

**Documentação de um**

**Produto de Software**

**MemóriAlegre: Brincando com a memória**

**Nome dos Alunos:**

|  |  |
| --- | --- |
| **RA** | **Nome** |
| **24.01734-5** | **Cauê de Oliveira Almiron** |
| **24.00630-0** | **Gustavo Seripierri** |
| **23.00427-4** | **Guilherme Ferreira Ambrogini** |
| **24.01178-9** | **Murilo Kaspar** |
| **24.01777-9** | **Henrique Impastaro** |
| **24.01897-0** | **Layla Samya Corrêa** |

**2024**

**ÍNDICE DETALHADO**

[1. Descrição/Resumo do Projeto 3](#_Toc164676634)

[2. Levantamento dos Requisitos do Sistema de Software 3](#_Toc164676635)

[2.1. Extração de Requisitos 3](#_Toc164676636)

[2.2. Análise da Coleta de Requisitos 3](#_Toc164676637)

[3. Especificação dos Requisitos do Sistema de Software 4](#_Toc164676638)

[3.1. Requisitos Funcionais 4](#_Toc164676639)

[3.2. Requisitos Não-Funcionais 4](#_Toc164676640)

[3.3. Especificação dos Casos de Uso 4](#_Toc164676641)

[4. Análise/Projeto 5](#_Toc164676642)

[4.1. Diagrama de Casos de Uso 5](#_Toc164676643)

[4.2. Diagrama de Classes 5](#_Toc164676644)

[4.3. Diagrama de Sequência 5](#_Toc164676645)

[4.4. Modelo de Banco de Dados 5](#_Toc164676646)

[4.5. Diagrama de Atividades (opcional) 5](#_Toc164676647)

[4.6. Diagrama de estados (opcional) 5](#_Toc164676648)

[5. Implementação 5](#_Toc164676649)

[6. Testes 6](#_Toc164676650)

[7. Resultados e Considerações 6](#_Toc164676651)

[8. Registro da Apresentação ao Parceiro 6](#_Toc164676652)

[Referências 7](#_Toc164676653)

[Apêndice I 8](#_Toc164676654)

# Descrição/Resumo do Projeto

|  |
| --- |
| Com uma abordagem centrada no entretenimento educativo, será desenvolvido um jogo de memória, permitindo que os usuários interajam com figuras didáticas. O principal objetivo é unir aprendizado e diversão, incentivando o uso da memória para superar desafios ao longo das fases.  Este jogo contará com diversas fases de dificuldade variada, sendo a primeira delas concebida como a mais acessível. Os jogadores terão a liberdade de ajustar a dificuldade de acordo com suas preferências. A implementação será realizada em Java, com integração ao banco de dados MySQL.  Sob o título "MemóriAlegre: Brincando com a memória", o jogo estará disponível para instalação imediata. A interface principal oferecerá opções de seleção de níveis de dificuldade, bem como abas dedicadas à "Configuração" e ao "Ranking". Durante a jogatina, um cronômetro acompanhará o tempo decorrido, contribuindo para a classificação dos jogadores no ranking. Para isso, será necessário que os jogadores informem seus nomes antes de cada partida. |

# Levantamento dos Requisitos do Sistema de Software

A extração de requisitos é um processo fundamental no desenvolvimento de sistemas, no qual as necessidades dos usuários e as especificações do sistema são identificadas e documentadas de forma detalhada. A seguir, apresentaremos a extração de requisitos do MemóriAlegre.

## Extração de Requisitos

Para extrair os requisitos necessários para o desenvolvimento do jogo educativo, realizam uma entrevista com os principais *stakeholders* envolvidos, incluindo educadores e/ou terapeutas. O objetivo da entrevista foi compreender as necessidades, preferências e restrições relacionadas ao público-alvo do jogo.

Roteiro de entrevista:

**Introdução:**

a. Saudações e agradecimento pela participação na entrevista.

b. Explicação do propósito da entrevista: coletar informações para o desenvolvimento de um jogo educativo.

**Perguntas gerais:**

a. Quais são as faixas etárias dos alunos que utilizarão o jogo?

b. Existem alunos com deficiências? (visual, auditiva, motora, cognitiva etc.)

c. Qual é o nível de habilidade cognitiva dos alunos em relação à memória e capacidade de concentração?

**Preferências e temas:**

a. Há alguma preferência em termos de tema ou imagens que os alunos gostariam de ver no jogo?

**Interação com dispositivos:**

a. Como os alunos interagem com dispositivos eletrônicos? (mouse, teclado, tela sensível ao toque etc.)

**Ambiente de uso:**

a. Qual é o ambiente em que o jogo será usado? (sala de aula, ambiente terapêutico, ambiente doméstico etc.)

b. Existe alguma restrição de tempo para as sessões de jogo?

**Acompanhamento e registro de desempenho:**

a. Como os resultados do desempenho dos alunos devem ser acompanhados e registrados?

**Encerramento:**

a. Agradecimento pela colaboração e pela contribuição para o desenvolvimento do projeto.

## Análise da Coleta de Requisitos

Com base nas respostas obtidas durante a entrevista, os requisitos para o desenvolvimento do jogo educativo são sumarizados da seguinte forma:

**Faixa Etária dos Usuários:** O jogo será desenvolvido para usuários de múltiplas idades, desde crianças até adultos e idosos.

**Perfil dos Usuários:** O público-alvo principal do jogo inclui pessoas com deficiência visual (completa e parcial), autismo, síndrome de *down* e deficiência física.

