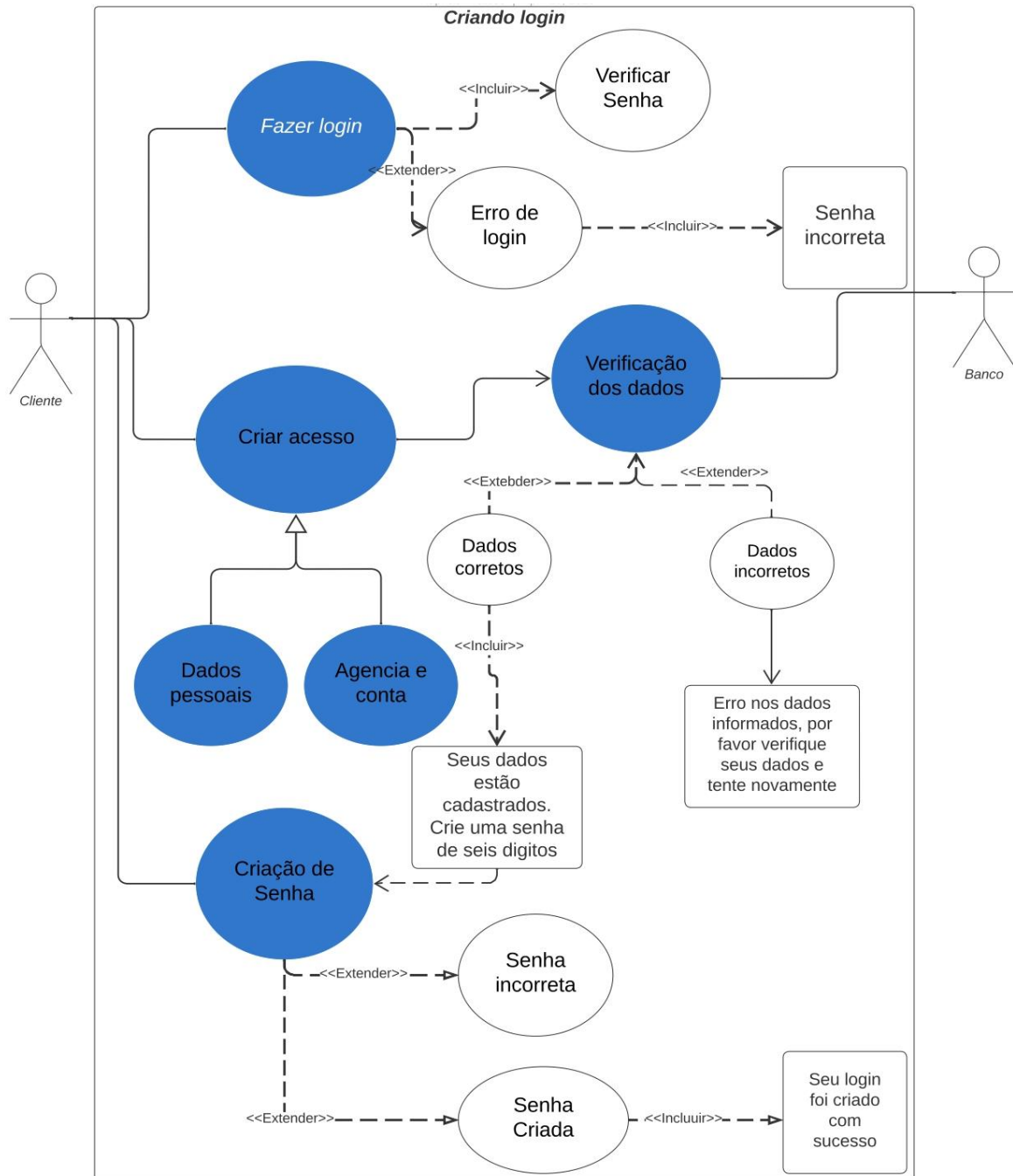
- Se digitada errada por 3 vezes o cartão será bloqueado;
- Para que possa ser feito um saque é necessário que o cliente tenha o saldo positivo disponível em sua conta em relação ao valor que quer sacar;
- Só poderá realizar os saques em caixas 24h ou em bancos conveniados.

