







**Ana Beatriz Chaves Cardoso da Silva**

**Anabell Roanes Rodrigues de Amorim**

**Nicole Freitas Flaminio**

**Banco Malvader**

Documento apresentado ao Curso de graduação de Bacharelado em Engenharia de Software da Universidade Católica de Brasília, como requisito parcial para obtenção da aprovação na disciplina de Laboratório de Banco de Dados.

Orientador: Prof. William Roberto Malvezzi

**Brasília**

**2024**

**RESUMO**

Referências: Ana Beatriz Cardoso da Silva; Roanes, Anabell Rodrigues de Amorim; Freitas, Nicole Flaminio. Banco Malvader, 2024. 50p. Bacharelado em Engenharia de Software – UCB – Universidade Católica de Brasília, Taguatinga – DF, 2024.

O presente trabalho consiste no desenvolvimento do Banco Malvader, uma aplicação executada em na linguagem Java utilizando Swing para a interface gráfica e um banco de dados MySQL. O objetivo desse sistema é gerir operações, através de conceitos de Programação Orientada a Objetos e persistência de dados.

O sistema permite a administração de contas correntes e poupanças, incluindo funcionalidades como consulta de dados, alterações cadastrais, apresentação de relatórios financeiros, entre outras funções de cadastro e consulta. Com uma interface organizada e intuitiva, a aplicação se divide em duas frentes com opções para funcionários e clientes.

Além disso, esse projeto conta com uma arquitetura modular, estabelecida por pacotes para modelos, controladores, acesso ao banco de dados e interface gráfica. Dessa maneira, o programa disponibiliza um resultado eficiente para a gestão simulada de um banco, assegurando a consistência, integridade e segurança das informações.

Palavras-chave: Banco de Dados, Java, Swing, Gerenciamento, Arquitetura Modular.

**ABSTRACT**

Reference: Chaves, Ana Beatriz Cardoso da Silva; Roanes, Anabell Rodrigues de Amorim; Freitas, Nicole Flaminio. Banco Malvader, 2024. 50 pages. Bachelor's Degree in Software Engineering – UCB – Catholic University of Brasília, Taguatinga – DF, 2024.

This work involves the development of Banco Malvader, an application implemented in the Java programming language using Swing for the graphical interface and a MySQL database. The system aims to manage operations through the application of Object-Oriented Programming concepts and data persistence.  
 The system enables the management of checking and savings accounts, including functionalities such as data queries, profile updates, financial reporting, and other registration and consultation features. With an organized and intuitive interface, the application is divided into two modules, offering options for both employees and customers.  
 Furthermore, the project incorporates a modular architecture structured into packages for models, controllers, database access, and the graphical user interface. This approach ensures an efficient solution for the simulated management of a banking system, guaranteeing consistency, integrity, and security of information.

Keywords: Database, Java, Swing, Management, Modular Architecture.

**Lista de Figuras**

Figura 1: Diagrama de atividades

Figura 2: Diagrama de Caso de Uso Geral

Figura 3: Diagrama de Caso de Uso Específico - Cliente

Figura 4: Diagrama de Caso de Uso Específico - Funcionário / Administrador

Figura 5: Diagrama de sequência - Consultar Saldo

Figura 6: Diagrama de sequência - Fazer Depósito

Figura 7: Diagrama de sequência - Fazer Saque

Figura 8: Diagrama de sequência - Exibir Extrato

Figura 9: Diagrama de sequência - Consultar Limite

Figura 10: Diagrama de classe - Caso de uso cliente

Figura 11: Diagrama de classe - Saque

Figura 12: Diagrama de classe - Saldo

Figura 13: Diagrama de classe - Extrato

Figura 14: Diagrama de classe - Depósito

Figura 15: Diagrama de classe - Funcionário

Figura 16: Diagrama de classe - Sair

Figura 17: Diagrama de classe - Consultar Limite

Figura 18: Diagrama de classe - Abertura de Conta

Figura 19: Diagrama de classe - Alterar dados

Figura 20: Diagrama de classe - Consultar dados

Figura 21: Diagrama de classe - Encerramento de conta

Figura 22: Diagrama de classe - Gerar relatórios

Figura 23: Diagrama de classe - Cadastrar funcionário

Figura 24: Diagrama de classe - Depósito

Figura 25: Diagrama de classe - Gerar relatórios

Figura 26: Diagrama de classe - Extrato

Figura 27: Diagrama de classe - Saque

Figura 28: Diagrama de classe - Saldo

Figura 29: Diagrama de classe - Consultar Limite

Figura 30: Diagrama de classe - Consultar Dados

Figura 31: Diagrama de classe - Sair

Figura 32: Diagrama de classe - Encerrar Conta

Figura 33: Diagrama de classe - Alterar Dados

Figura 34: Diagrama de classe - Cadastrar Funcionário

Figura 35: Diagrama de classe - Abertura de Conta

Figura 36: Diagrama de classe - Sair

Figura 37: Diagrama de classe - Cadastrar Funcionário

Figura 38: Modelo Lógico

Figura 39: Tela Login

Figura 40: Menu Funcionário

Figura 41: Tela de Cadastro de Cliente

Figura 42: Tela de Informações da Conta

Figura 43:Cadastro de Funcionários

Figura 44: Menu Cliente

Figura 45: Visualizar Extrato

**Lista de tabelas**

Tabela 1 - Regras de Negócio

Tabela 2 - Requisitos Funcionais

Tabela 3 - Requisitos Não funcionais

Tabela 4: Especificação de Caso de Uso - Consultar Saldo

Tabela 5: Especificação de Caso de Uso - Realizar Depósito

Tabela 6: Especificação de Caso de Uso - Realizar Saque

Tabela 7: Especificação de Caso de Uso - Exibir Extrato

Tabela 8: Especificação de Caso de Uso - Consultar Limite

Tabela 9: Especificação de Caso de Uso - Abrir Conta

Tabela 10: Especificação de Caso de Uso - Consultar dados

Tabela 11: Especificação de Caso de Uso - Gerar Relatório

Tabela 12: Especificação de Caso de Uso - Encerrar Conta

Tabela 13: Especificação de Caso de Uso - Alterar dados

Tabela 14: Especificação de Caso de Uso - Cadastrar Funcionário

Tabela 15: Termos e descrição

**SUMÁRIO**

**1. INTRODUÇÃO**

**2. OBJETIVO**

2.1 Objetivo Geral

2.2 Objetivos Específicos

**3**. **ABRANGÊNCIA DO SISTEMA**

3.1 Regras de Negócio

3.1 Requisitos funcionais

3.2 Requisitos não funcionais

3.3 Metodologia e Tecnologia Utilizada

**4. DOCUMENTAÇÃO E ANÁLISE**

4.3 Diagrama de atividades

4.1 Diagrama de Caso de Uso Geral

4.2 Diagramas de Caso de Uso Específicos

4.3 Especificação dos Casos de Uso

4.4 Diagramas de Sequências

4.5 Diagrama de Classe

4.6 Modelo lógico

**5. PROJETO DE INTERFACES**

5.1 Tela Inicial

5.2 Tela Login

5.3 Telas da funcionalidade principal

**6. CONCLUSÃO**

**7. GLOSSÁRIO**

**8. REFERÊNCIAS BIBLIOGRÁFICAS**

1. **INTRODUÇÃO**

O aumento do uso de tecnologias digitais para administração de recursos financeiros expõe a necessidade do desenvolvimento de aplicações eficientes e seguras para a gestão bancária. Partindo desse ponto, o projeto Banco Malvader disponibiliza um sistema desenvolvido em Java, com interface gráfica operada na biblioteca Swing e gerenciamento de dados MySQL. A finalidade é originar uma aplicação que incorpore princípios avançados de Programação Orientada a Objetos (POO) e persistência de dados, além das boas práticas de engenharia de software, na simulação de operações bancárias.

O sistema abrange recursos indispensáveis, tais como o gerenciamento de contas correntes e poupança, alterações e consultas cadastrais, disponibilização de relatórios financeiros e outras operações. A aplicação é organizada em uma arquitetura modular, que estrutura em pacotes os modelos, controladores, interface gráfica e acesso ao banco de dados. Dessa maneira, a separação de responsabilidades, reutilização de código e manutenção dos sistemas pode ser facilitada.

A interface gráfica foi projetada de forma intuitiva e dividida entre os perfis de usuário, proporcionando consistência, integridade e segurança das informações. Além disso, o Banco Malvader tem o objetivo de colocar em prática o conhecimento técnico e aprendizado avançado de desenvolvimento de sistemas.

1. **OBJETIVO**

## 2.1 Objetivo Geral

Desenvolvimento de um sistema bancário simulado, denominado Banco Malvader, utilizando a linguagem Java com Swing para a interface gráfica e MySQL como banco de dados, com foco em implementar conceitos avançados de Programação Orientada a Objetos, arquitetura modular e persistência de dados.

2.2 Objetivos Específicos

* Execução de funcionalidades de administração de contas, como cadastro, consulta, atualização e exclusão de dados.
* Criação de uma interface gráfica intuitiva e organizada, segmentada entre usuários cliente e funcionário.
* Integração de um banco de dados MySQL ao sistema.
* Aplicação de uma arquitetura modular, dividindo responsabilidades entre pacotes de modelos, controladores, interface gráfica e acesso ao banco de dados.
* Apresentação de relatórios financeiros detalhados.
* Assegurar boas práticas de desenvolvimento, como reutilização de código e aplicação de padrões de projeto.

1. **ABRANGÊNCIA DO SISTEMA**

3.1 Regras de Negócio

Tabela 1 - Regras de Negócio

| NÚMERO | NOME | DESCRIÇÃO | SETOR |
| --- | --- | --- | --- |
| RN01 | Cadastro de clientes | Possibilita o cadastro de novos clientes, armazenando informações pessoais e financeiras. |  |
| RN02 | Alteração de dados cadastrais | Permite a atualização de informações dos clientes, respeitando as regras de segurança. |  |
| RN03 | Consulta de saldo | Apresenta o saldo atualizado de contas correntes e poupança dos clientes. |  |
| RN04 | Movimentação de contas | Registra transações como depósitos, saques e transferências. |  |
| RN05 | Emissão de relatórios financeiros | Gera relatórios detalhados sobre movimentações e saldos. |  |
| RN06 | Acesso por perfil usuário | Restringe as opções do sistema com base no perfil, seja funcionário ou cliente. |  |
| RN07 | Validação de transações | Assegura que as transações financeiras respeitem limites e regras bancárias. |  |
| RN08 | Autenticação de usuários | Exige login com credenciais para acesso ao sistema. |  |
| RN09 | Exclusão de conta | Possibilita a exclusão de contas inativas, garantindo a preservação do histórico. |  |

3.1 Requisitos funcionais

Tabela 2 - Requisitos Funcionais

| NÚMERO | NOME | DESCRIÇÃO | RN |
| --- | --- | --- | --- |
| RF01 | Cadastrar cliente | Permitir que novos clientes sejam cadastrados no sistema, com validação dos dados inseridos. | RN01 |
| RF02 | Atualizar cadastro | Permitir a edição de informações cadastrais dos clientes após autenticação no sistema. | RN02 |
| RF03 | Consultar saldo | Exibir o saldo atualizado das contas vinculadas a um cliente. | RN03 |
| RF04 | Registrar Movimentação | Registrar operações financeiras, como depósitos, saques e transferências, atualizando o saldo. | RN04 |
| RF05 | Gerar Relatórios Financeiros | Disponibilizar relatórios detalhados sobre movimentações financeiras. | RN05 |
| RF06 | Controlar o acesso | Restringir funcionalidades disponíveis com base no perfil do usuário (cliente ou funcionário). | RN06 |
| RF07 | Validar Transações | Validar operações financeiras conforme limites definidos e regras do banco. | RN07 |
| RF08 | Autenticar Usuário | Exigir login com credenciais para acesso às funcionalidades do sistema. | RN09 |
| RF09 | Excluir Contas | Permitir que contas inativas sejam excluídas, mantendo o histórico no banco de dados. | RN10 |
| RF10 | Visualizar extrato | Permitir ao cliente consultar o extrato detalhado de movimentações financeiras da conta | RN03 |

3.2 Requisitos não funcionais

Tabela 3 - Requisitos não funcionais

| NÚMERO | NOME | DESCRIÇÃO | RN |
| --- | --- | --- | --- |
| RNF01 | Performance do sistema | O sistema deve responder a qualquer ação do usuário de maneira rápida e eficiente. | RN01, RN04 |
| RNF02 | Escalabilidade | A arquitetura do sistema deve permitir expansão para suporte a diversas contas ativas. | RN06 |
| RNF03 | Segurança de dados | Todas as informações devem ser armazenadas de forma criptografada no banco de dados. | RN09 |
| RNF04 | Conformidade Legal | O sistema deve seguir as leis de proteção de dados (como LGPD ou GDPR). | RN09 |
| RNF05 | Compatibilidade | O software deve funcionar em sistemas operacionais Windows, macOS e Linux. | RN06 |
| RNF06 | Manutenibilidade | A arquitetura modular deve facilitar ajustes ou inclusão de novas funcionalidades. | RN08 |
| RNF07 | Disponibilidade | O sistema deve estar disponível para uso, salvo períodos de manutenção. | RN05 |
| RNF08 | Usabilidade | A interface gráfica deve ser intuitiva, com elementos organizados e visualmente agradáveis. | RN02 |
| RNF09 | Documentação | O projeto deve conter documentação detalhada para desenvolvedores e administradores. | RN08 |

3.3 Metodologia e Tecnologia Utilizada

Durante o desenvolvimento do Banco Malvader, foi utilizada uma metodologia

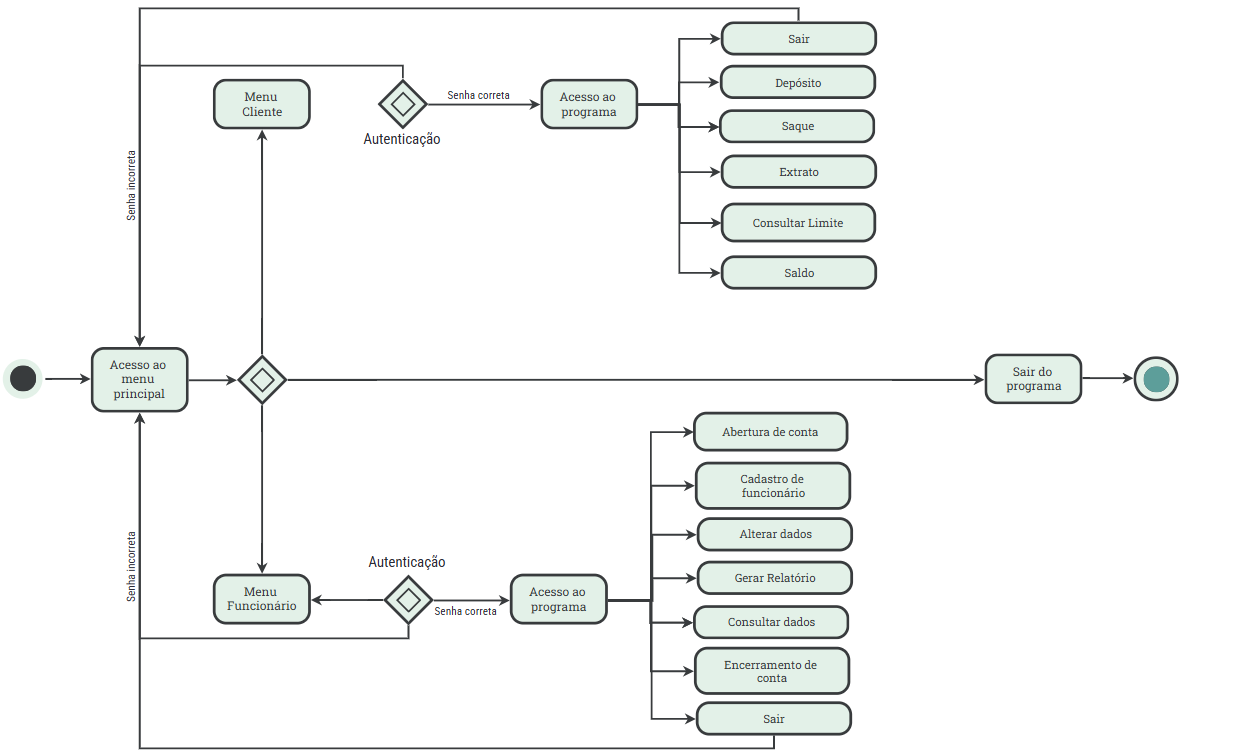
inspirada nos princípios do Scrum, seguindo etapas de planejamento, desenvolvimento, testes e entrega.

Quanto a implementação do backend e interface gráfica foi utilizada a linguagem de programação Java com suporte a Programação Orientada a Objetos juntamente com Swing. O MySQL foi usado para armazenar informações das contas e realizar operações de persistência de dados, o GitHub para o controle de versão e gerenciamento do código-fonte e a arquitetura modular.