**Tema e Conteúdo do Jogo:** Os itens no jogo devem ser relacionados ao dia a dia, como animais, frutas, números e formas geométricas.

**Habilidades Cognitivas e Motoras:** Os alunos possuem múltiplas dificuldades cognitivas e motoras, mas todos têm a capacidade de jogar o jogo da memória conforme proposto.

**Plataformas de Acesso:** O software será desenvolvido para computadores, mas também deve ser acessível em tablets disponíveis na escola.

**Ambiente de Uso:** O jogo será utilizado principalmente em sala de aula, sendo voltado para esse ambiente específico.

**Restrições de Tempo:** Não foram mencionadas restrições de tempo para as sessões de jogo.

**Acompanhamento e Registro de Desempenho:** Os resultados do desempenho dos alunos devem ser acompanhados e registrados, com acesso disponível para os professores.

# Especificação dos Requisitos do Sistema de Software

No tópico abaixo, trataremos dos Requisitos Funcionais e não funcionais do sistema, mostrando também o catálogo de atores que atuarão no funcionamento do sistema e por fim, o detalhamento das especificações do Caso de Uso.

## Requisitos Funcionais

|  |  |
| --- | --- |
| Identificador | Descrição |
| RF01 – Editar Configurações | O sistema terá um sistema obrigatório de configurações de tempo e tema de cada partida. |
| RF02 – Jogar Partida | Iniciará uma partida do jogo. |
| RF03 – Ranquear Jogadores | O sistema contará com um ranking onde mostrará os dados de cada partida jogada. |
| RF04 – Armazenar Dados dos Jogadores | O sistema possuirá um banco de dados integrado. |

**Catálogo de Atores:**

|  |  |
| --- | --- |
| **Nome** | **Descrição** |
| Aluno | O aluno será o responsável por jogar e consultar a tabela de pontuação. |
| Professor | O professor será o responsável por inserir o nome de cada aluno que irá jogar e configurar as opções de cada partida, como configuração de tempo. Será possível, consultar a tabela de ranking com a pontuação dos alunos. |
| DBMS | O DBMS, Database Management System, é responsável pelo gerenciamento, segurança e manipulação dos dados, sendo essas informações de login, pontuação dos jogadores e entre outros. |
| Sysadmin | Administrador de Sistemas, refere-se à pessoa ou equipe encarregada de configurar e administrar a infraestrutura do sistema, compreendendo tanto o hardware quanto o software, bem como os sistemas operacionais essenciais para garantir o funcionamento eficiente do sistema. |

## Requisitos Não-Funcionais

|  |  |
| --- | --- |
| Identificador | Descrição |
| RNF01 – Ajuste de volume de áudio | Permitirá aos jogadores alterar o volume do som de acordo com suas preferências. |
| RNF02 – Adaptação de cores para alunos daltônicos | Adaptará as cores do jogo para facilitar a identificação e distinção. |
| RNF03 – Acessibilidade da interface | Garantirá que a interface do jogo seja acessível para todos os jogadores, incluindo aqueles com deficiências visuais ou motoras, com opções de navegação simples e claramente identificáveis. |
| RNF04 – Cronômetro de tempo | Exibirá um cronômetro durante o jogo para que os jogadores possam acompanhar o tempo decorrido desde o início da partida |
| RNF05 – Contagem de pontuação | Registrará e exibirá a pontuação dos jogadores ao longo do jogo |
| RNF06 – Proteção de dados e pontuação | Implementará medidas de segurança para proteger as informações dos jogadores, garantindo que apenas eles acessem e modifiquem suas pontuações |
| RNF07 – Portabilidade multiplataforma | Desenvolverá o jogo de forma que possa ser jogado em diferentes dispositivos, como computadores, tablets e smartphones |
| RNF08 – Facilidade de manutenção | Criará o jogo com um código limpo e organizado, facilitando a identificação e correção de erros, bem como a implementação de atualizações e novos recursos no futuro. |

## Especificação dos Casos de Uso

Casos de Uso para o Professor:

# Configurar Opções:

* Breve descrição: Permite ao professor configurar as opções de jogo, incluindo o nome do aluno, o tempo de jogo e o tema das cartas de memória.
* Fluxo Básico:

1 - O professor acessa a opção de configuração de jogo.

2 - O sistema exibe as opções disponíveis para configuração.

3 - O professor insere o nome do aluno.

* Fluxo Alternativo:

Se o professor desejar sair sem salvar as configurações, ele pode cancelar a operação.

Pré-condições: O professor está autenticado como administrador.

Pós-condições: As opções de jogo são configuradas para o aluno.

Consultar Pontuação:

Breve descrição: Permite ao professor consultar a tabela de pontuações dos alunos.

Fluxo Básico:

O professor acessa a opção de consulta de pontuações.

O sistema exibe a tabela de pontuações de até os 10 alunos com as melhores pontuações com a classificação do tema, tempo e nome.

Fluxo Alternativo:

Se não houver pontuações registradas, o sistema informa que a tabela está vazia.

Pré-condições: O professor está autenticado como administrador e esta na tela de Configurações ou o aluno terminou a partida para acessar a tabela.

Pós-condições: Nenhuma.

Fazer Login:

Breve descrição: Permite ao professor fazer login no sistema.

Fluxo Básico:

O professor acessa a página de login.

O professor insere suas credenciais (nome de usuário e senha).

O sistema verifica as credenciais.