##### Suporte

- É necessário que haja um FAQ com as dúvidas mais frequentes dos clientes.

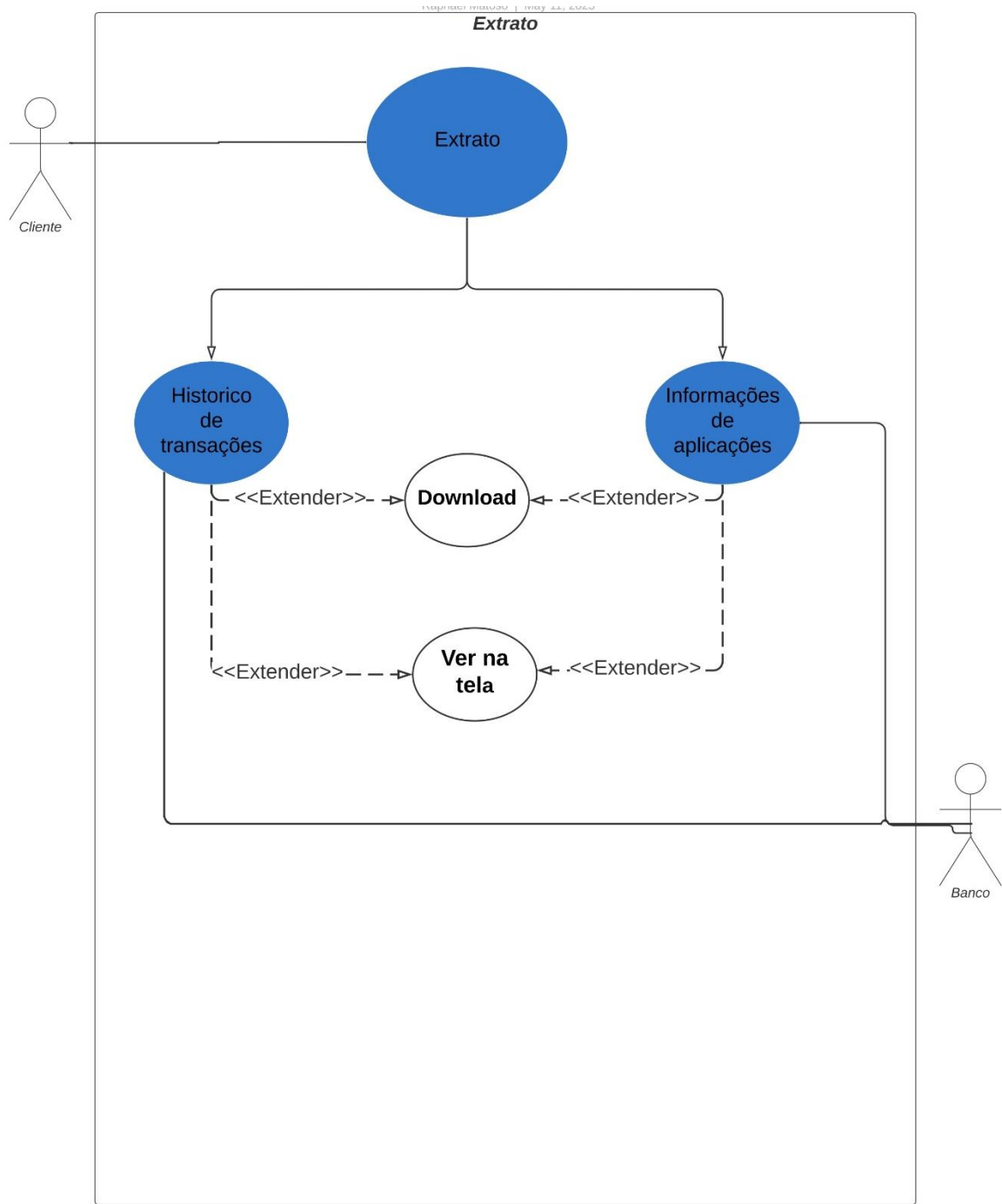
## 4 DIAGRAMAS DE CASO DE USO

### 4.1 Login e criação de acesso

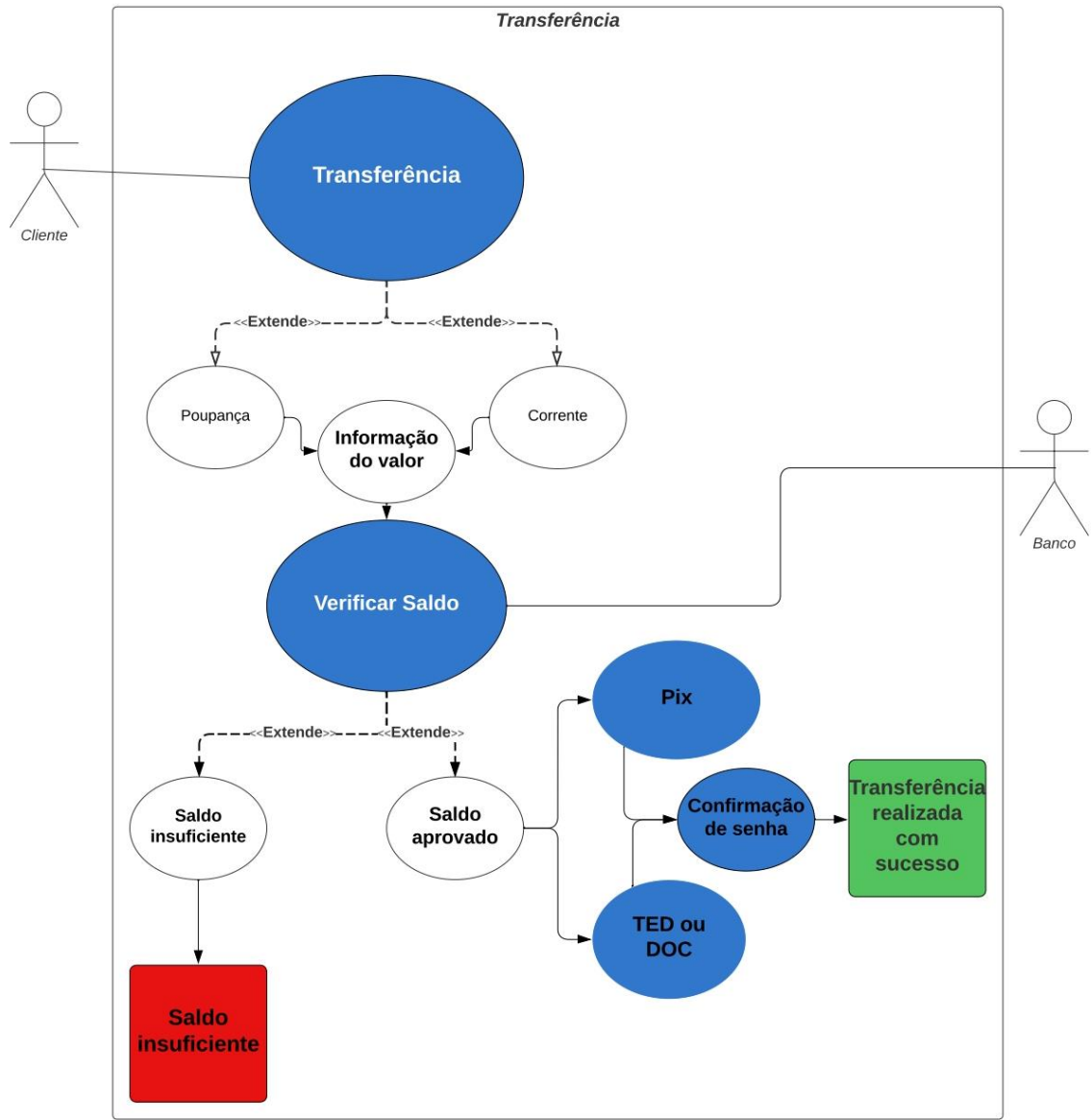




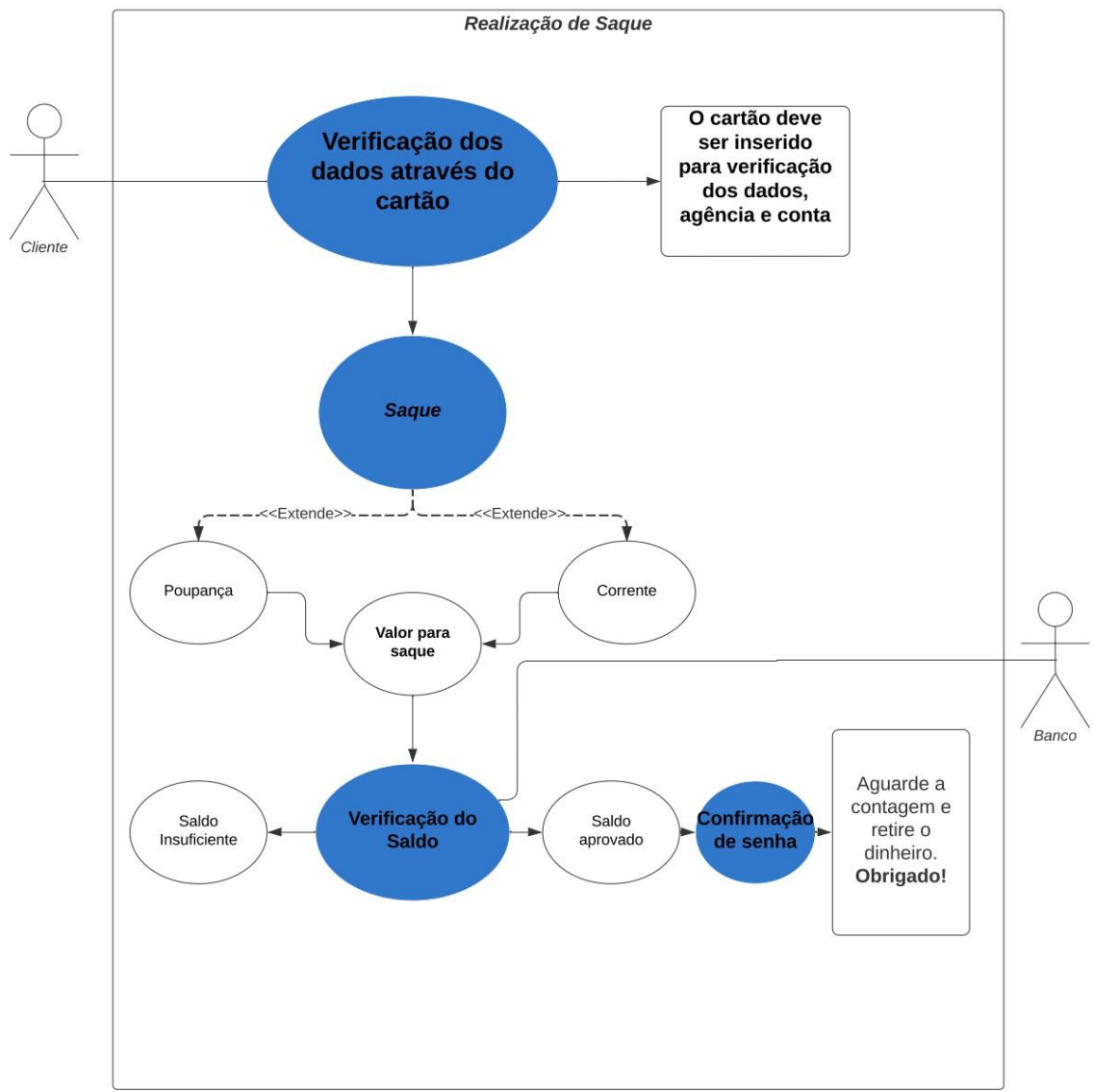
4.2 Extrato



4.3 Transferências

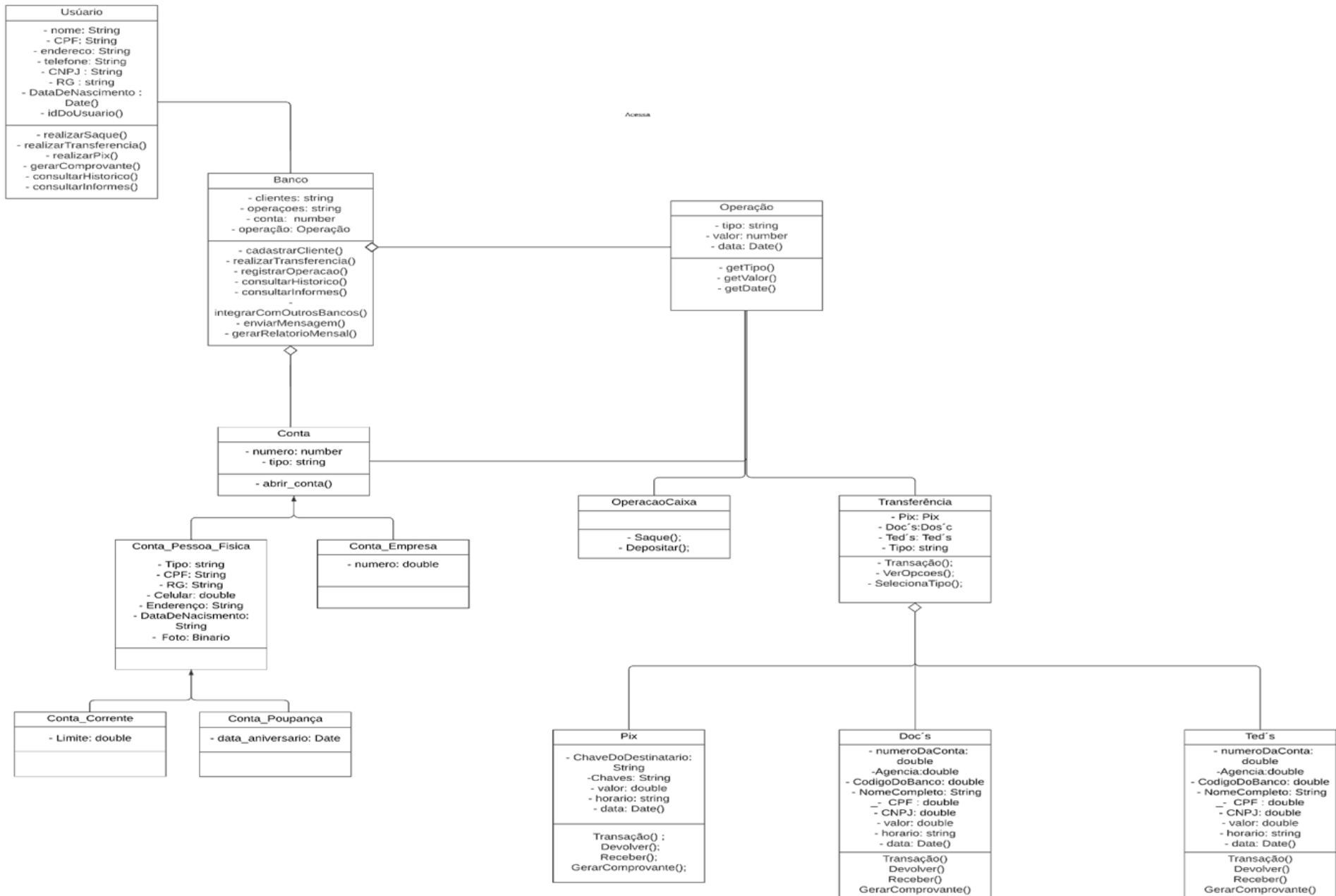


4.4 Saque



## 5 DIAGRAMA DE CLASSES

Abriu Conta



## 6 INTERFACE

### 6.1 Tela Inicial

```
-----Seja Muito Bem Vindo-----
**** Seleccione uma operação que deseja realizar ****
-----
| Opção 1 - Criar Conta |
| Opção 2 - Depositar  |
| Opção 3 - Sacar       |
| Opção 4 - Transferir  |
| Opção 5 - Listar      |
| Opção 6 - Ajuda       |
| Opção 7 - Sair        |
```

```