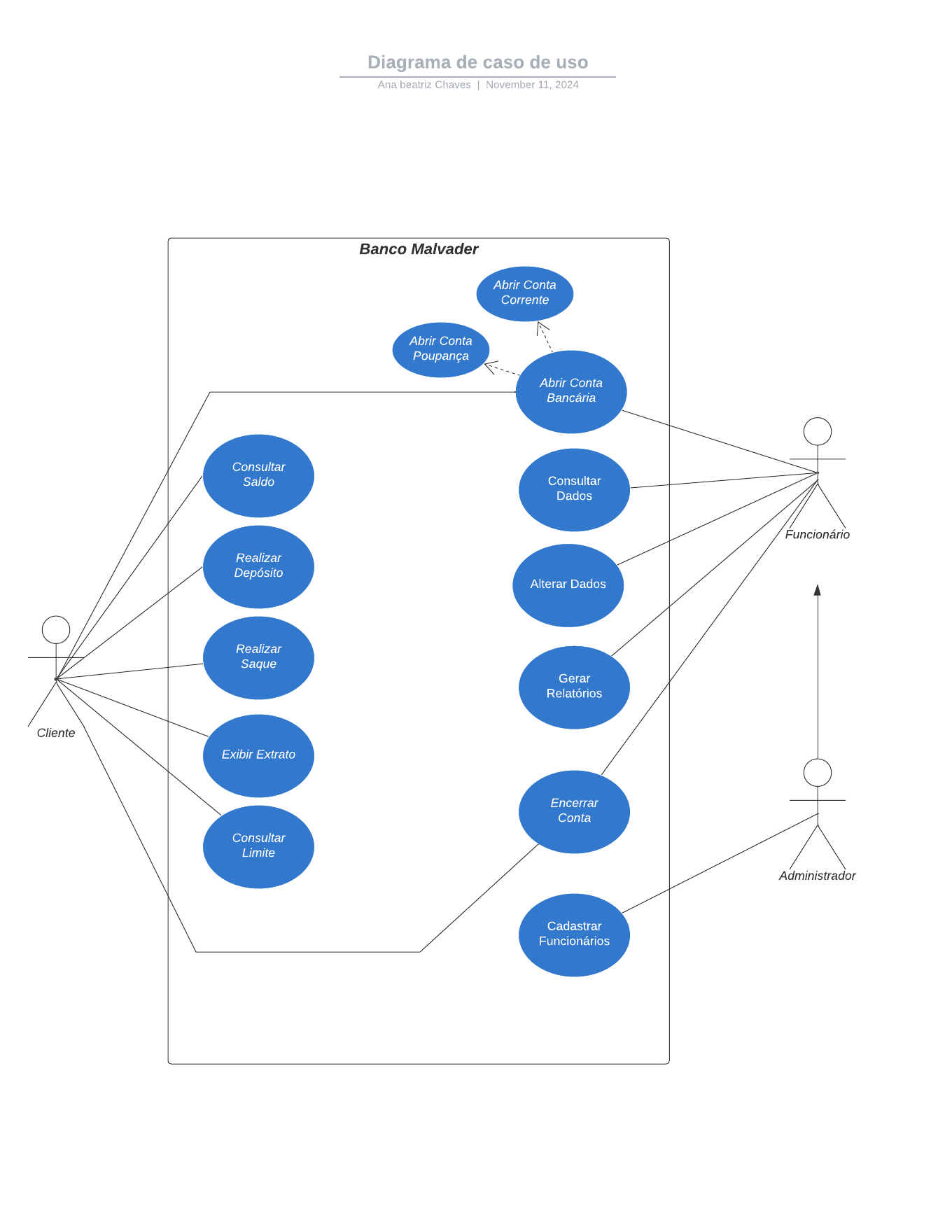
A tecnologia e metodologia utilizadas foram escolhidas para que a combinação das mesmas assegurasse um desenvolvimento eficaz e um produto final funcional.

1. **DOCUMENTAÇÃO E ANÁLISE**

4.3 Diagrama de atividades

Figura 1: Diagrama de atividades

4.1 Diagrama de Caso de Uso Geral

Figura 2: Diagrama de Caso de Uso Geral

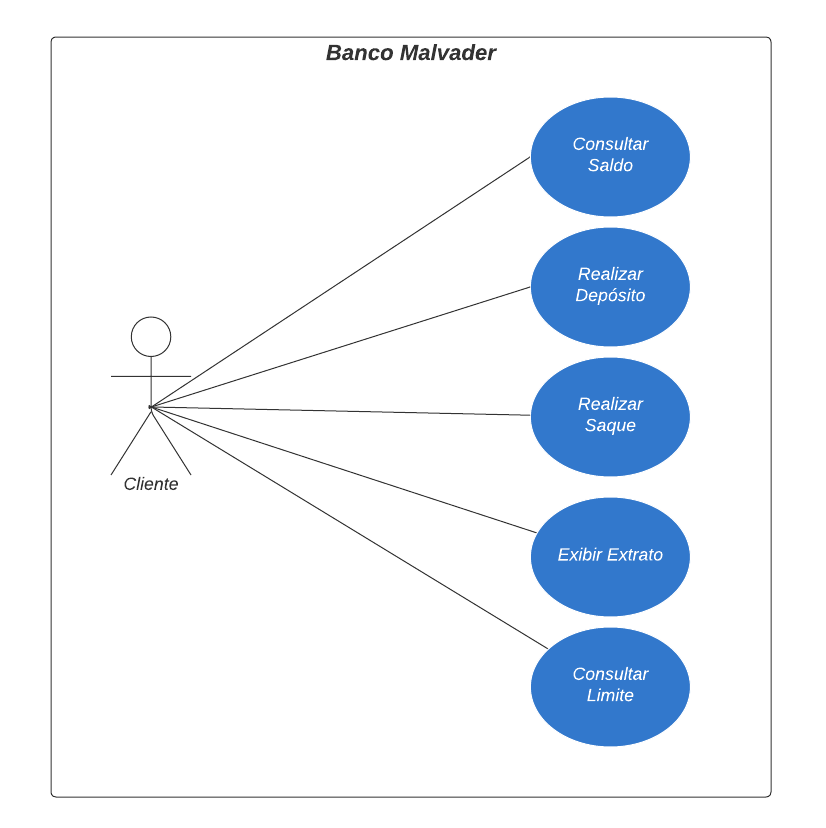
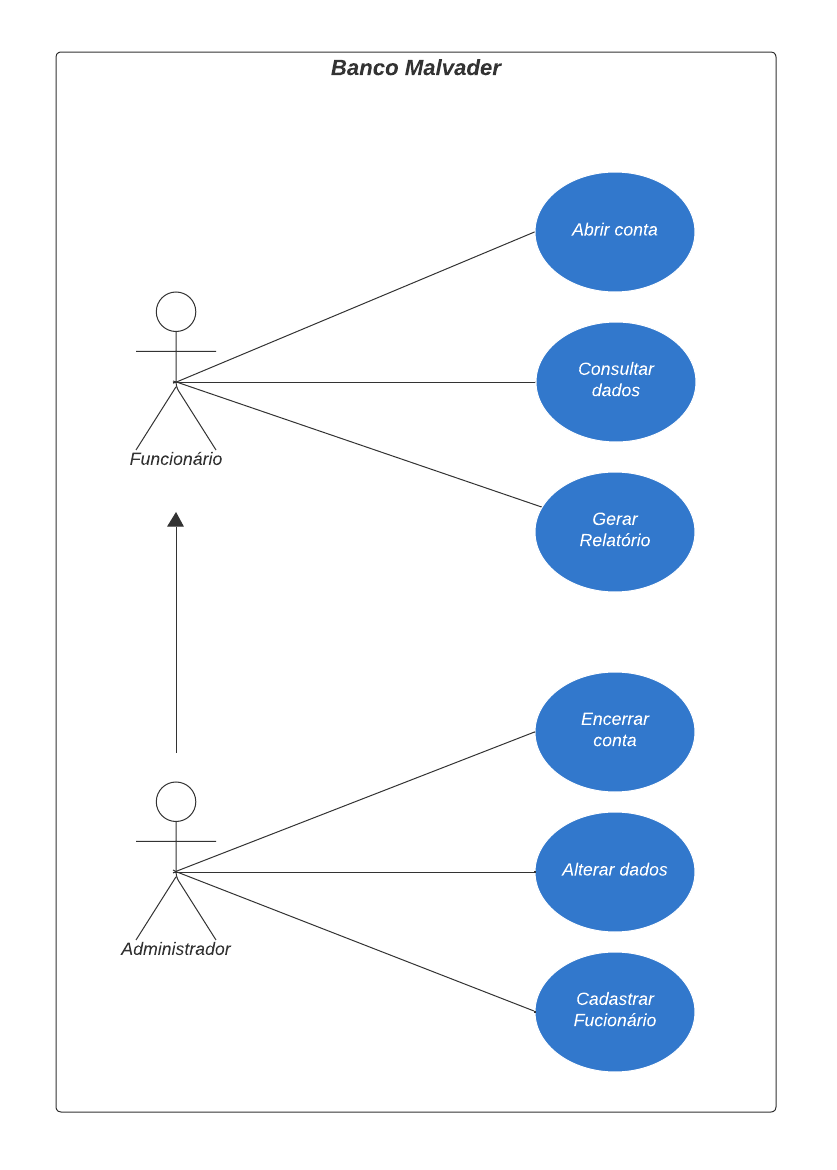
4.2 Diagramas de Caso de Uso Específicos

Figura 3: Diagrama de Caso de Uso Específico - Cliente

Figura 4: Diagrama de Caso de Uso Específico - Funcionário / Administrador

4.3 Especificação dos Casos de Uso

Tabela 4: Especificação de Caso de Uso - Consultar Saldo

| Identificador | UC - Cliente |
| --- | --- |
| Nome e descrição | Permite que o ator consulte o saldo no sistema. |
| Ator | Cliente. |
| Prioridade | Alta. |
| Requisito Não Funcional | Não possui. |
| Pré-condição | O ator deve estar autenticado no sistema no perfil do cliente. |
| Pós-condição | O sistema deve permitir acesso ao perfil correspondente. |
| Fluxo Principal | P1. O ator seleciona a opção “Saldo”;  P2. O sistema exibe uma tela para inserir a senha do cliente;  P3. O sistema exibe a página com o saldo do cliente;  P4. Caso de uso encerrado. |
| Fluxo alternativo | Não Possui. |
| Fluxo de Exceção | P2.1 Caso a senha esteja incorreta o sistema é redirecionado para a página de menu cliente;  P2.2 Caso a senha esteja correta é mostrado o saldo em tela; |

Tabela 5: Especificação de Caso de Uso - Realizar Depósito

| Identificador | UC - Cliente |
| --- | --- |
| Nome e descrição | Permite que o ator deposite valores em sua conta. |
| Ator | Cliente. |
| Prioridade | Alta. |
| Requisito Não Funcional | Não possui. |
| Pré-condição | O ator deve estar autenticado no sistema no perfil do cliente. |
| Pós-condição | O sistema deve permitir acesso ao perfil correspondente. |
| Fluxo Principal | P1. O ator seleciona a opção “Depósito”;  P2. O sistema exibe uma tela para inserir o valor a ser depositado;  P3. O sistema registra os valores no banco de dados;  P4. Caso de uso encerrado. |
| Fluxo alternativo | Não Possui. |
| Fluxo de Exceção | Não Possui. |

Tabela 6: Especificação de Caso de Uso - Realizar Saque

| Identificador | UC - Cliente |
| --- | --- |
| Nome e descrição | Permite que o ator saque valores de sua conta no sistema. |
| Ator | Cliente. |
| Prioridade | Alta. |
| Requisito Não Funcional | Não possui. |
| Pré-condição | O ator deve estar autenticado no sistema no perfil do cliente. |
| Pós-condição | O sistema deve permitir acesso ao perfil correspondente. |
| Fluxo Principal | P1. O ator seleciona a opção “Saque”;  P2. O sistema exibe uma tela para inserir a senha do cliente;  P3. O sistema exibe a página para ser inserido o valor do saque;  P4. O sistema debita o valor do saldo.  P5. Caso de uso encerrado. |
| Fluxo alternativo | Não Possui. |
| Fluxo de Exceção | P2.1 Caso a senha esteja incorreta o sistema é redirecionado para a página de menu cliente;  P2.2 Caso a senha esteja correta é solicitado o valor do saque;  P3.1 Caso o valor do saque seja maior do que o saldo em conta é exibida a mensagem “Saldo insuficiente”; |

Tabela 7: Especificação de Caso de Uso - Exibir Extrato

| Identificador | UC - Cliente |
| --- | --- |
| Nome e descrição | Permite que o ator retire o extrato das movimentações em conta no sistema e seja exportado em forma de relatório. |
| Ator | Cliente. |
| Prioridade | Alta. |
| Requisito Não Funcional | Não possui. |
| Pré-condição | O ator deve estar autenticado no sistema no perfil do cliente. |
| Pós-condição | O sistema deve permitir acesso ao perfil correspondente. |
| Fluxo Principal | P1. O ator seleciona a opção “Extrato”;  P2. O sistema exibe uma tela para inserir a senha do cliente;  P3. O sistema exibe a página com o extrato do cliente em formato excel;  P4. Caso de uso encerrado. |
| Fluxo alternativo | Não Possui. |
| Fluxo de Exceção | P2.1 Caso a senha esteja incorreta o sistema é redirecionado para a página de menu cliente;  P2.2 Caso a senha esteja correta é mostrado o extrato em tela no excel; |

Tabela 8: Especificação de Caso de Uso - Consultar Limite

| Identificador | UC - Cliente |
| --- | --- |
| Nome e descrição | Permite que o ator consulte o limite disponível em conta no sistema. |
| Ator | Cliente. |
| Prioridade | Alta. |
| Requisito Não Funcional | Não possui. |
| Pré-condição | O ator deve estar autenticado no sistema no perfil do cliente. |
| Pós-condição | O sistema deve permitir acesso ao perfil correspondente. |
| Fluxo Principal | P1. O ator seleciona a opção “Consultar Limite”;  P2. O sistema exibe uma tela para inserir a senha do cliente;  P3. O sistema exibe em tela o status do limite em conta;  P4. Caso de uso encerrado. |
| Fluxo alternativo | Não Possui. |
| Fluxo de Exceção | P2.1 Caso a senha esteja incorreta o sistema é redirecionado para a página de menu cliente;  P2.2 Caso a senha esteja correta é mostrado em tela se o limite está disponível ou indisponível; |

Tabela 9: Especificação de Caso de Uso - Abrir Conta

| Identificador | UC - Funcionário / Administrador |
| --- | --- |
| Nome e descrição | Permite que o ator realize a abertura de novas contas bancárias no sistema. |
| Ator | Funcionário / Administrador. |
| Prioridade | Alta. |
| Requisito Não Funcional | Não possui. |
| Pré-condição | O ator deve estar autenticado no sistema no perfil do Funcionário. |
| Pós-condição | O sistema deve permitir acesso ao perfil correspondente. |
| Fluxo Principal | P1. O ator seleciona a opção “Abertura de conta”;  P2. O sistema exibe uma tela um submenu com opções de conta poupança, conta corrente ou voltar para o menu funcionário;  P3. O ator escolhe uma opção;  P4. Caso de uso encerrado. |
| Fluxo alternativo | P2.1 Caso o ator clique na opção “Conta Poupança - CP”, o sistema irá exibir os campos para preenchimento do cadastro;  P2.2 Caso o ator clique na opção “Conta Corrente - CC”, o sistema irá exibir os campos para preenchimento do cadastro;  P2.3 Caso o ator clique na opção “Voltar”, o sistema irá ser redirecionado ao menu funcionário. |
| Fluxo de Exceção | Não Possui. |

Tabela 10:: Especificação de Caso de Uso - Consultar dados

| Identificador | UC - Funcionário / Administrador |
| --- | --- |
| Nome e descrição | Permite que o ator consulte os dados de uma conta, de um funcionário ou de um cliente no sistema. |
| Ator | Funcionário. |
| Prioridade | Alta. |
| Requisito Não Funcional | Não possui. |
| Pré-condição | O ator deve estar autenticado no sistema no perfil do funcionário. |
| Pós-condição | O sistema deve permitir acesso ao perfil correspondente. |
| Fluxo Principal | P1. O ator seleciona a opção “Consultar Dados”;  P2. O sistema exibe um submenu com opções de consultar conta, consultar funcionário, consultar cliente ou voltar;  P3. O ator escolhe uma opção;  P4. Caso de uso encerrado. |
| Fluxo alternativo | Não Possui. |
| Fluxo de Exceção | P2.1 Caso o ator clique na opção “Consultar conta”, o sistema irá exibir os campos de preenchimento para realizar a busca;  P2.2 Caso o ator clique na opção “Consultar funcionário”, o sistema irá exibir os campos de preenchimento para realizar a busca;  P2.3 Caso o ator clique na opção “Consultar cliente”, o sistema irá exibir os campos de preenchimento para realizar a busca;  P2.4 Caso o ator clique na opção “Voltar”, o sistema irá ser redirecionado ao menu funcionário. |

Tabela 11: Especificação de Caso de Uso - Gerar Relatório

| Identificador | UC - Funcionário / Administrador |
| --- | --- |
| Nome e descrição | Permite que o ator gere relatórios no sistema. |
| Ator | Funcionário. |
| Prioridade | Alta. |
| Requisito Não Funcional | Não possui. |
| Pré-condição | O ator deve estar autenticado no sistema no perfil de funcionário. |
| Pós-condição | O sistema deve permitir acesso ao perfil correspondente. |
| Fluxo Principal | P1. O ator seleciona a opção “Gerar Relatórios”;  P2. O sistema exibe uma opção de “Relatório Geral”;  P3. O sistema exibe uma tela para inserir a senha do funcionário;  P4. O sistema exibe em tela o relatório de movimentação exportado para excel.  P5. Caso de uso encerrado. |
| Fluxo alternativo | Não Possui. |
| Fluxo de Exceção | P3.1 Caso a senha esteja incorreta o sistema é redirecionado para a página de menu funcionário;  P3.2 Caso a senha esteja correta é mostrado em tela o relatório de movimentação exportado para excel. |