Se as credenciais estiverem corretas, o professor é autenticado e direcionado a tela de Configurações.

Fluxo Alternativo:

Se as credenciais estiverem incorretas, o sistema exibe uma mensagem de erro.

Pré-condições: Ter conexão estável com o Banco de Dados.

Pós-condições: O professor está autenticado no sistema.

Casos de Uso para Aluno:

Iniciar Partida:

Breve descrição: Permite ao aluno iniciar uma partida de jogo da memória.

Fluxo Básico:

O aluno acessa a opção de iniciar partida.

O sistema inicia a contagem regressiva de acordo com o tempo configurado pelo professor.

O aluno joga o jogo da memória.

Quando o tempo acabar ou o aluno terminar a partida, o sistema registra a pontuação.

Fluxo Alternativo:

Se o aluno quiser sair do jogo antes do término, ele pode abandonar a partida.

Pré-condições: O aluno foi registrado no sistema.

Pós-condições: A pontuação, nome, tema e tempo do aluno são registradas.

Caso de Uso para o Banco de Dados:

Administrar Dados:

Breve descrição: Permite ao sistema gerenciar os dados do jogo, incluindo registros de alunos, pontuações e configurações de jogo.

Fluxo Básico:

O banco de dados recebe solicitações do sistema para inserir, atualizar ou consultar dados.

O banco de dados executa as operações solicitadas.

Fluxo Alternativo:

Se houver falhas na comunicação com o banco de dados, o sistema exibe uma mensagem de erro e tenta novamente.

Pré-condições: Ter conexão estável com o Banco de Dados.

Pós-condições: Os dados são inseridos, atualizados ou consultados conforme necessário.

Caso de Uso para o System Admin:

Manter Sistema Operacional:

Breve descrição: Permite ao administrador do sistema realizar operações de manutenção no sistema operacional onde o jogo está hospedado.

Fluxo Básico:

O administrador acessa o sistema operacional do servidor onde o jogo está hospedado.

O administrador realiza as operações de manutenção necessárias, como atualizações de segurança, reinicializações ou ajustes de configuração.

Fluxo Alternativo:

Se ocorrerem problemas durante as operações de manutenção, o administrador reverte as alterações.

Pré-condições: O administrador tem acesso privilegiado ao sistema operacional.

Pós-condições: O sistema operacional é mantido e funcional.

# Análise/Projeto

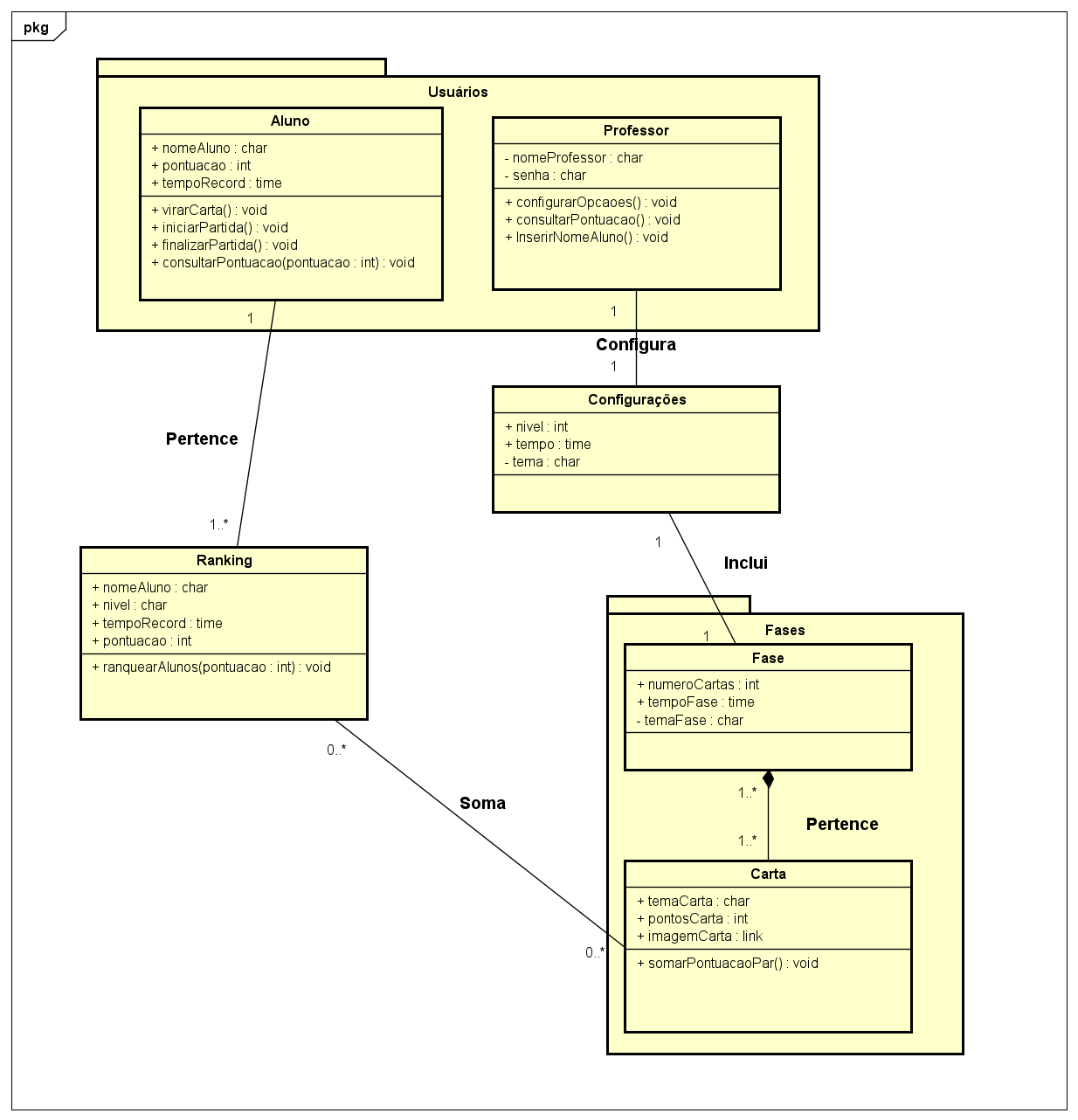
A seguir, será apresentado todos os Diagramas desenvolvidos em torno do projeto MemóriAlegre, sendo eles, Diagrama de Casos de Uso, Classes, Sequência, Atividades e por fim, um esboço do modelo do Banco de Dados.