public static void operacoes(){
    System.out.println(x:"-----");
    System.out.println(x:"-----Seja Muito Bem Vindo-----");
    System.out.println(x:"-----");
    System.out.println(x:"**** Seleccione uma operação que deseja realizar ****");
    System.out.println(x:"-----");
    System.out.println(x:"| Opção 1 - Criar Conta |");
    System.out.println(x:"| Opção 2 - Depositar  |");
    System.out.println(x:"| Opção 3 - Sacar       |");
    System.out.println(x:"| Opção 4 - Transferir  |");
    System.out.println(x:"| Opção 5 - Listar      |");
    System.out.println(x:"| Opção 6 - Ajuda       |");
    System.out.println(x:"| Opção 7 - Sair        |");

    int op = entrada.nextInt();

    switch(op){
        case 1:
            criarConta();
            break;
        case 2:
            depositar();
            break;
        case 3:
            sacar();
            break;
        case 4:
            transferir();
            break;
        case 5:
            listar();
            break;
        case 6:
            help();
            break;
        case 7:
            System.out.println(x:"Obrigado por usar nossos serviços!");
            System.exit(status:0);
        default:
            System.out.println(x:"Opção Inválida!");
            operacoes();
            break;
    }
}
```

## 6.2 Criar Conta

```
Nome:
Renato

CPF:
123456789

E-mail:
renato@renato.com
Sua conta foi criada com sucesso!
```

```
public static void criarConta() {

    System.out.println(x: "Nome: ");