Tabela 12: Especificação de Caso de Uso - Encerrar Conta

| Identificador | UC - Funcionário / Administrador |
| --- | --- |
| Nome e descrição | Permite que o ator encerre determinada conta no sistema. |
| Ator | Administrador. |
| Prioridade | Alta. |
| Requisito Não Funcional | Não possui. |
| Pré-condição | O ator deve estar autenticado no sistema no perfil do Funcionário com permissão de administrador. |
| Pós-condição | O sistema deve permitir acesso ao perfil correspondente. |
| Fluxo Principal | P1. O ator seleciona a opção “Encerramento de conta”;  P2. O sistema exibe uma tela para inserir a senha de administrador;  P3. O sistema exibe em tela um campo para digitar a conta para encerramento;  P4. A conta é encerrada com sucesso;  P5. Caso de uso encerrado. |
| Fluxo alternativo | Não Possui. |
| Fluxo de Exceção | P2.1 Caso a senha esteja incorreta o sistema é redirecionado para a página de menu funcionário;  P2.2 Caso a senha esteja correta é mostrado em tela um campo para digitar a conta para encerramento. |

Tabela 13: Especificação de Caso de Uso - Alterar dados

| Identificador | UC - Funcionário / Administrador |
| --- | --- |
| Nome e descrição | Permite que o ator altere os dados de uma conta, um funcionário ou um cliente no sistema. |
| Ator | Administrador. |
| Prioridade | Alta. |
| Requisito Não Funcional | Não possui. |
| Pré-condição | O ator deve estar autenticado no sistema no perfil do funcionário com permissão de administrador. |
| Pós-condição | O sistema deve permitir acesso ao perfil correspondente. |
| Fluxo Principal | P1. O ator seleciona a opção “Alterar dados”;  P2. O sistema exibe um submenu com opções de alterar conta, alterar funcionário, alterar cliente ou voltar;  P3. O sistema exibe uma tela para inserir a senha de administrador;  P4. O ator escolhe uma opção;  P5. Caso de uso encerrado. |
| Fluxo alternativo | Não Possui. |
| Fluxo de Exceção | P2.1 Caso o ator clique na opção “Alterar conta”, o sistema irá exibir os campos de preenchimento para realizar a alteração;  P2.2 Caso o ator clique na opção “Alterar funcionário”, o sistema irá exibir os campos de preenchimento para realizar a alteração;  P2.3 Caso o ator clique na opção “Alterar cliente”, o sistema irá exibir os campos de preenchimento para realizar a alteração;  P2.4 Caso o ator clique na opção “Voltar”, o sistema irá ser redirecionado ao menu funcionário. |

Tabela 14: Especificação de Caso de Uso - Cadastrar Funcionário

| Identificador | UC - Funcionário / Administrador |
| --- | --- |
| Nome e descrição | Permite que o ator realize novos cadastros de funcionários no sistema. |
| Ator | Administrador. |
| Prioridade | Alta. |
| Requisito Não Funcional | Não possui. |
| Pré-condição | O ator deve estar autenticado no sistema no perfil do funcionário com permissão de administrador. |
| Pós-condição | O sistema deve permitir acesso ao perfil correspondente. |
| Fluxo Principal | P1. O ator seleciona a opção “Cadastro de funcionário”;  P2. O sistema exibe uma tela para inserir a senha de administrador;  P3. O sistema exibe em tela os campos para preenchimento de cadastro;  P4. O sistema exibe a mensagem “Conta cadastrada com sucesso!”;  P5. Caso de uso encerrado. |
| Fluxo alternativo | Não Possui. |
| Fluxo de Exceção | P2.1 Caso a senha esteja incorreta o sistema é redirecionado para a página de menu funcionário;  P2.2 Caso a senha esteja correta é mostrado em tela os campos para preenchimento de cadastro; |

4.4 Diagramas de Sequências

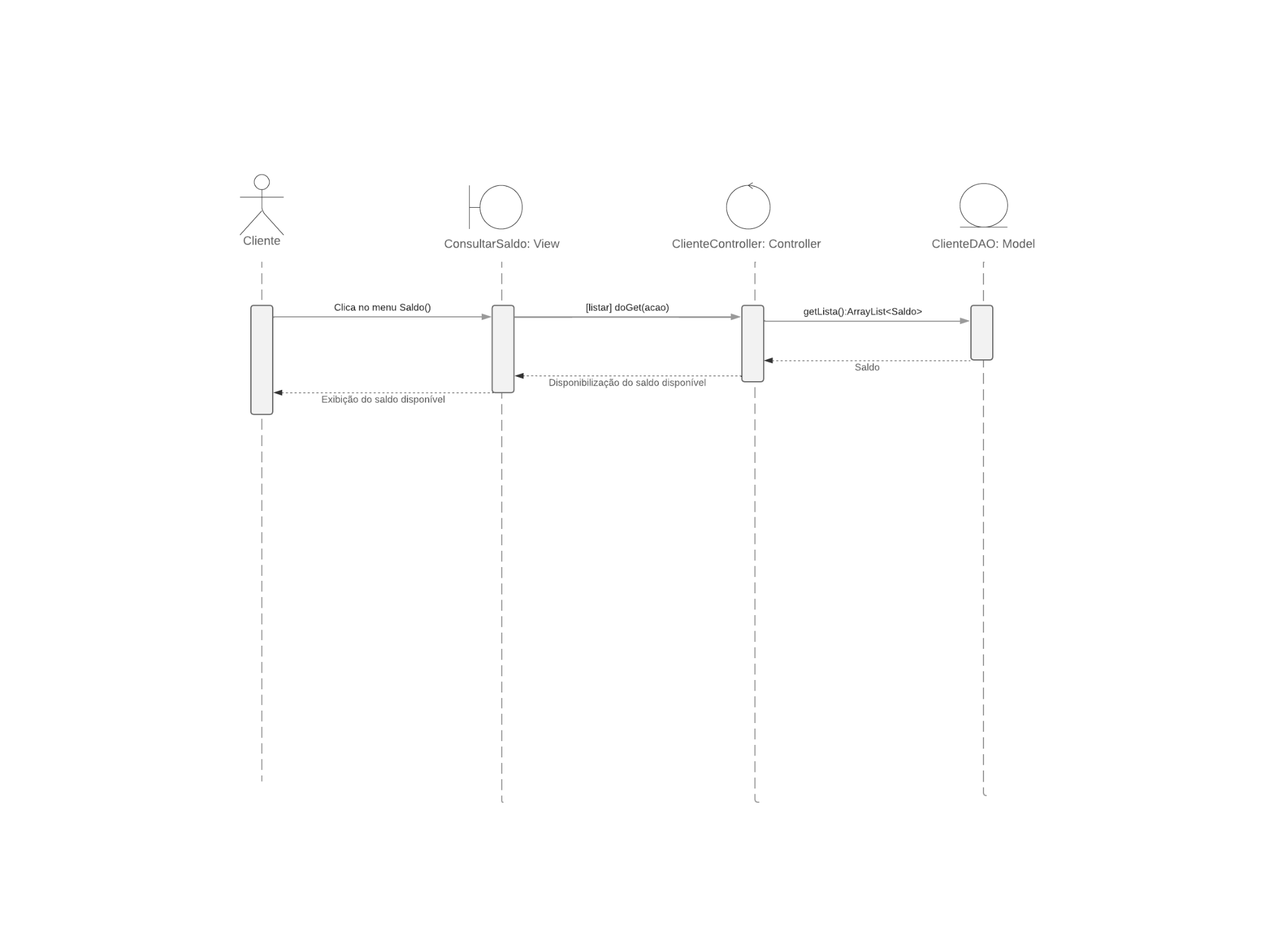
Figura 5: Diagrama de sequência - Consultar Saldo

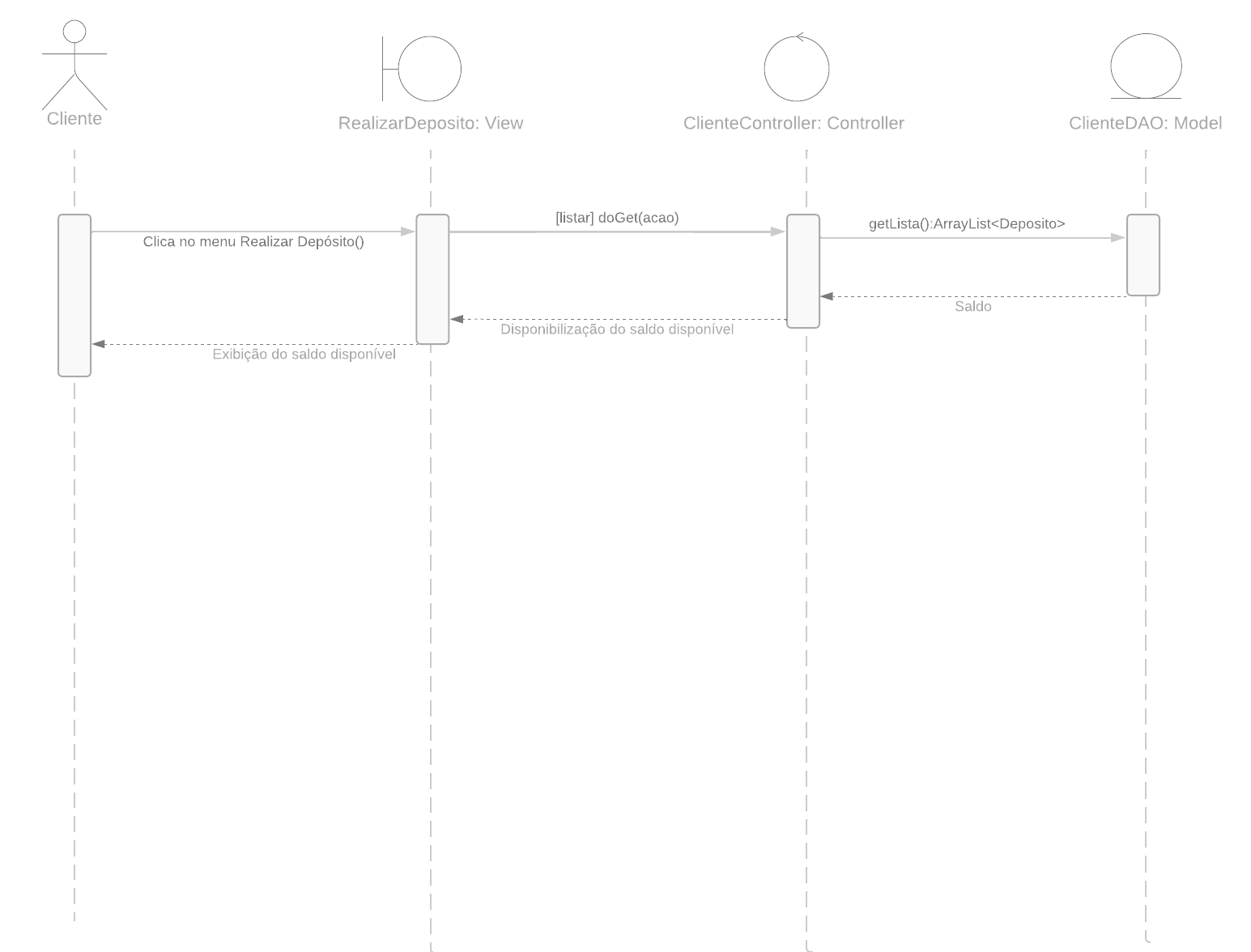
Figura 6: Diagrama de sequência - Fazer Depósito  
  