## Diagrama de Casos de Uso

Diagrama

Descrição gerada automaticamente

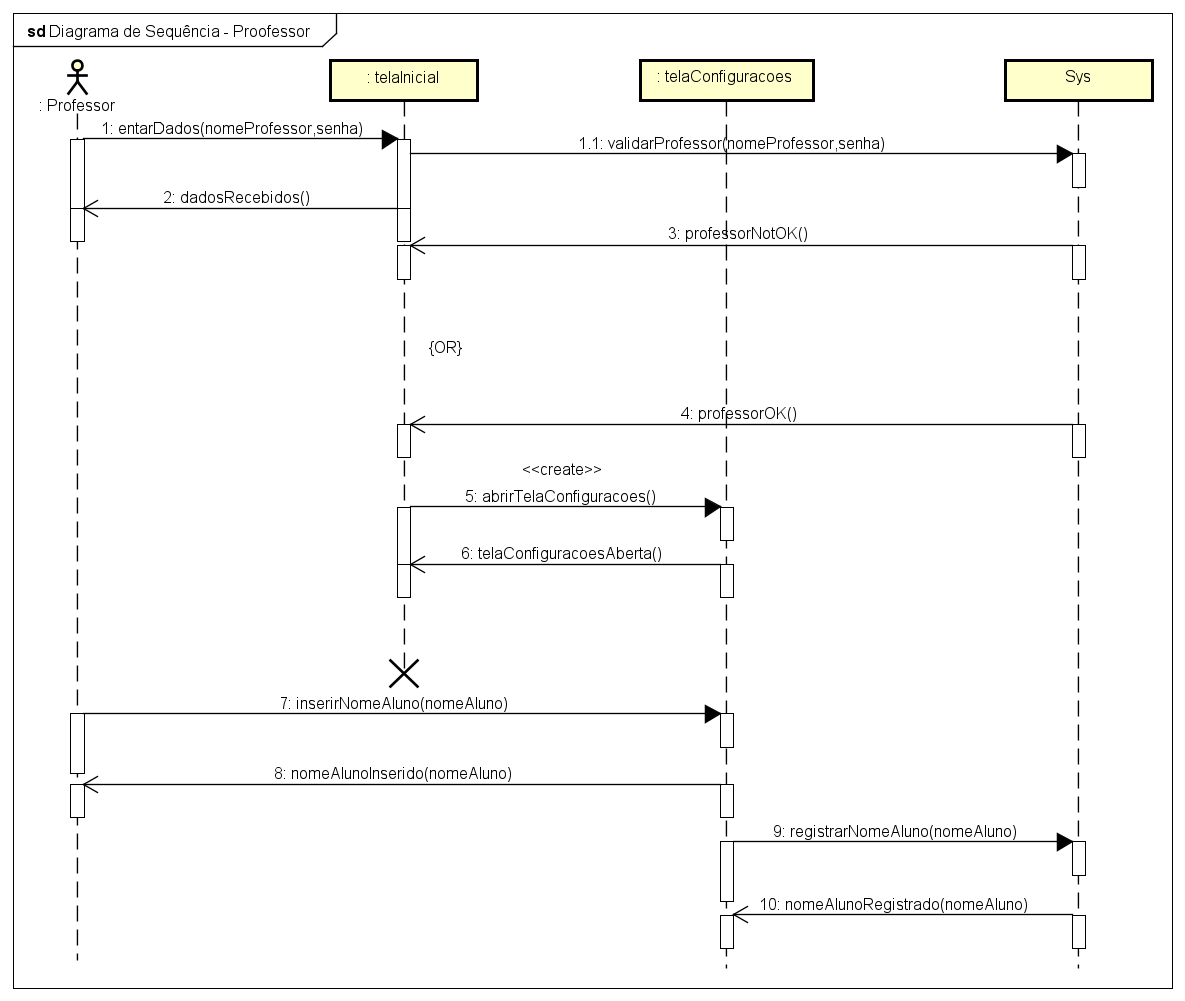
## Diagrama de Classes



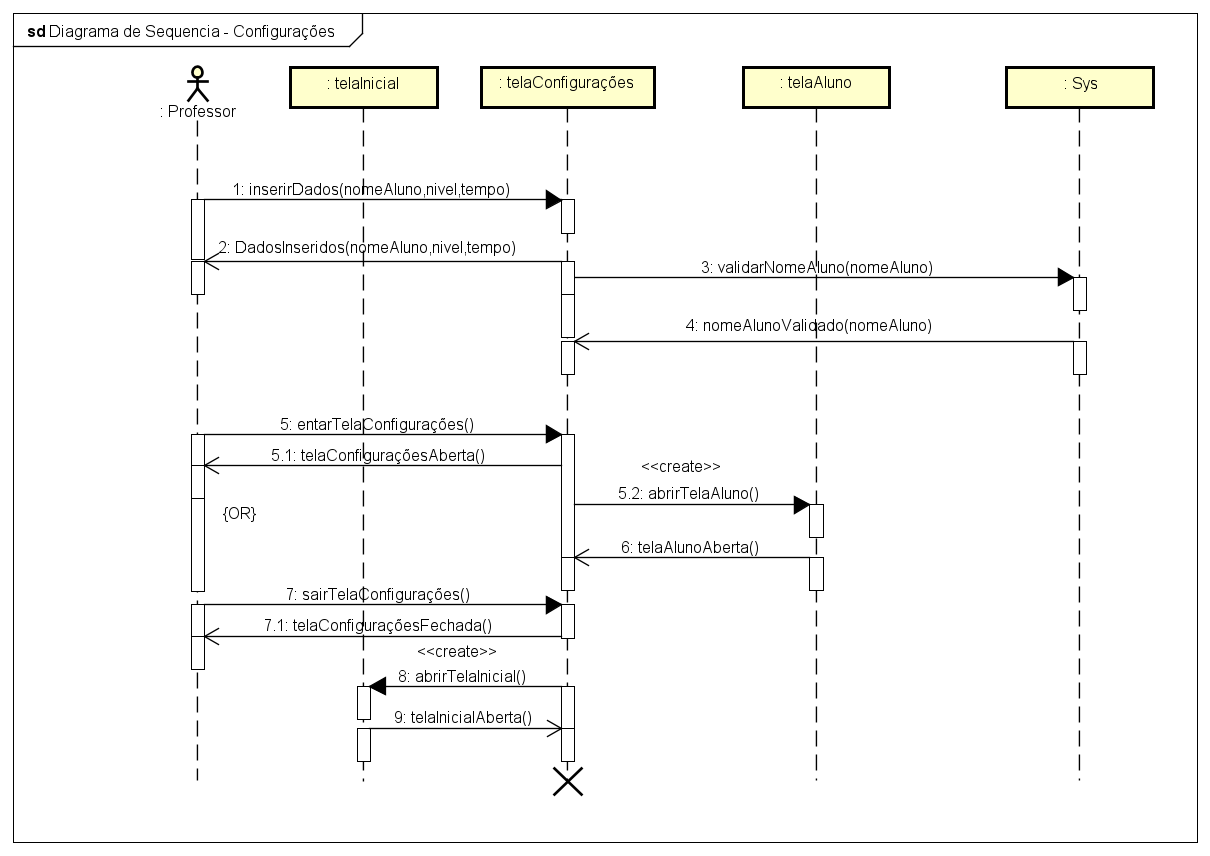
## Diagrama de Sequência

O próximo tópico