    String nome = entrada.next();

    System.out.println(x: "\nCPF: ");
    String cpf = entrada.next();

    System.out.println(x: "\nE-mail: ");
    String email = entrada.next();

    Cliente c = new Cliente(nome, cpf, email);
    Conta conta = new Conta(c);

    contasBancarias.add(conta);

    System.out.println(x: "Sua conta foi criada com sucesso!");

    operacoes();
}
```

## 6.3 Depositar

```
Número da conta:
123456

Qual valor quer depositar?
250
Deposito realizado com sucesso!
```

```

public static void depositar(){
    System.out.println(x:"\nNúmero da conta: ");
    int numConta = entrada.nextInt();

    Conta conta = encontrarConta(numConta);

    if(conta != null){
        System.out.println(x:"\nQual valor quer depositar?");
        double valorDeposito = entrada.nextDouble();
        conta.depositar(valorDeposito);
    } else {
        System.out.println(x:"A conta não foi encontrada");
    }
    operacoes();
}

```

#### 6.4 Sacar

```

Número da conta:
123456

Qual valor quer sacar?
250
Saque realizado com sucesso!

```

```

public static void sacar(){
    System.out.println(x:"\nNúmero da conta: ");
    int numConta = entrada.nextInt();

    Conta conta = encontrarConta(numConta);

    if(conta != null){
        System.out.println(x:"\nQual valor quer sacar?");
        double valorSaque = entrada.nextDouble();
        conta.sacar(valorSaque);
    } else {
        System.out.println(x:"A conta não foi encontrada");
    }
    operacoes();
}