Figura 7: Diagrama de sequência - Fazer Saque

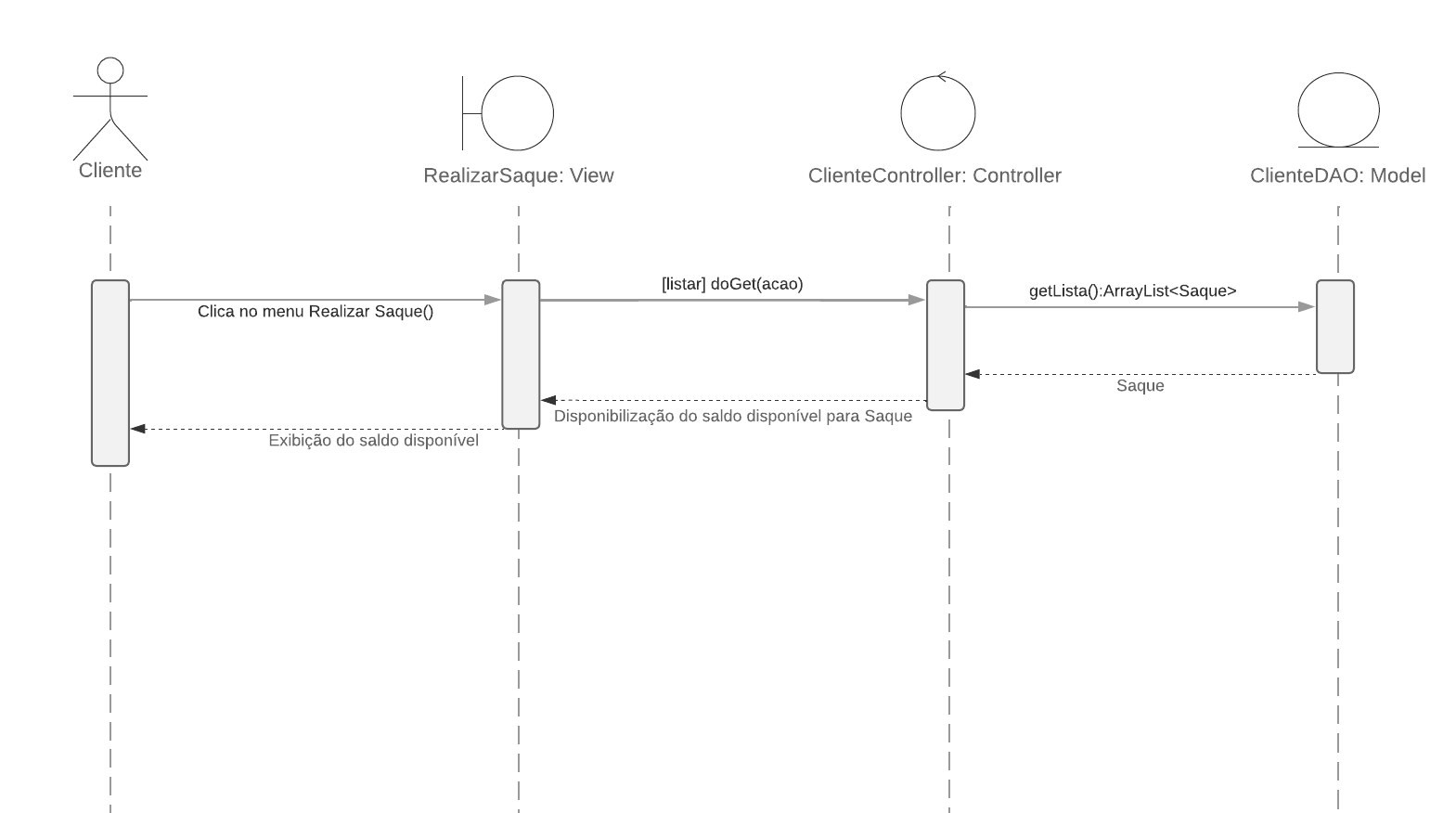


Figura 8: Diagrama de sequência - Exibir Extrato

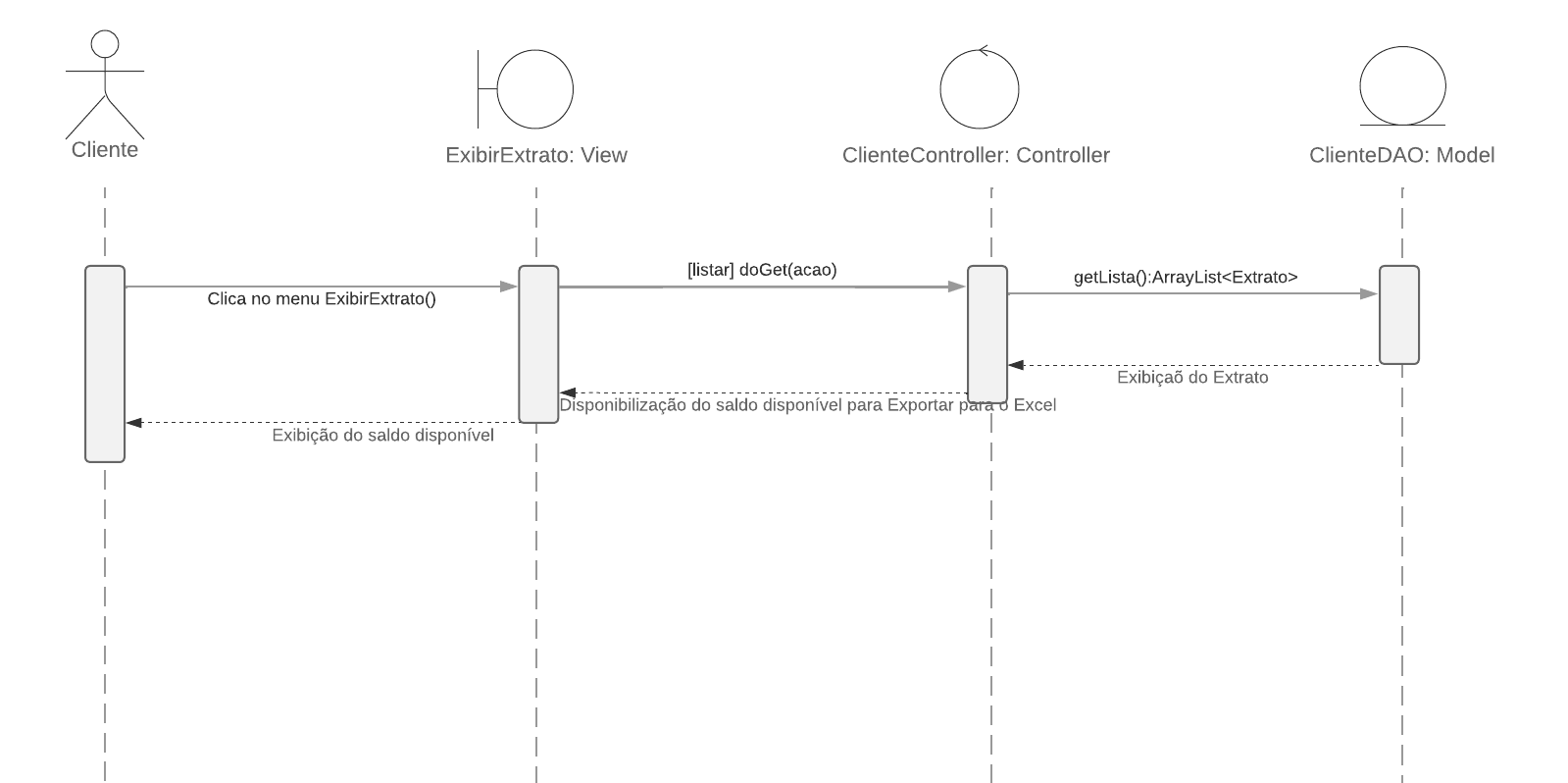
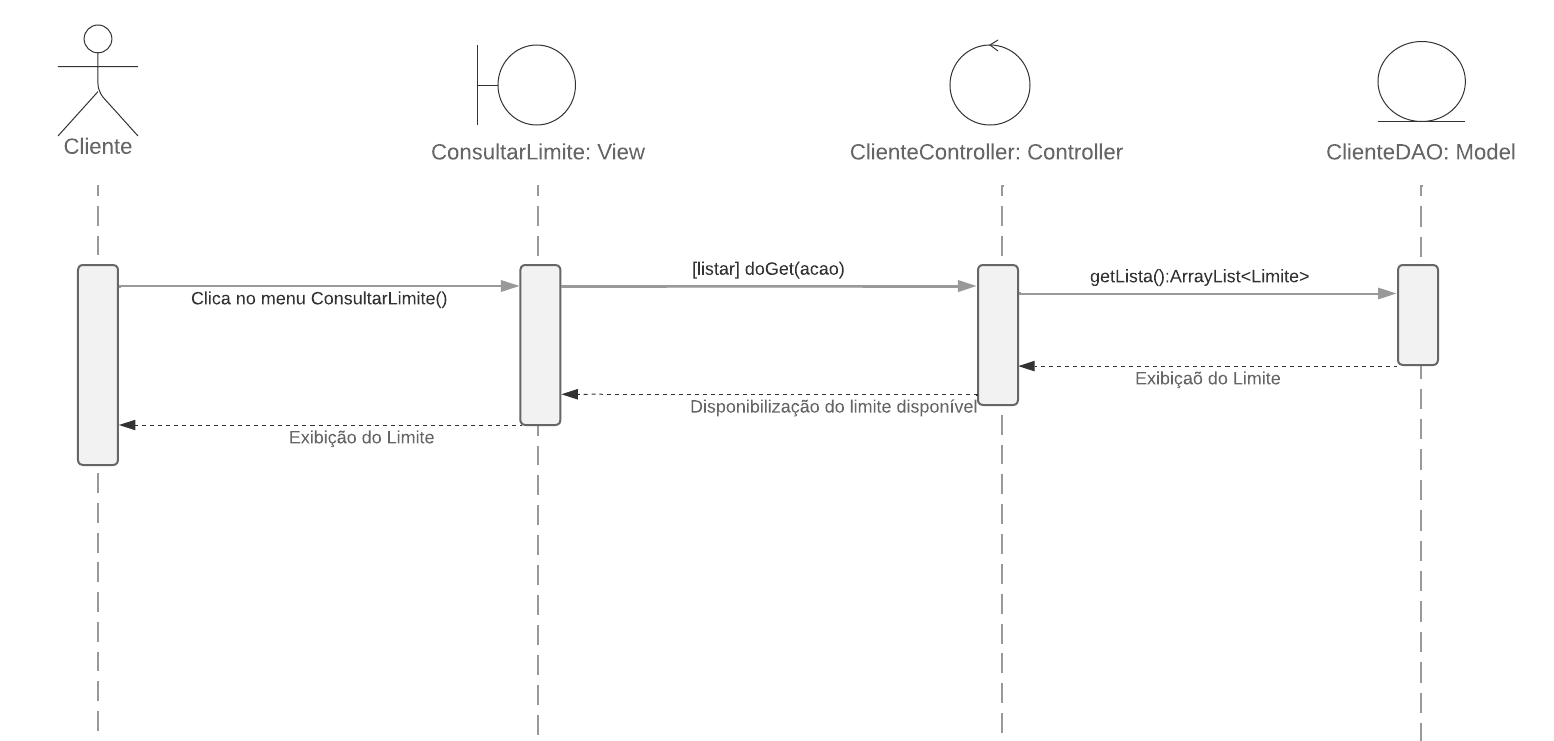