abordará todos os diagramas de sequência, sendo eles, Diagrama de Sequência de Professor, Configurações, Aluno, Fase, Carta e Ranking.

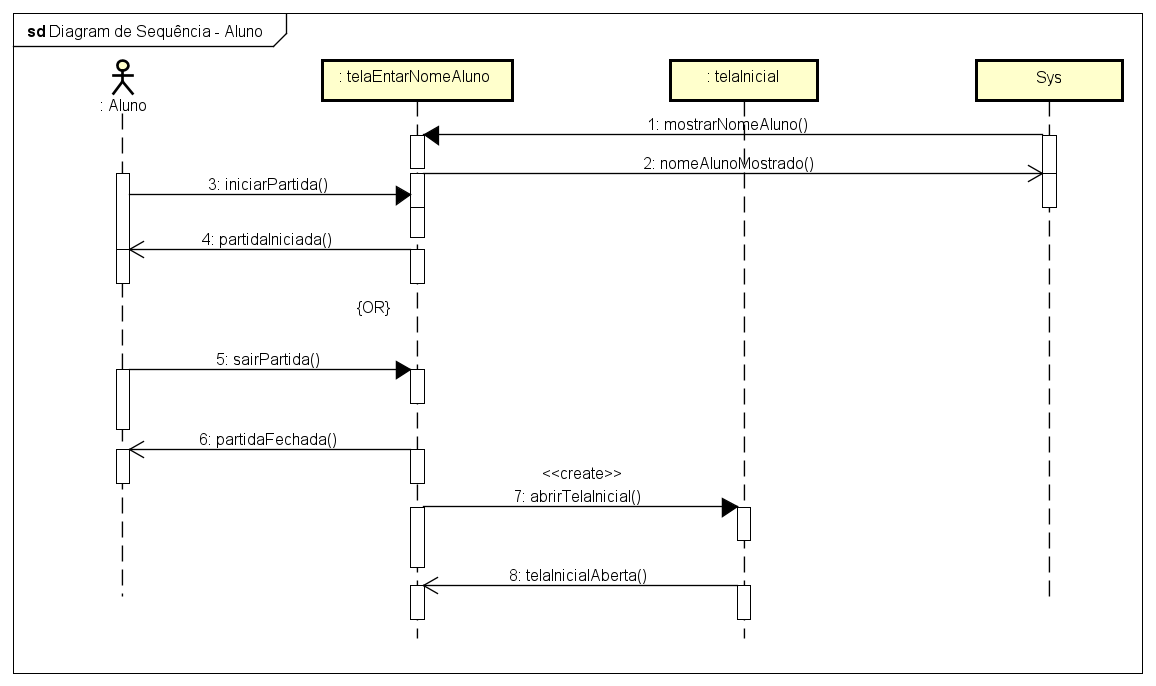
## 2.3.1. Diagrama de Sequência - Professor