```

## 6.5 Transferir

```
Número da conta do Remetente:  
123456  
Para qual conta deseja transferir?  
654321  
Qual o valor da transferência?  
500  
Transferência realizada com sucesso!
```

```
public static void transferir(){  
    System.out.println(x:"\nNúmero da conta do Remetente: ");  
    int numConta = entrada.nextInt();  
  
    Conta contaRemetente = encontrarConta(numConta);  
  
    if (contaRemetente != null){  
        System.out.println(x:"Para qual conta deseja transferir? ");  
        int numContaDestinatario = entrada.nextInt();  
  
        Conta contaDestinatario = encontrarConta(numContaDestinatario);  
  
        if (contaDestinatario != null){  
            System.out.println(x:"Qual o valor da transferência? ");  
            double valor = entrada.nextDouble();  
            contaRemetente.transferir(contaDestinatario, valor);  
        }  
    }  
  
    operacoes();  
}
```

## 6.6 Repositório de Clientes

```
Número da Conta: 1  
Nome: Renato  
CPF: 123456789  
E-mail: renato@renato.com  
Saldo: R$ 1.000,00
```

```
Número da Conta: 2  
Nome: Rogerio  
CPF: 321654987  
E-mail: rogerio@rogerio  
Saldo: R$ 0,00
```



```

public static void listar(){

    if (contasBancarias.size() > 0 ){
        for (Conta conta: contasBancarias){
            System.out.println(conta);
        }
    } else {
        System.out.println(x:"Não há contas cadastradas!");
    }

    operacoes();
}

```

## 6.7 Menu de ajuda

```

1 - Cadastrar nova conta
2 - Realizar depósitos
3 - Realizar saques
4 - Realizar transferências
5 - Lista de clientes
6 - retornar ao menu principal

```

```

public static void help(){
    System.out.println(x:"1 - Cadastrar nova conta");
    System.out.println(x:"2 - Realizar depósitos");
    System.out.println(x:"3 - Realizar saques");
    System.out.println(x:"4 - Realizar transferências");
    System.out.println(x:"5 - Lista de clientes");
    System.out.println(x:"6 - retornar ao menu principal");

    int opt = entrada.nextInt();
}

```

## **7 CONSIDERAÇÃO FINAIS**

É importante destacar que as transações bancárias envolvem dados sensíveis e informações confidenciais dos usuários, o que exige a implementação de medidas de segurança robustas para garantir a integridade dos dados e prevenir fraudes. Além disso, a facilidade de uso do sistema também são fundamentais para garantir uma boa experiência do usuário, o que pode influenciar na fidelização do cliente e na reputação da instituição bancária. Dessa forma, é essencial que o sistema seja projetado e desenvolvido levando em consideração os requisitos de segurança e usabilidade, para garantir a satisfação e confiança dos usuários. Dito isso conseguimos entender que em qualquer desenvolvimento de um sistema existem diversos aspectos que precisam ser levados em consideração antes mesmo de iniciar a parte de desenvolvimento do código em si. A maior parte do sucesso de um sistema está em como ele foi planejado e estruturado.

Dentro das maiores dificuldades encontradas em desenvolver esse projeto está em fazer a implementação de todos os requisitos que são necessários para que um software da área financeira precisa. Tal fato evidenciou que mesmo em pequenos projetos existe a necessidade de diversos profissionais.

A falta de expertise sobre aplicações financeiras nos desafiou em fazer as regras de negócios e os requisitos o que implicou diretamente na estruturação do projeto.

## **8 BIBLIOGRAFIA**

Materiais de apoio das disciplinas de Engenharia de Software, Análise e Projeto de Sistemas e Programação Orientada a Objetos.