Figura 9: Diagrama de sequência - Consultar Limite



4.5 Diagrama de Classe

Figura 10: Diagrama de classe - Caso de uso cliente

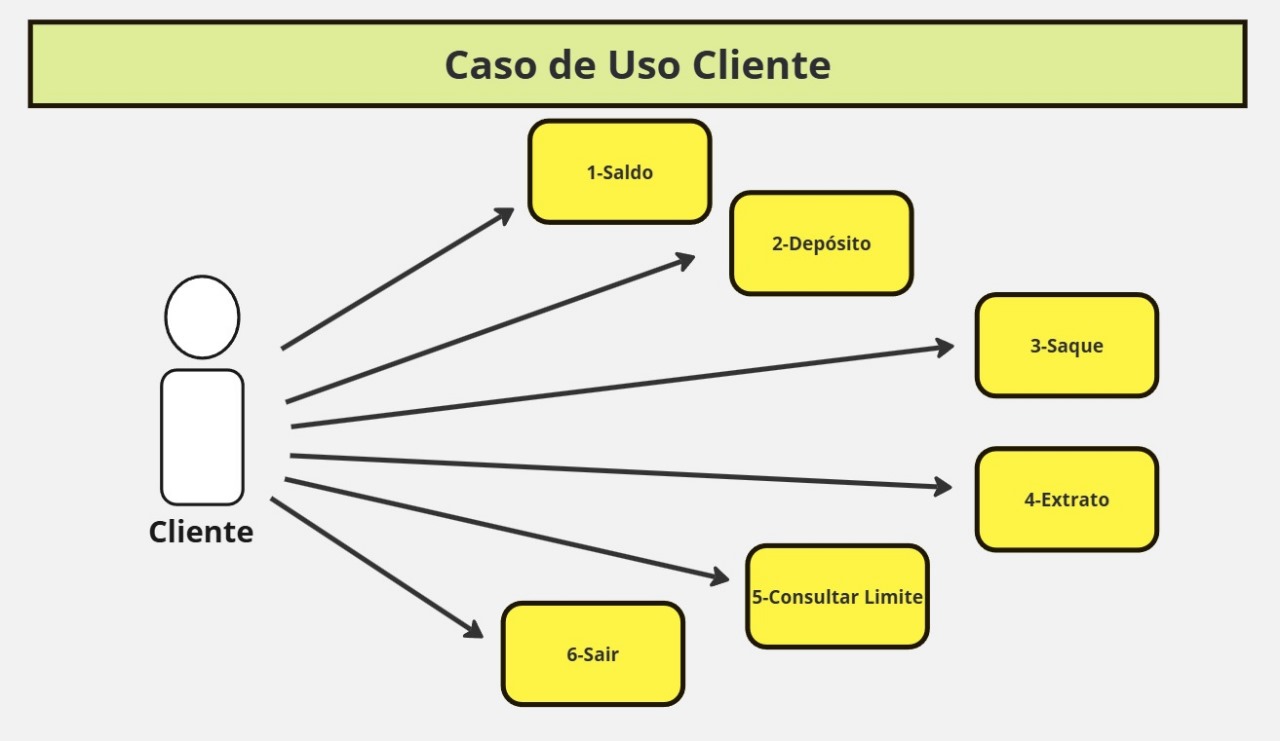


Figura 11: Diagrama de classe - Saque

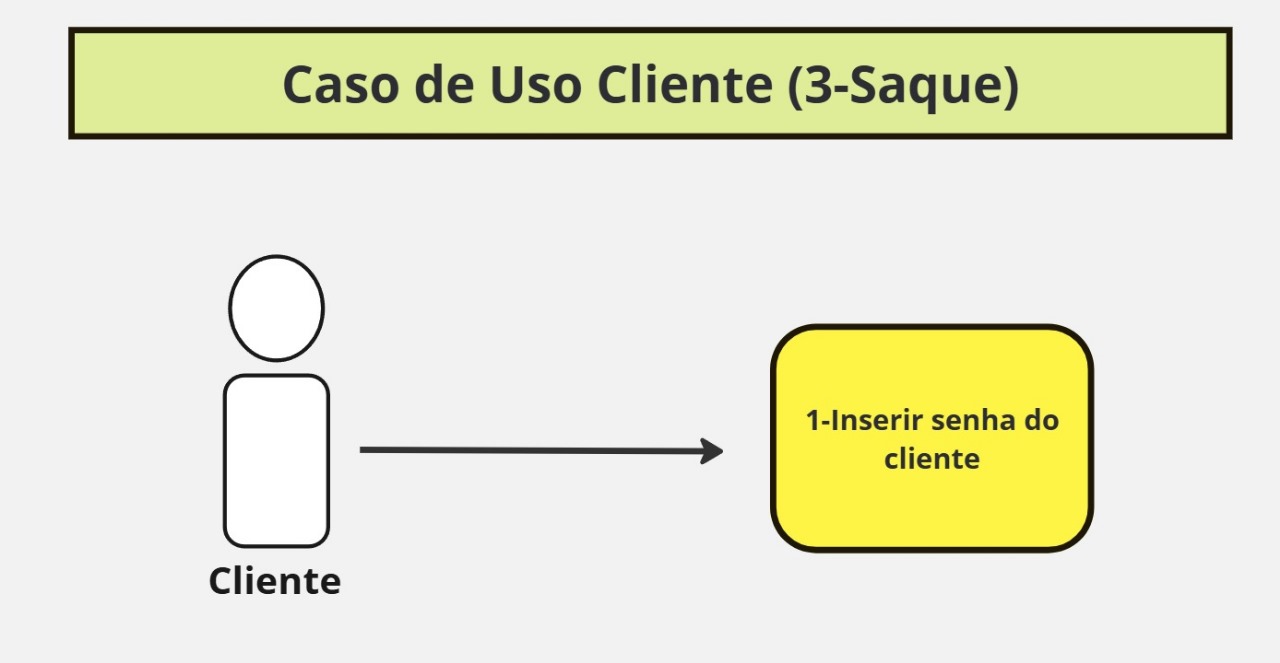


Figura 12: Diagrama de classe - Saldo

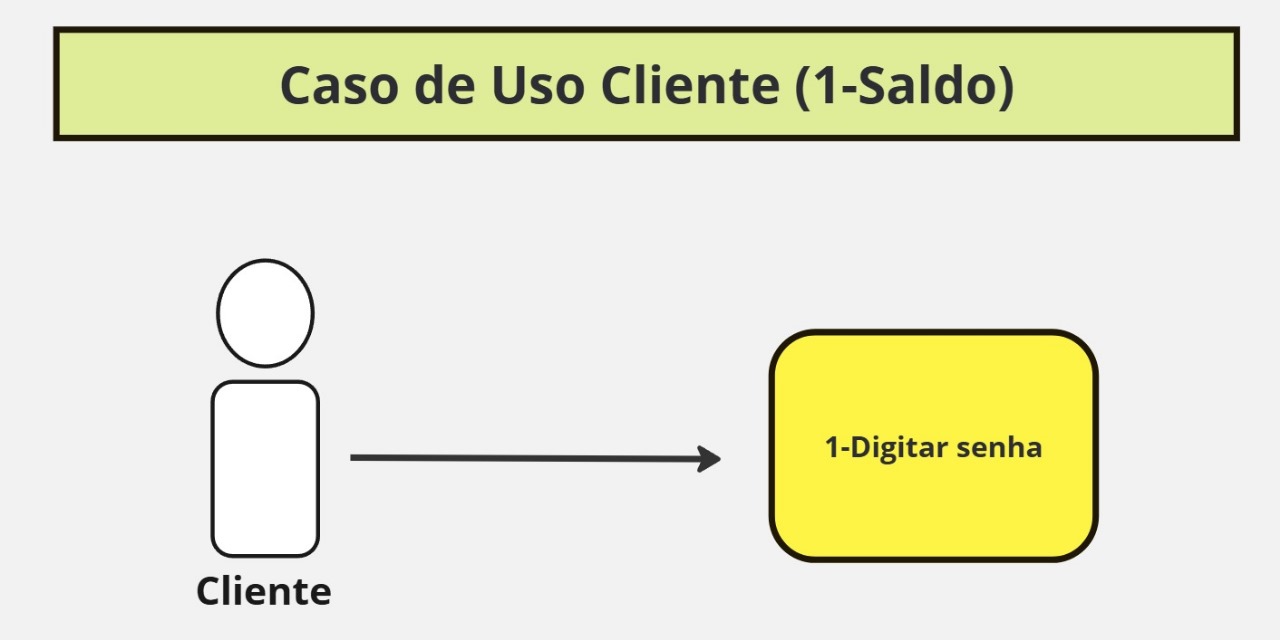


Figura 13: Diagrama de classe - Extrato

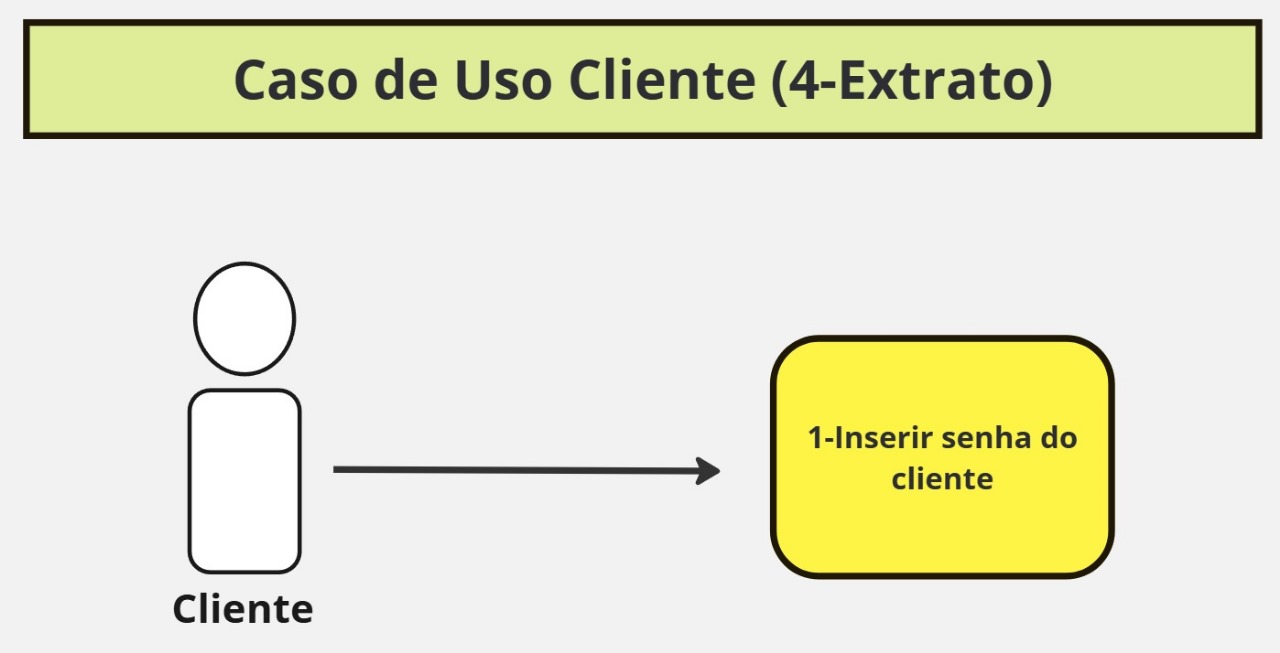


Figura 14: Diagrama de classe - Depósito

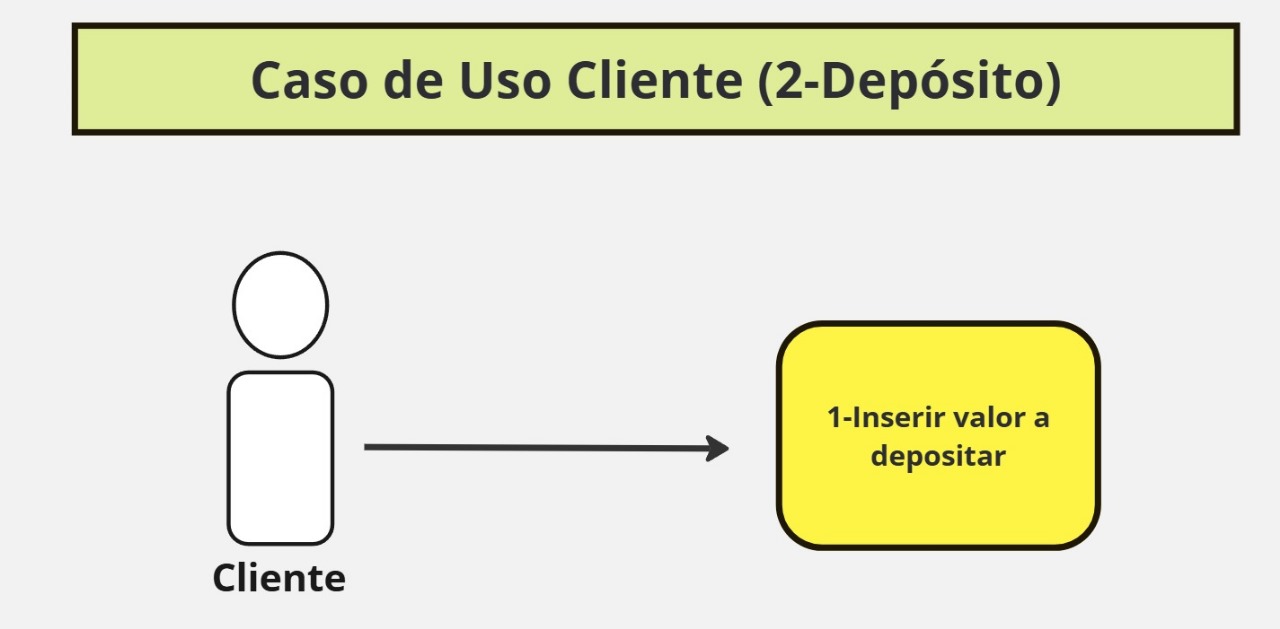


Figura 15: Diagrama de classe - Funcionário

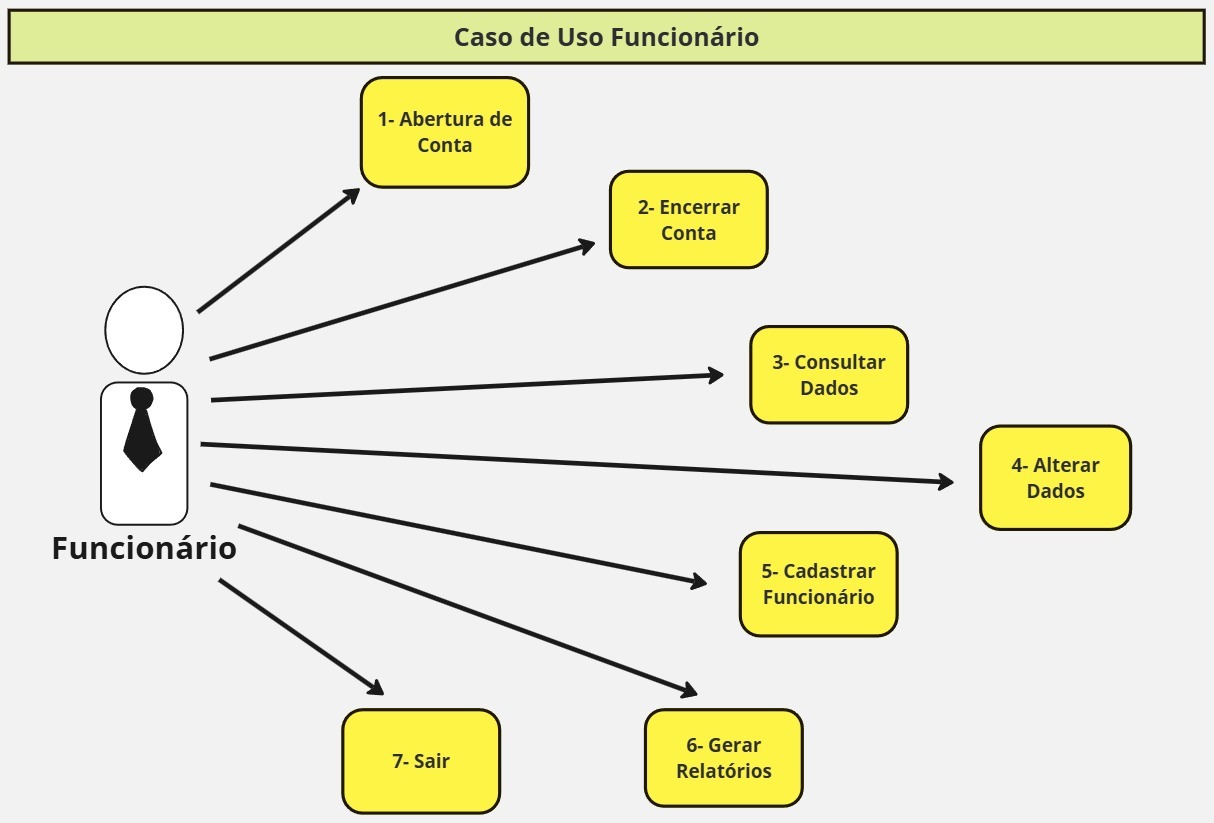


Figura 16: Diagrama de classe - Sair

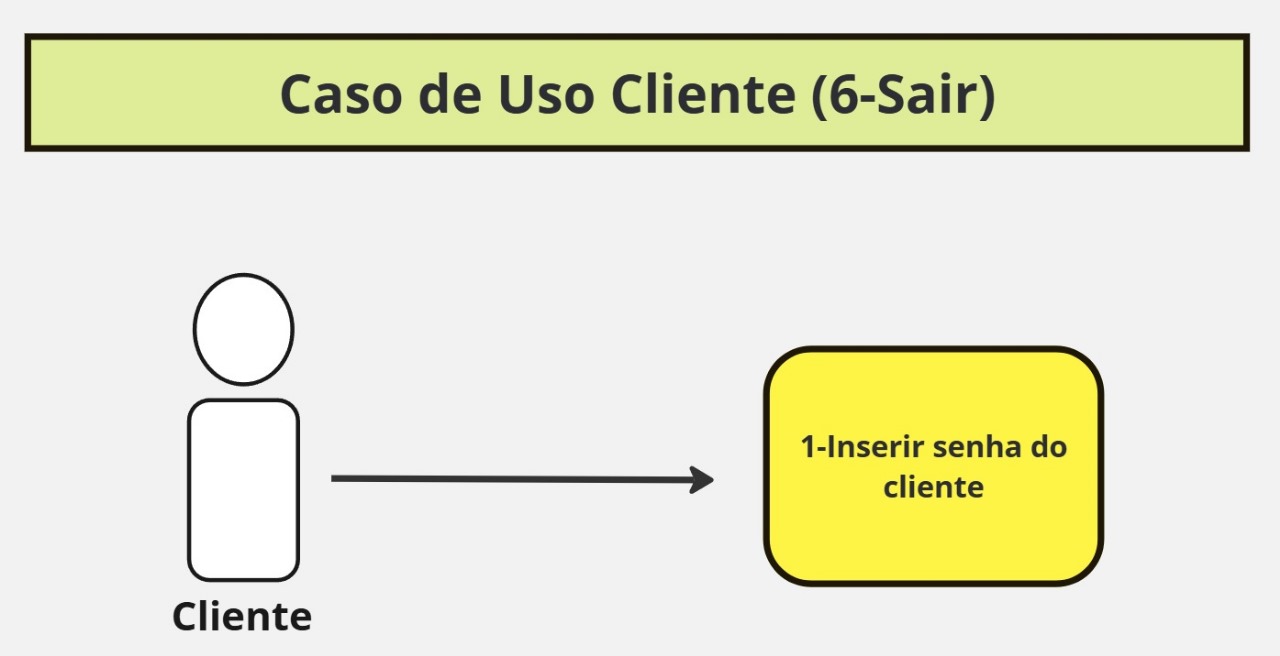


Figura 17: Diagrama de classe - Consultar Limite

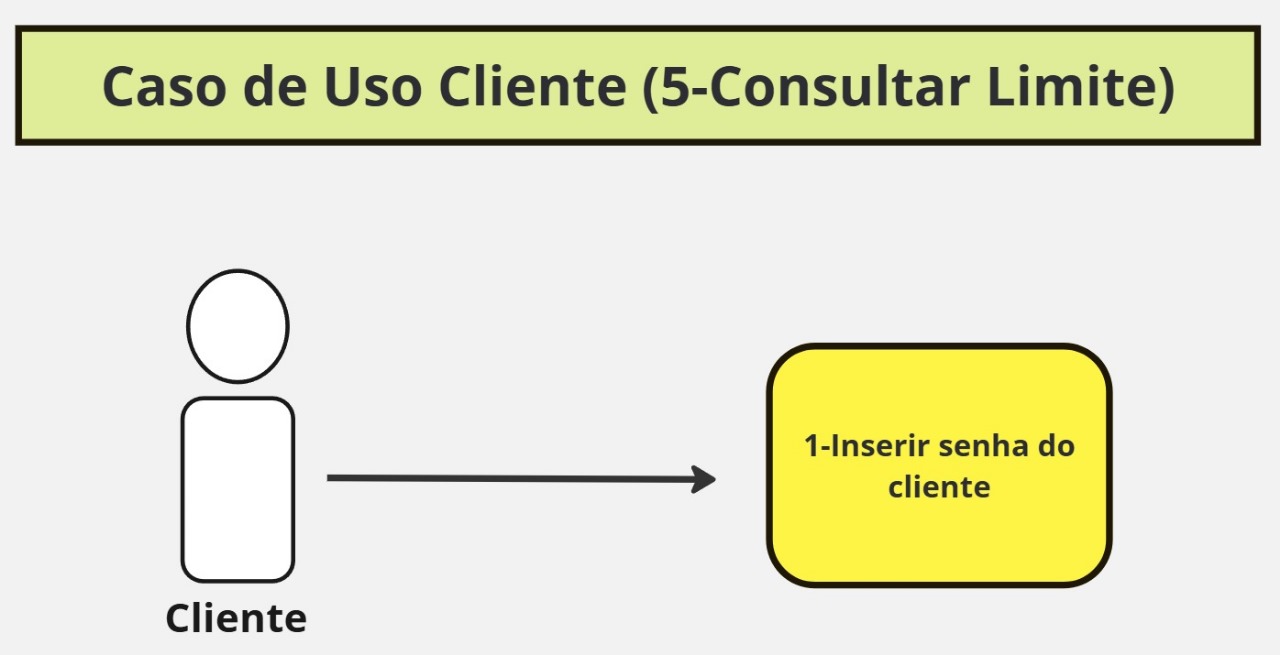


Figura 18: Diagrama de classe - Abertura de Conta

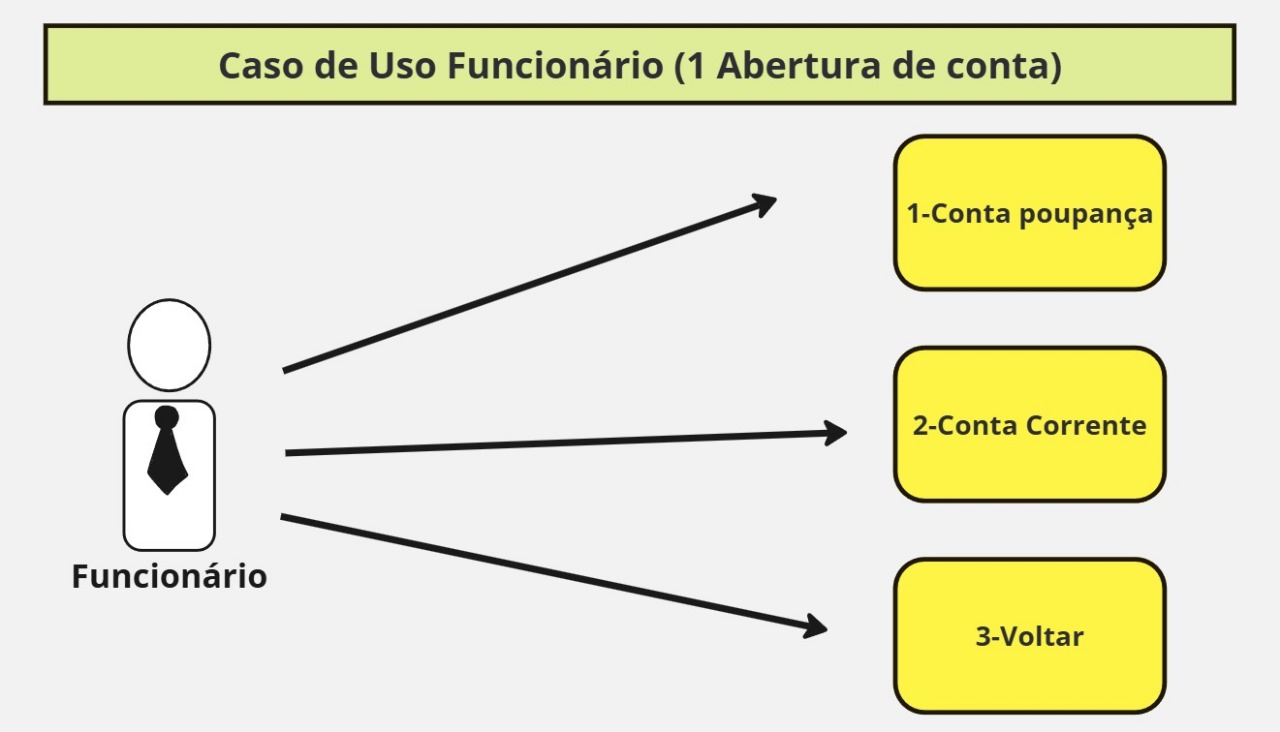


Figura 19: Diagrama de classe - Alterar Dados

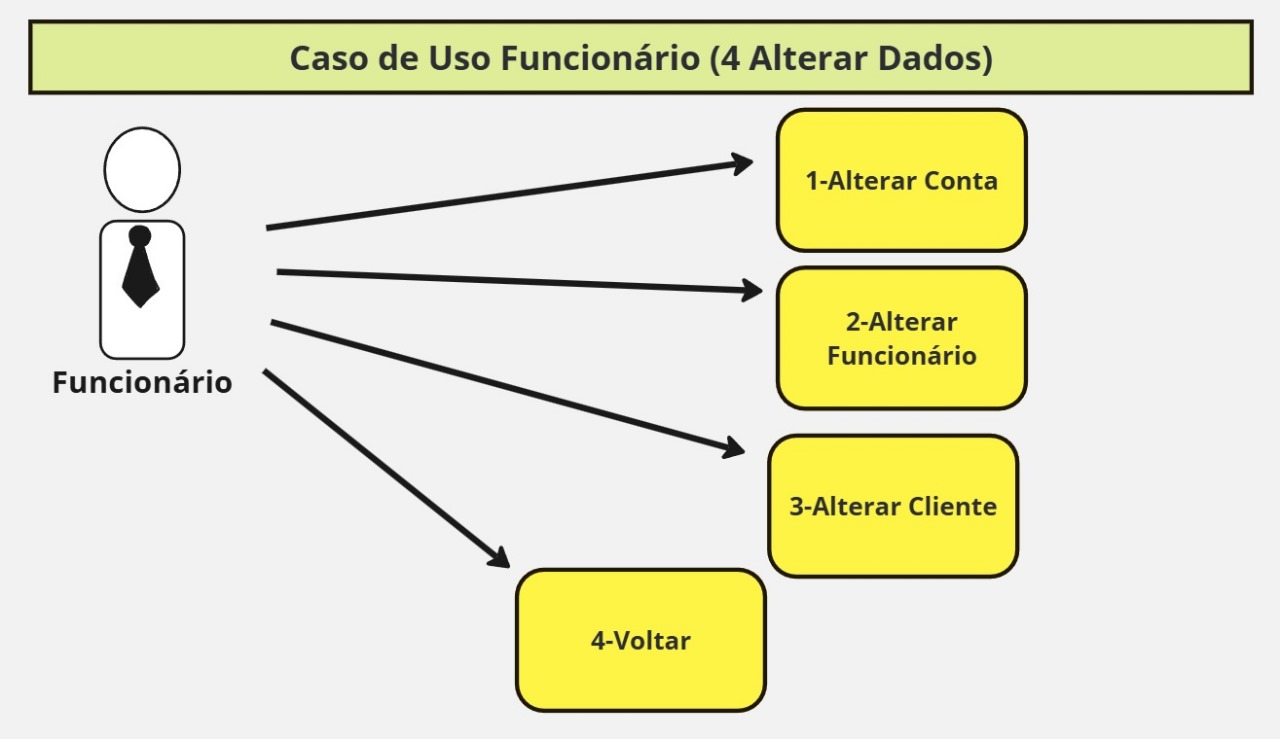


Figura 20: Diagrama de classe - Consultar Dados

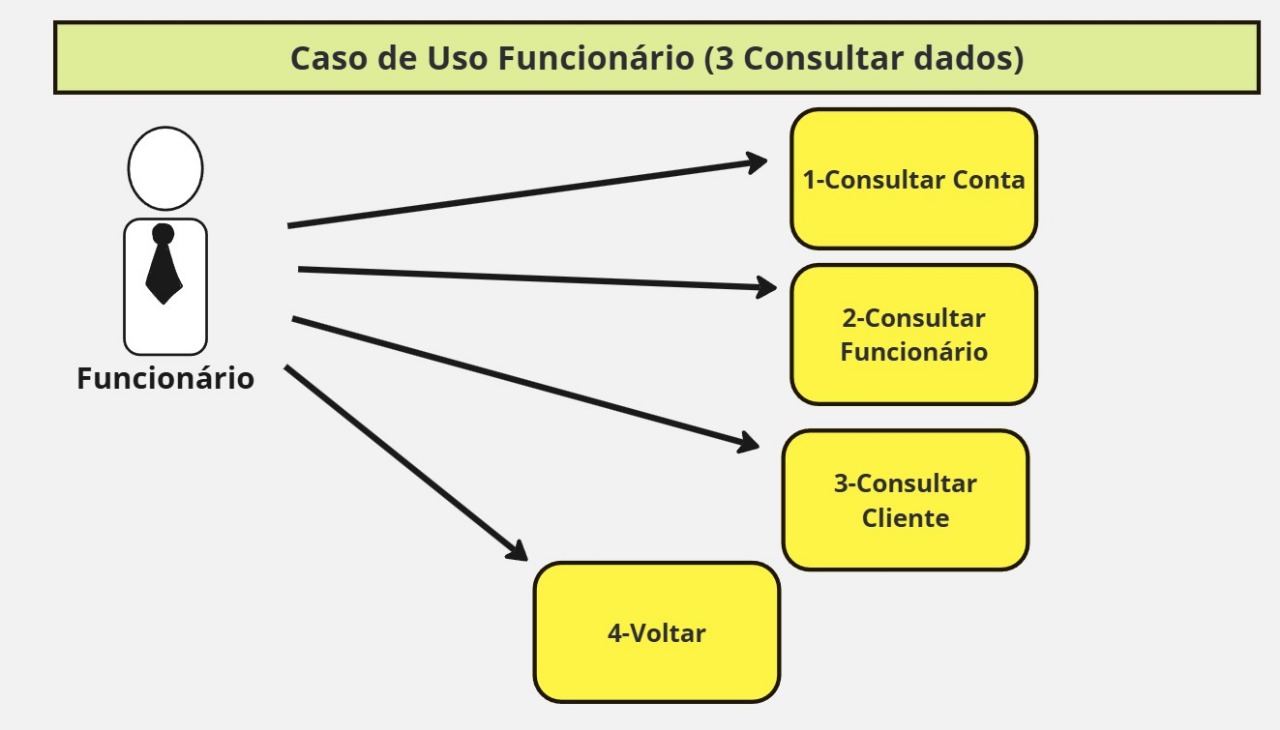


Figura 21: Diagrama de classe - Encerramento de Conta

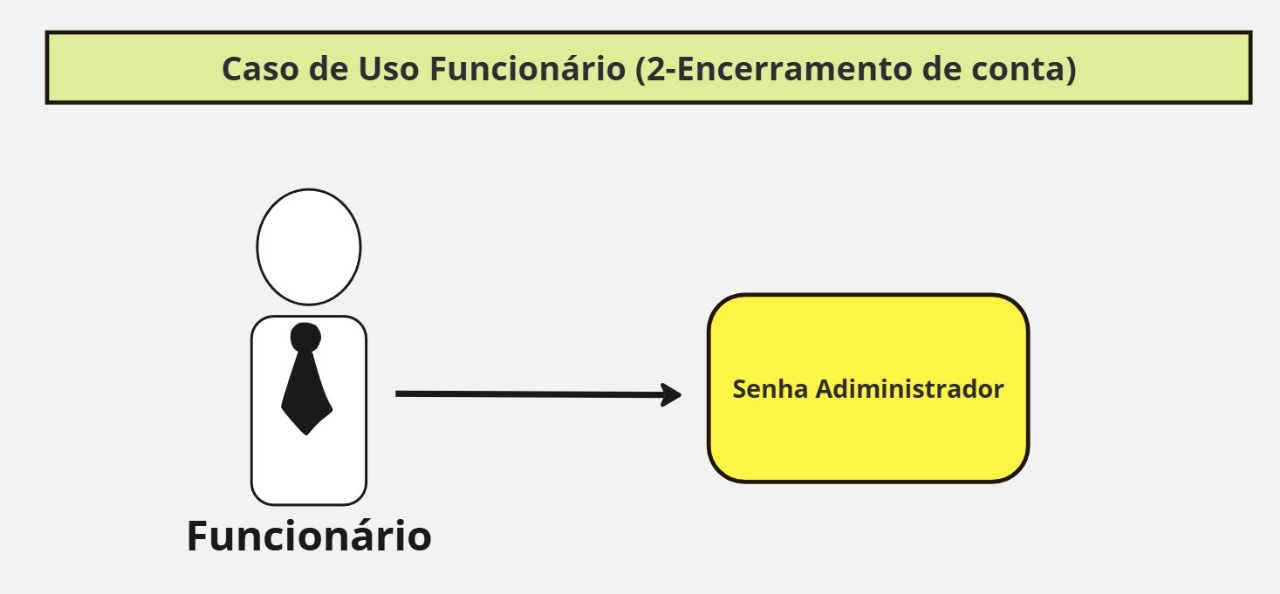


Figura 22: Diagrama de classe - Gerar Relatórios

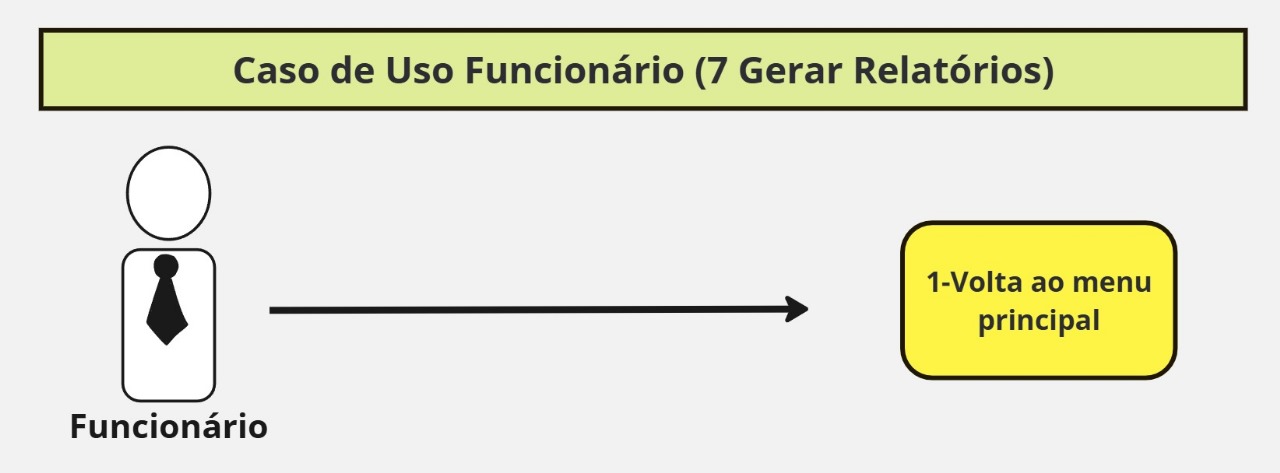


Figura 23: Diagrama de classe - Cadastrar Funcionário

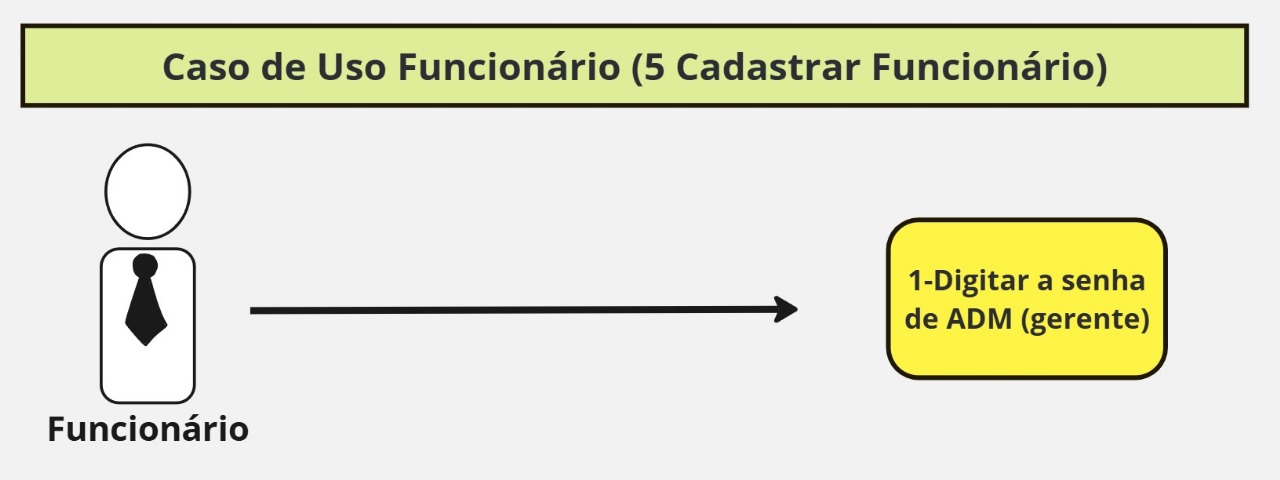


Figura 24: Diagrama de classe - Depósito