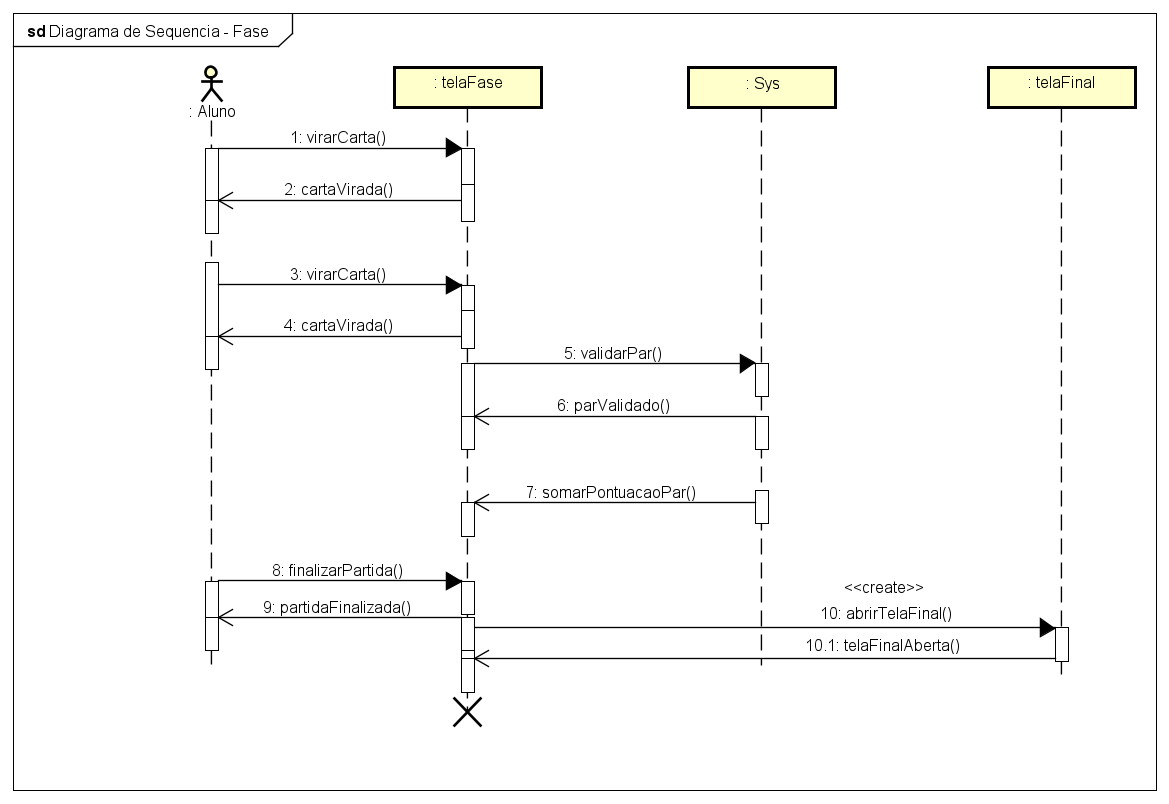
## 2.3.2. Diagrama de Sequência - Configurações



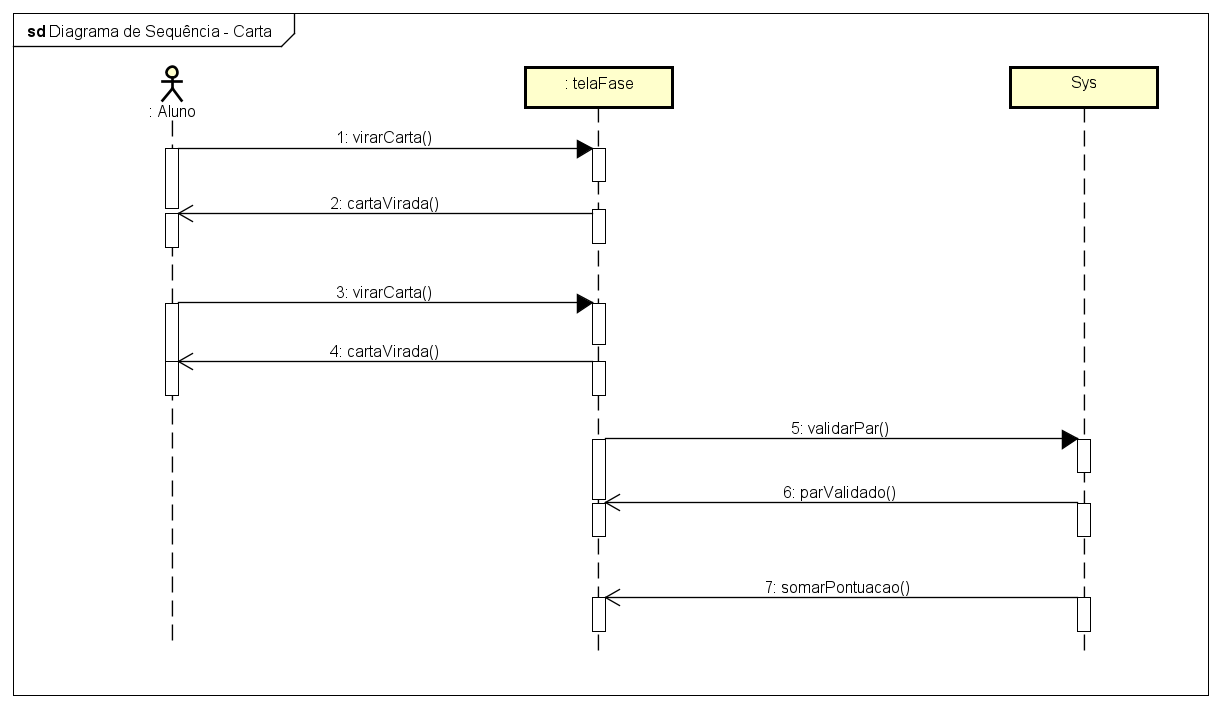
## 2.3.3. Diagrama de Sequência - Aluno



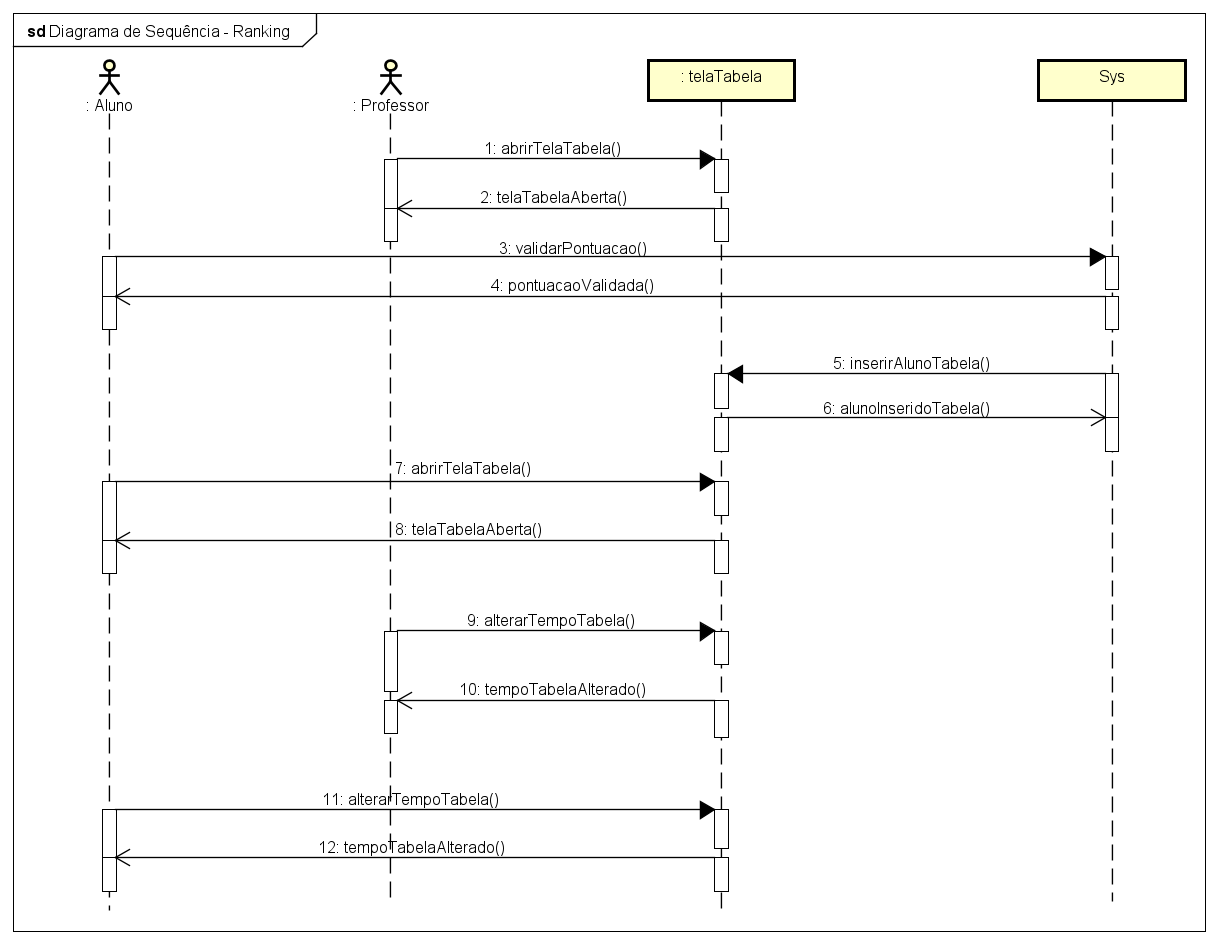
## 2.3.4. Diagrama de Sequência - Fase



## 2.3.5. Diagrama de Sequência - Carta



## 2.3.6. Diagrama de Sequência - Ranking



## Modelo de Banco de Dados

No tópico abaixo, será apresentado o modelo lógico relacional do Banco de Dados como nossa proposta de solução. Junto disso, estará o link do repositório do projeto no Github, onde estará o script físico do Banco de Dados.