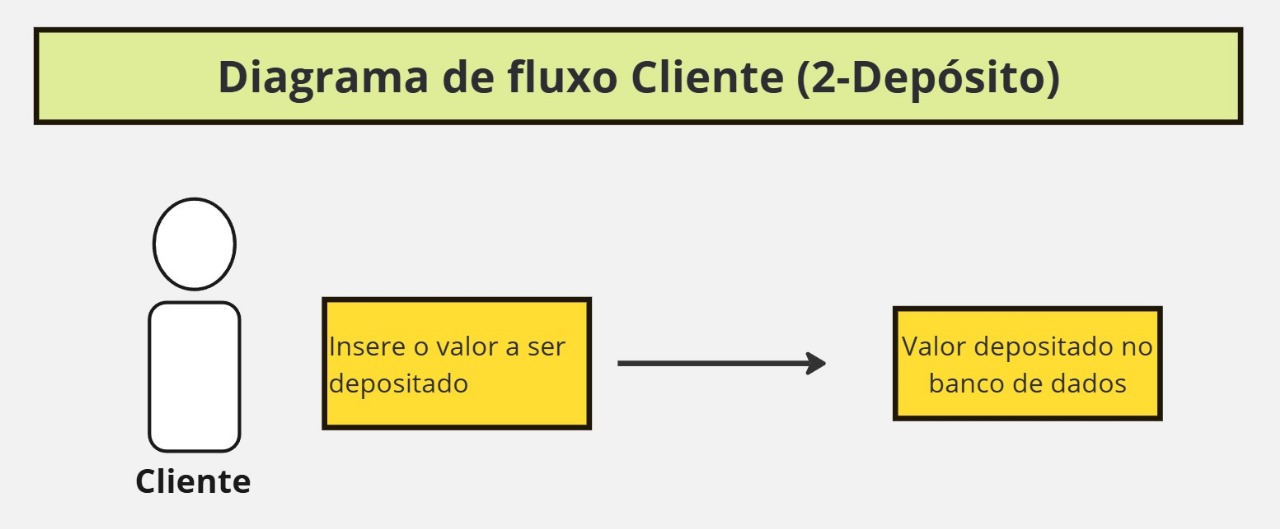


Figura 25: Diagrama de classe - Gerar Relatórios

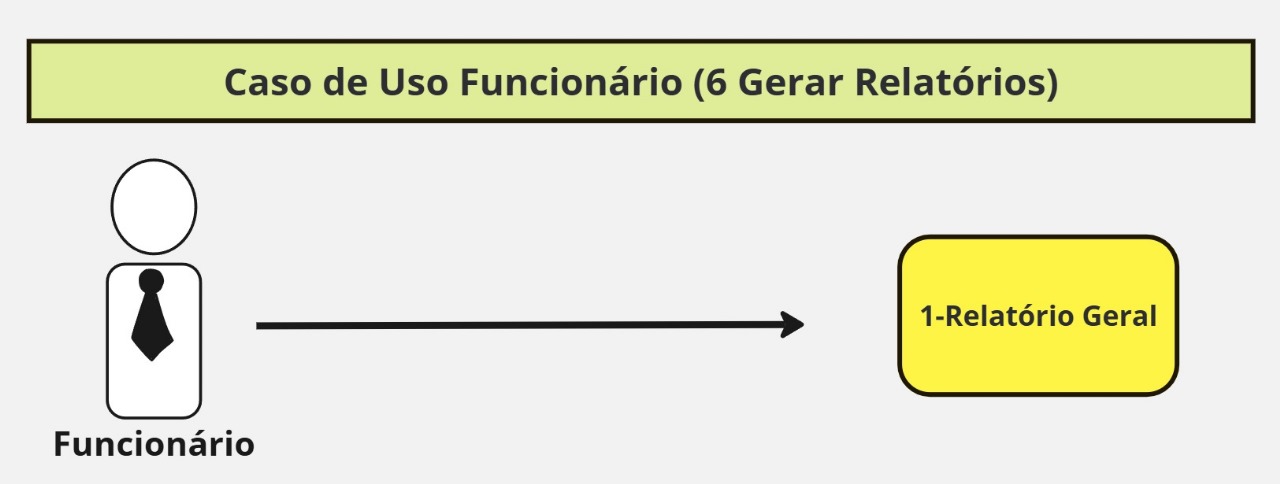


Figura 26: Diagrama de classe - Extrato

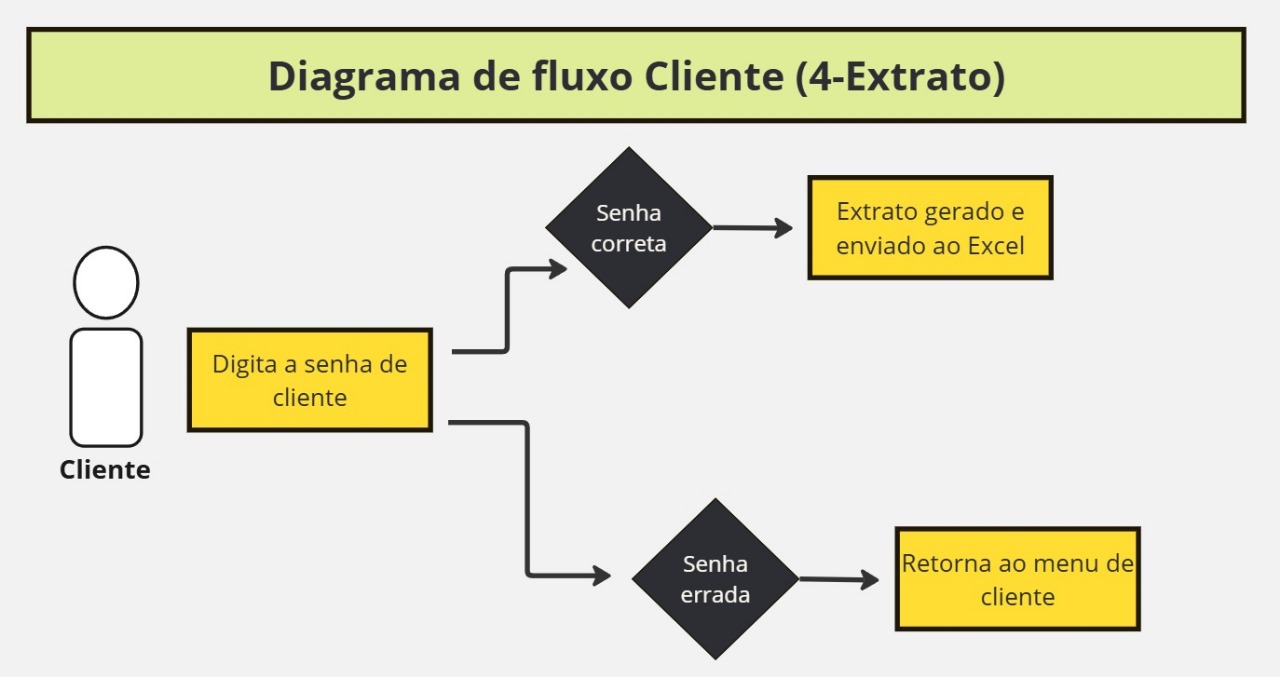


Figura 27: Diagrama de classe - Saque

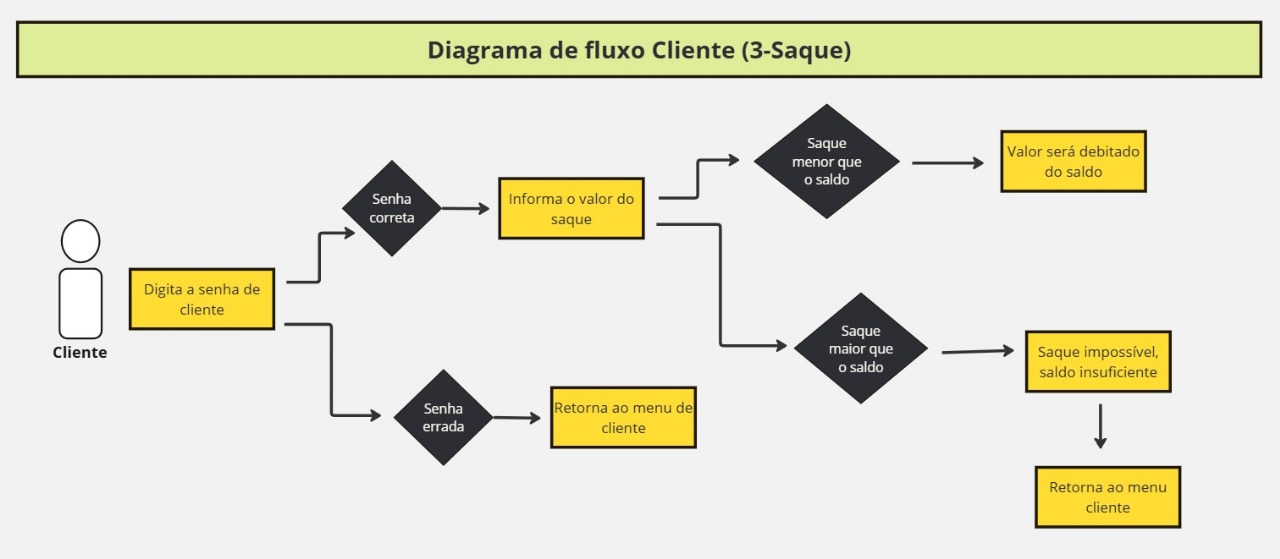


Figura 28: Diagrama de classe - Saldo

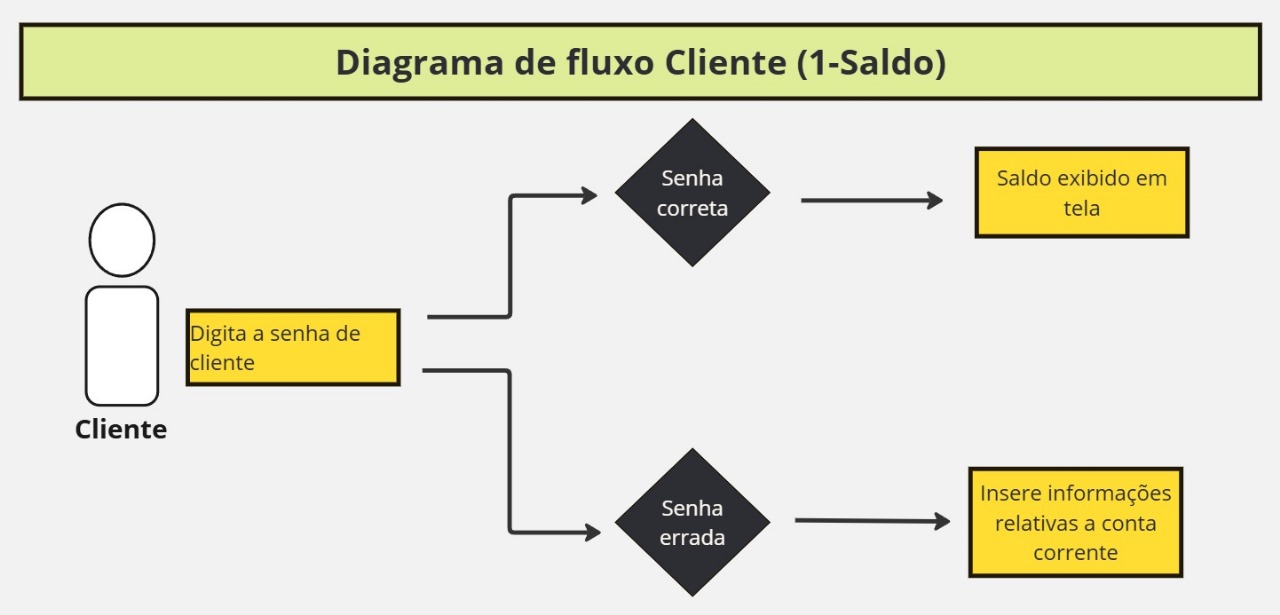


Figura 29: Diagrama de classe - Consultar Limite

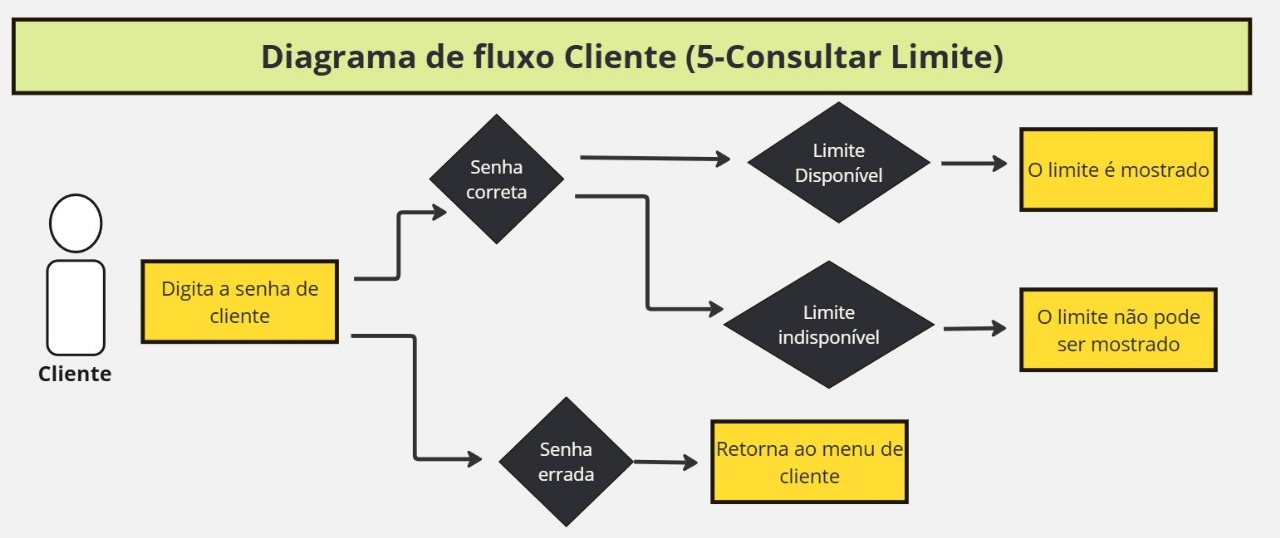


Figura 30: Diagrama de classe - Consultar Dados

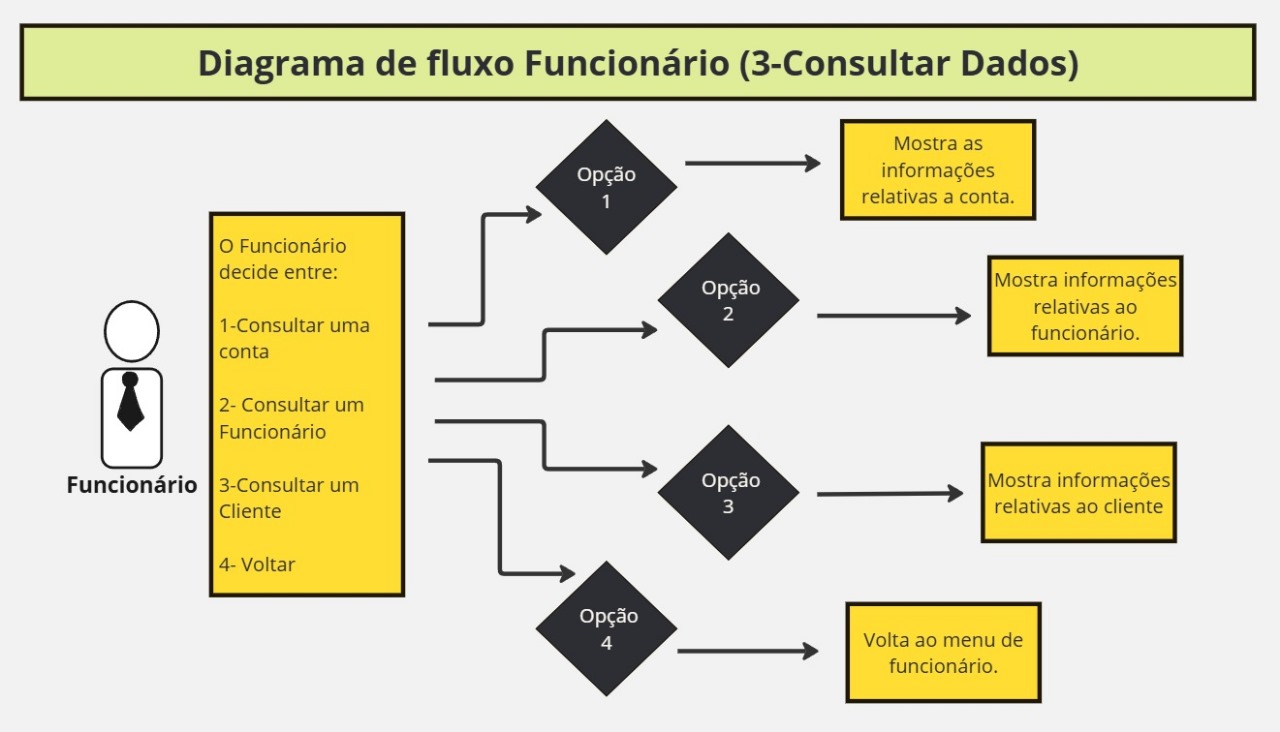


Figura 31: Diagrama de classe - Sair

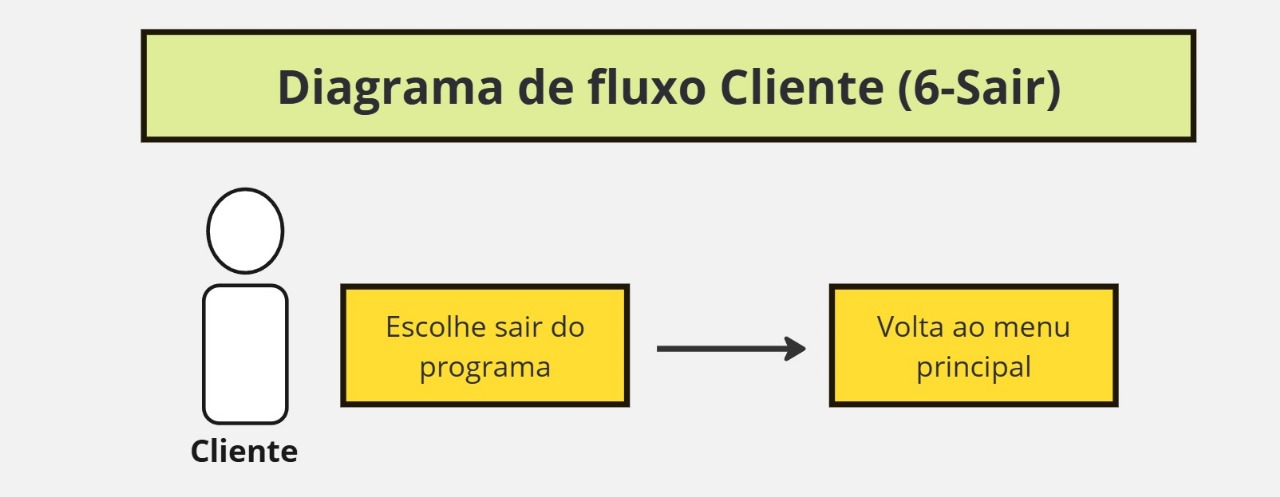


Figura 32: Diagrama de classe - Encerrar Conta

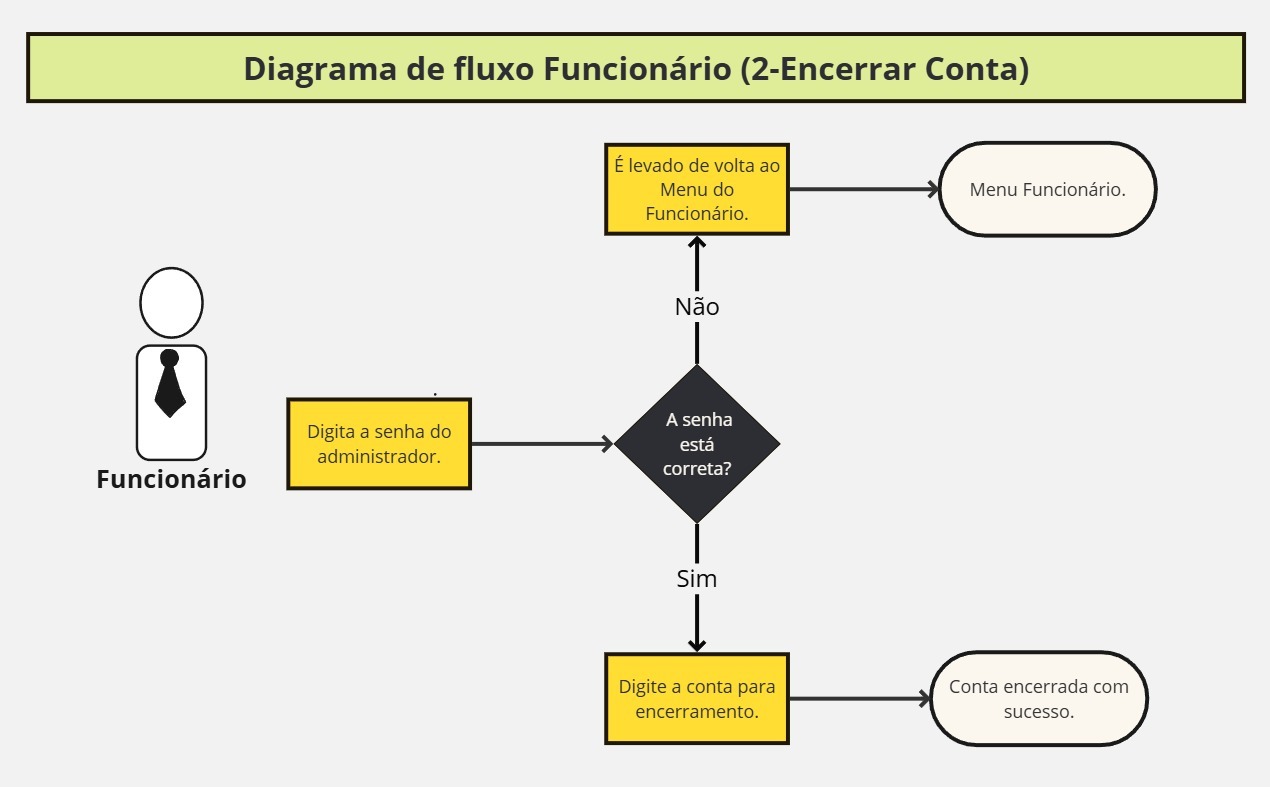


Figura 33: Diagrama de classe - Alterar Dados

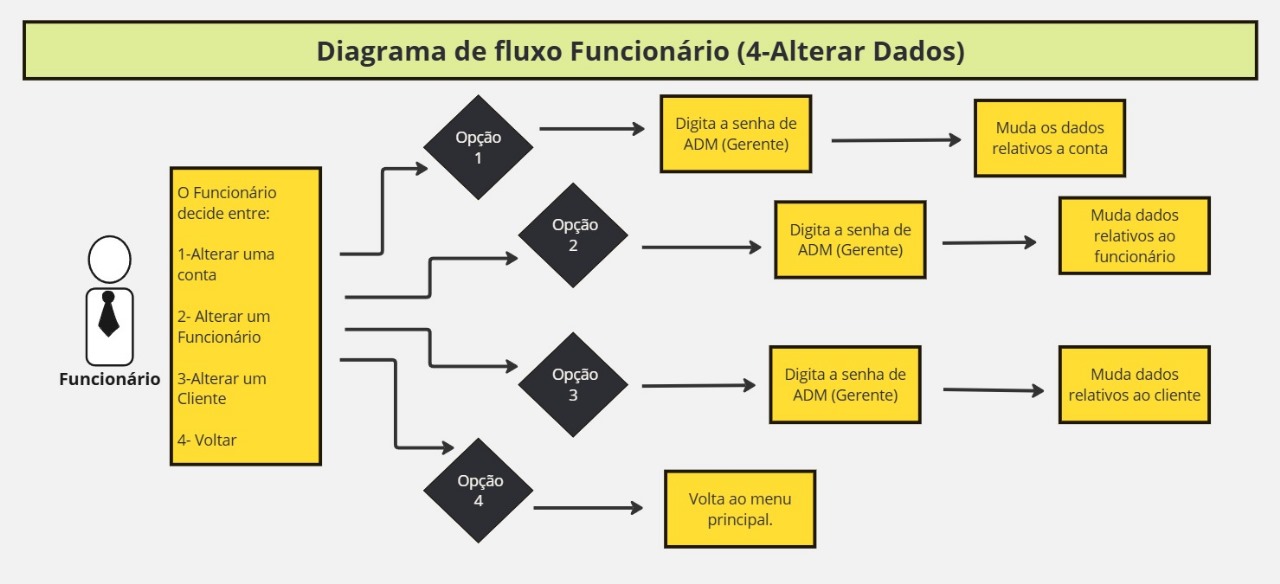


Figura 34: Diagrama de classe - Cadastrar Funcionário

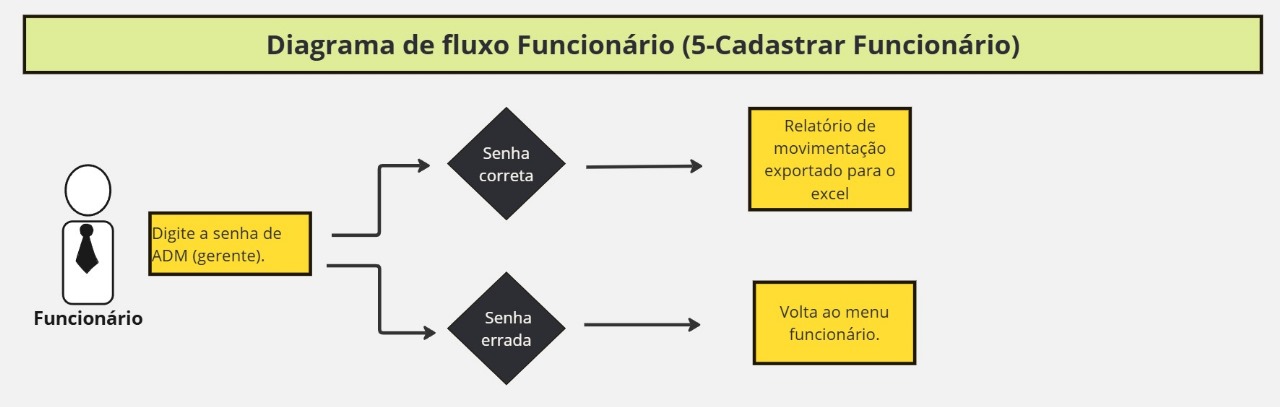


Figura 35: Diagrama de classe - Abertura de Conta

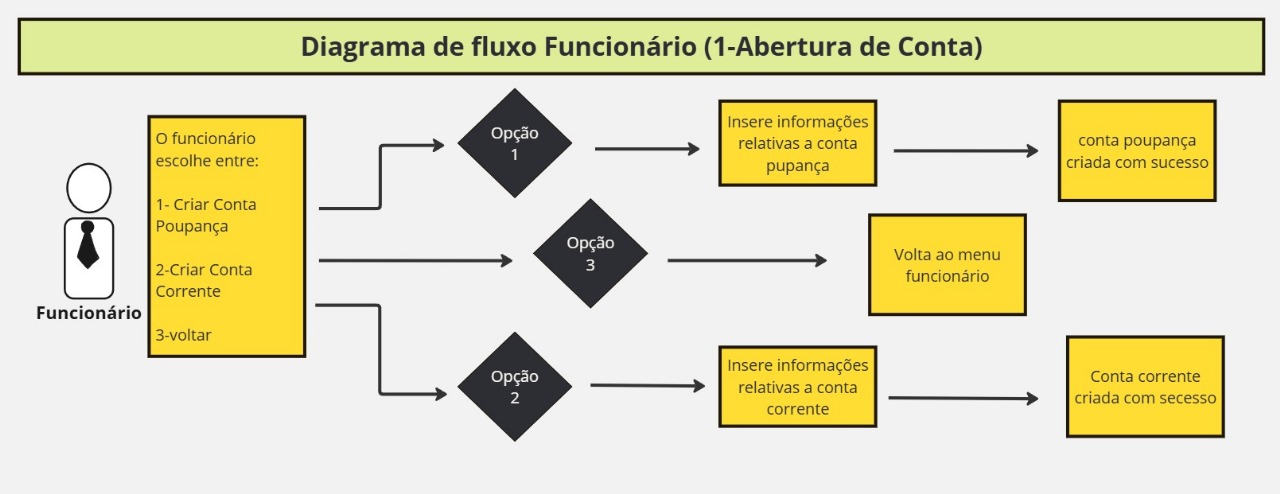


Figura 36: Diagrama de classe - Sair

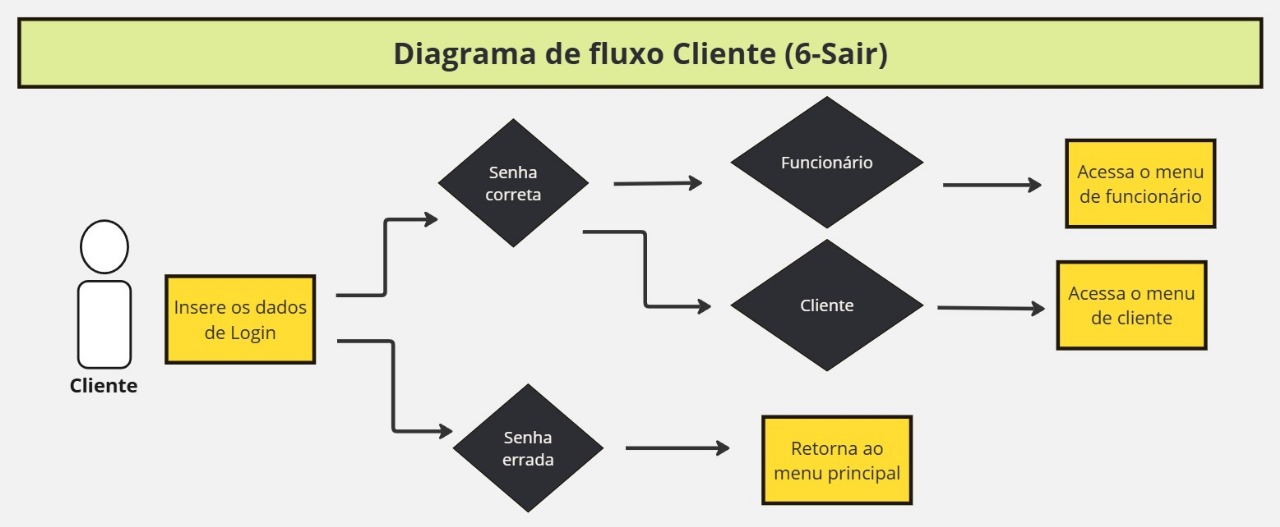
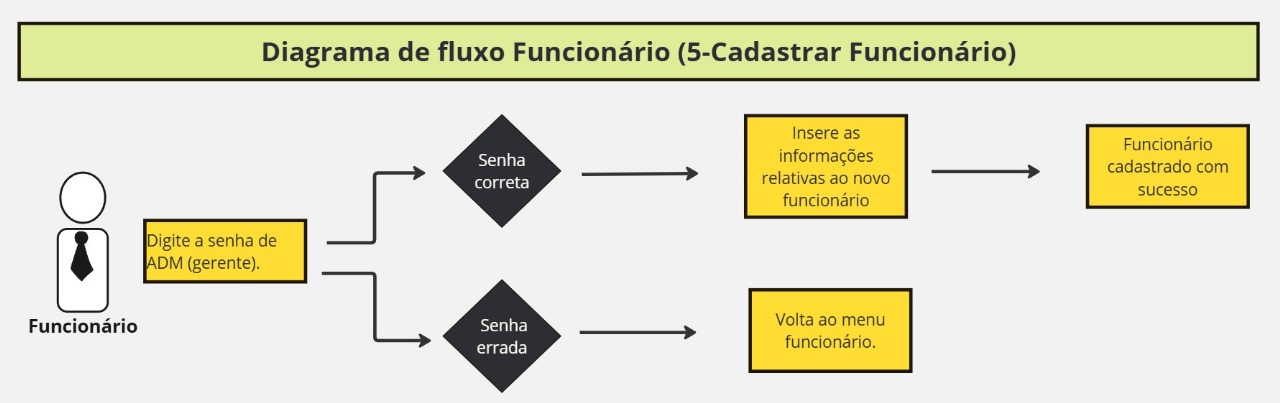
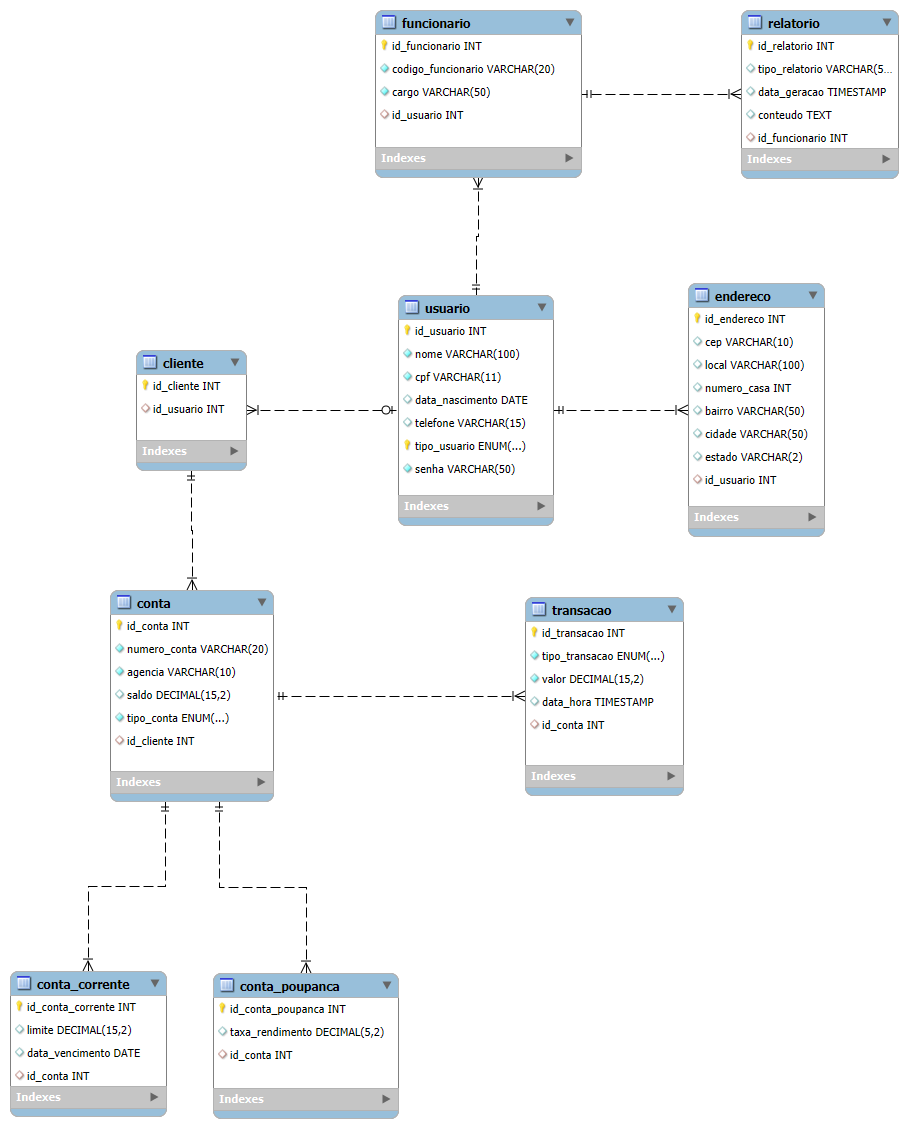


Figura 37: Diagrama de classe - Cadastrar Funcionário



4.6 Modelo lógico

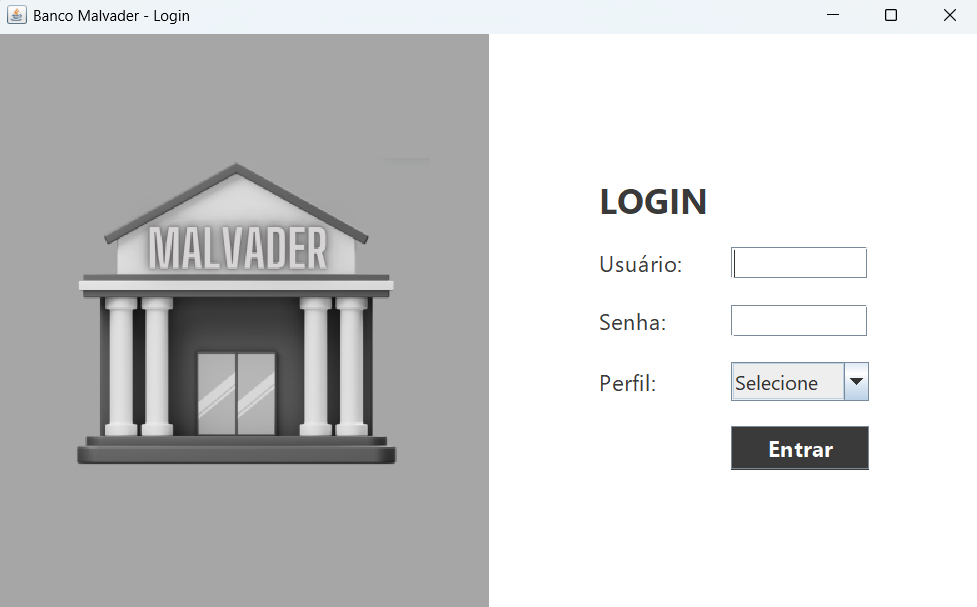
Figura 38: Modelo Lógico



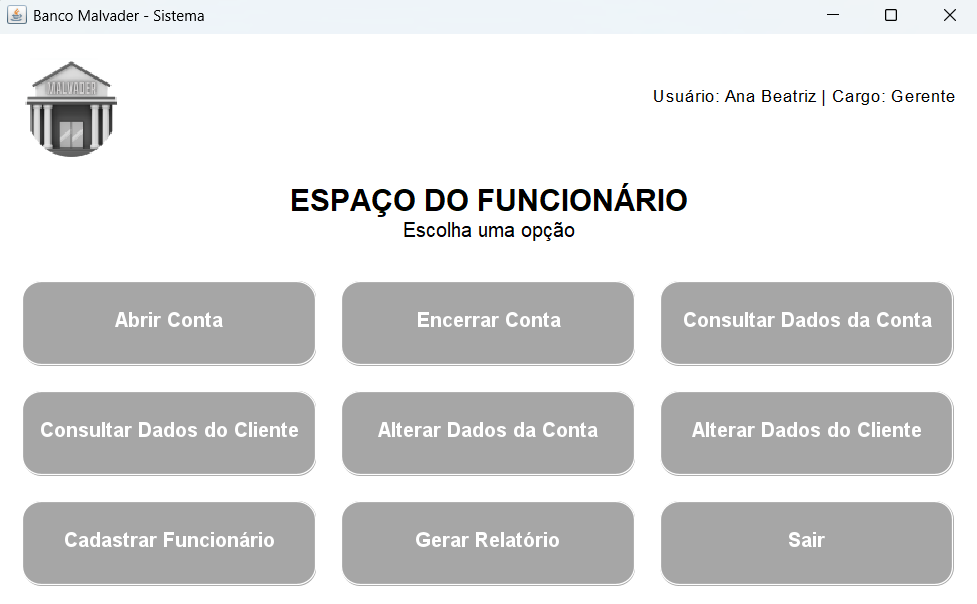
1. **PROJETO DE INTERFACES**

5.1 Tela Login

Figura 39: Tela login



5.2 Tela Menu Funcionário

Figura 40: Menu Funcionário

5.3 Tela de cadastro de clientes

Figura 41: Tela de Cadastro de Cliente

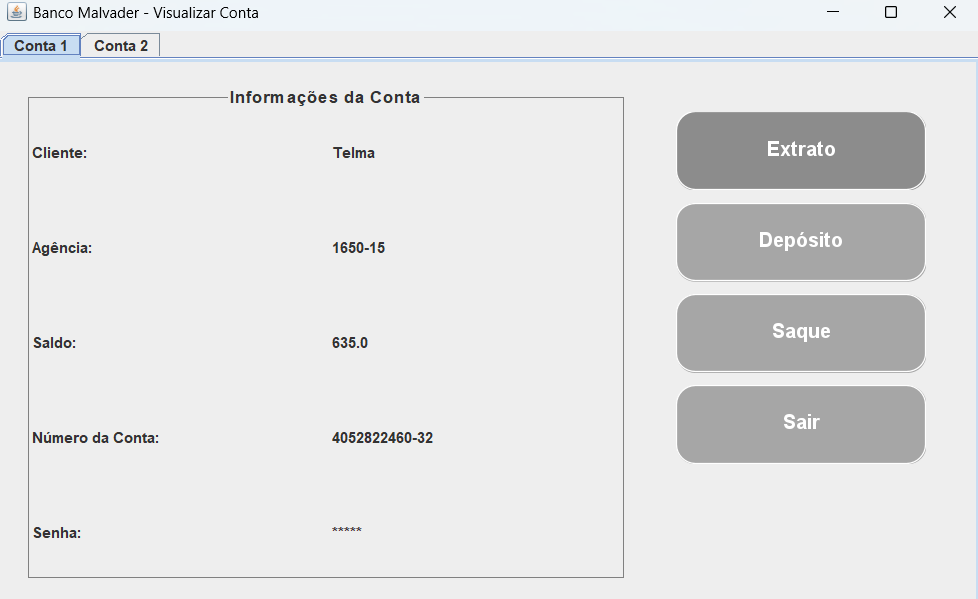
5.4 Tela de visualizar informações da conta

Figura 42: Tela de Informações da Conta

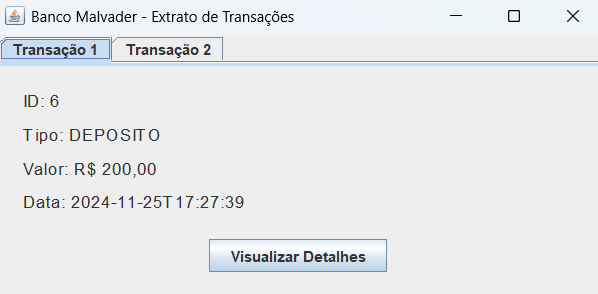
5.5 Tela de cadastro de funcionários

Figura 43: Cadastro de Funcionários

5.6 Tela Menu Cliente

Figura 44: Menu Cliente

5.7 Tela de visualizar extrato

Figura 45: Visualizar extrato

1. **CONCLUSÃO**

O desenvolvimento do Banco Malvader possibilitou a aplicação prática de conceitos técnicos de Engenharia de Software, com ênfase na Programação Orientada a Objetos. O uso da linguagem Java junto com a biblioteca Swing para a interface gráfica e a integração com o banco de dados MySQL proporcionou o um sistema funcional, que atende aos requisitos propostos para a gestão simulada de um banco.

Apresentando uma aplicação que propõe funcionalidades robustas, o projeto cumpriu seu objetivo geral. Em conclusão, o Banco Malvader destaca a importância de metodologias estruturadas no desenvolvimento de sistemas e oferece uma base sólida para implementações mais complexas.

1. **GLOSSÁRIO**

Com o objetivo de facilitar o entendimento aos leitores que consultarem a documentação do projeto, esse glossário reúne os principais termos técnicos e conceitos abordados até o presente ponto.