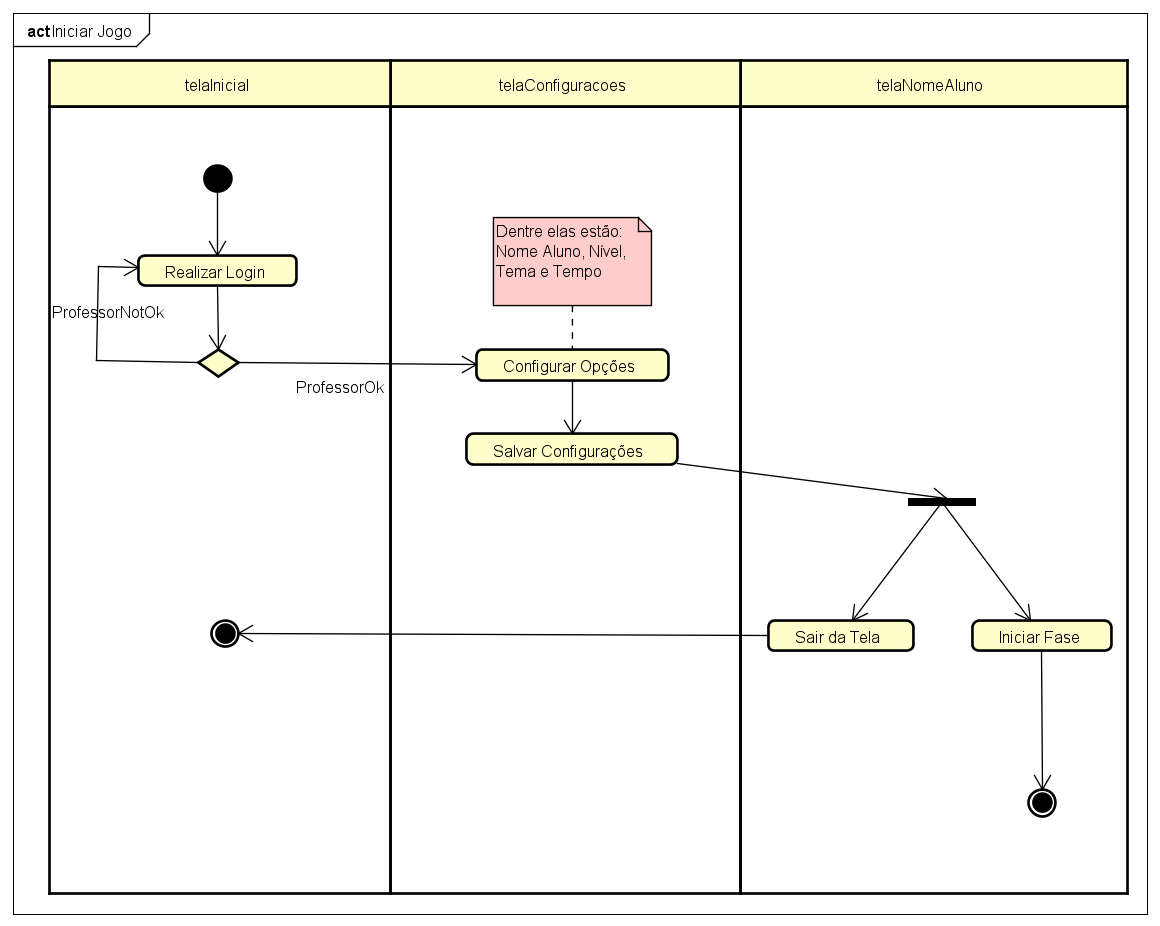
Diagrama

Descrição gerada automaticamente

[Github/memorialegre.sql](https://github.com/cauealmiron/TTI105-Projeto-Integrador-O-Semeador/blob/9c49a36a4bd11a74aef9e2ec92ba7faca1267d84/memorialegre.sql)

## Diagrama de Atividades (opcional)

No tópico a seguir, será apresentado o modelo criado do Diagrama de Atividades. O diagrama de atividades é uma representação meticulosa das tarefas e do fluxo de uma atividade para outra dentro de um sistema. É comumente empregado em métodos que incorporam regras de negócio, fornecendo uma visualização clara e estruturada das operações envolvidas.



## Diagrama de estados (opcional)

O diagrama de estados especifica as sequências de estados pelas quais o objeto pode passar durante seu ciclo de vida em resposta a eventos.

*Disciplina de Apoio: Modelagem Orientada a Objetos.*

# Implementação

Neste item indicar o link de acesso ou repositório de toda implementação de programação orientada a objetos Java e Python (opcional).

*Disciplina de Apoio: Programação Orientada a Objetos, Lógica de Programação e Banco de Dados Relacional.*

# Testes

Neste item indicar o link de acesso ou repositório de todas as evidências de testes unitários realizados no projeto, de acordo, com os casos de uso especificados.

*Disciplina de Apoio: Programação Orientada a Objetos.*

# Resultados e Considerações

Neste item devem ser apresentados os principais “prints” das telas do sistema de software desenvolvido, com uma breve explicação de cada tela e ao final as considerações gerais do projeto, sob o ponto de vista dos requisitos que foram implementados e os resultados obtidos.

# Registro da Apresentação ao Parceiro

Neste item devem ser apresentados os registros firmados com os parceiros do projeto.

Referências

Neste item incluir as referências bibliográficas que forem utilizadas na documentação, seguindo o padrão ABNT.

Apêndice I

Neste item deve ser anexado o roteiro de entrevista ou questionário respondido.

*Disciplina de Apoio: Modelagem Orientada a Objetos.*