Tabela 15: Termos e descrição

| **TERMO** | **DESCRIÇÃO** |
| --- | --- |
| Relatório Financeiro | Documento gerado pelo sistema contendo informações detalhadas sobre as movimentações financeiras. |
| Programação Orientada a Objetos (POO) | Paradigma de programação utilizado para modelar o sistema com base em objetos, atributos e métodos. |
| Persistência de Dados | Técnica que garante o armazenamento e recuperação das informações em um banco de dados. |
| MySQL | Sistema de gerenciamento de banco de dados relacional utilizado para armazenar informações do sistema. |
| Regras de Negócio | Conjunto de diretrizes que definem as operações e restrições específicas do funcionamento do sistema bancário. |
| Interface Gráfica | Elementos visuais e interativos do sistema que permitem a interação entre o usuário e a aplicação. |
| Integridade de Dados | Garantia de que as informações armazenadas no banco de dados estão corretas e consistentes. |

1. **REFERÊNCIAS BIBLIOGRÁFICAS**

* MySQL Documentation.*MySQL 8.0 Reference Manual*. Disponível em: [**https://dev.mysql.com/doc/**](https://dev.mysql.com/doc/)
* DEITEL, H. M., DEITEL, P. J., Java: Como Programar. 8ª ed. Bookman, 2010.
* SCHILDT, Herbert. Java para iniciantes. Porto Alegre: Bookman, 2015
* MANZANO, José Augusto N. G.