

**UDF - CENTRO UNIVERSITÁRIO DO DISTRITO FEDERAL**

**Análise e Desenvolvimento de Sistemas**

**DISCIPLINA – ANÁLISE E PROJETO DE  
SISTEMAS**

**MonitorOn**

**DOCUMENTAÇÃO DE NEGÓCIOS, REQUISITOS E MODELAGEM UML**

**Alunos:**

Samuel Assunção Souza Lisboa - 46134841

Tainara dos Anjos da Silva - 38980126

João Madeiro Gomes Neto - 38818728

**Professor:**

Gabriel de Oliveira Alves

**Brasília – 2025.2**

## **Sumário**

1. INTRODUÇÃO	1
2. JUSTIFICATIVA	1

3. OBJETIVOS	1
3.1. Objetivo Geral	1
3.2. Objetivos Específicos	1
4. DESCRIÇÃO DO SISTEMA PROPOSTO	1
5. REQUISITOS	2
5.1. Requisitos Funcionais	2
5.2. Requisitos Não Funcionais	2
5.3. Regras de Negócio	2
6. DIAGRAMAS UML	3
6.1. Diagrama de Casos de Uso	3
6.1.1. Resumo	3
6.1.2. Desenho/imagem do diagrama	3
6.1.3. Descrição dos Casos de Uso	3
6.2. Diagrama de Classe	4
6.2.1. Resumo	4
6.2.2. Desenho/Imagem do Diagrama	4
6.3. Diagrama de Atividades	5
6.3.1. Resumo	5
6.3.2. Desenho/Imagem do Diagrama	5
6.4. Diagrama de Sequência	5
6.4.1. Resumo	5
6.4.2. Desenho/Imagem do Diagrama	5
6.5. Diagrama de Estado (ou Máquina de Estados)	6
6.5.1. Resumo	6
6.5.2. Desenho/Imagem do Diagrama	6

6.6.	Diagrama de Componentes	6
6.6.1.	Resumo	6
6.6.2.	Desenho/Imagem do Diagrama	7
6.7.	Diagrama de Objetos	7
6.7.1.	Resumo	7
6.7.2.	Desenho/Imagem do Diagrama	7
7.	CONCLUSÃO	7
8.	REFERÊNCIAS	8

## **1. INTRODUÇÃO**

Pensando em aprimorar a experiência acadêmica de alunos e instituições de ensino, desenvolvemos um sistema acadêmico completo, moderno e funcional. A plataforma reúne todas as funcionalidades já conhecidas e essenciais para a rotina universitária: área do aluno, emissão de certificados, acompanhamento de faltas, avisos importantes, envio de atestados online, lançamento de notas, atividades, controle de disciplinas, acesso a documentações e muito mais.

Contudo, o grande diferencial do nosso sistema vai além da gestão tradicional. Implementamos duas funcionalidades inovadoras que transformam a forma como os alunos interagem com o ambiente acadêmico:

Sistema de Ranking Acadêmico: Um ranking com os 30 melhores alunos de cada curso, considerando todas as turmas, com premiações e vagas exclusivas para estágios. Essa funcionalidade visa motivar os estudantes a buscarem um desempenho acima da média, promovendo excelência acadêmica e reconhecimento pelo esforço.

Sistema de Ajuda Colaborativa: Alunos com dificuldades podem solicitar ajuda diretamente pelo sistema, que identifica e conecta o solicitante com os estudantes mais qualificados para prestar apoio, com base em seu desempenho nas disciplinas. Quem colabora recebe horas complementares, criando um ciclo de aprendizagem colaborativa e incentivando a socialização entre os alunos.

Com foco em inovação, engajamento e melhoria contínua do aprendizado, nosso sistema acadêmico não apenas organiza, mas também transforma a vivência educacional em algo mais dinâmico, inclusivo e motivador.

## **2. JUSTIFICATIVA**

Nosso sistema acadêmico foi criado para simplificar a vida de alunos e

instituições, centralizando tudo o que é necessário para o dia a dia escolar: notas, faltas, atividades e documentos. Mas ele vai além: com um ranking de desempenho e premiações, incentivamos os alunos a buscarem sempre o melhor resultado. Além disso, o sistema de ajuda colaborativa conecta alunos com dificuldades aos colegas mais preparados para ajudar, criando um ambiente de aprendizado mais

colaborativo e motivador. Em resumo, nossa plataforma torna a gestão acadêmica mais eficiente e o aprendizado mais dinâmico e engajador.

### **3. OBJETIVOS**

#### **3.1. Objetivo Geral**

Desenvolver uma plataforma acadêmica completa e inovadora que centralize e simplifique a gestão das atividades escolares, proporcionando uma experiência mais eficiente tanto para alunos quanto para instituições de ensino. Além das funcionalidades tradicionais, o sistema busca promover o engajamento, a excelência acadêmica e a colaboração entre os estudantes, por meio de recursos como o ranking de desempenho e o sistema de ajuda entre colegas.

#### **3.2. Objetivos Específicos**

- Identificar e descrever os requisitos funcionais e não funcionais do sistema.
- Elaborar diagramas UML que representam diferentes visões do sistema.
- Produzir a documentação de negócios e de requisitos para apoiar o desenvolvimento.
- Aplicar metodologias de elicitação e análise de requisitos.

### **4. DESCRIÇÃO DO SISTEMA PROPOSTO**

O sistema acadêmico que propomos quer facilitar a vida de alunos, professores e administradores, reunindo todas as atividades acadêmicas em um só lugar. Com uma interface simples e acessível, os estudantes poderão acompanhar notas, faltas e documentos, enquanto os professores gerenciam turmas e lançam avaliações de forma prática. Além das funções básicas, o sistema traz novidades como o ranking acadêmico e a ajuda colaborativa, que incentivam o desempenho e a troca de conhecimento entre os alunos. Totalmente online e acessível de

qualquer dispositivo, a plataforma oferece mais praticidade e dinamismo, tornando a gestão acadêmica mais eficiente e a experiência educacional mais envolvente.

### Público-Alvo

O sistema foi desenvolvido para atender a diferentes tipos de usuários dentro de uma instituição de ensino, incluindo:

1. Alunos: O foco principal do sistema são os estudantes, que poderão acessar todas as informações acadêmicas de maneira fácil e intuitiva, como notas, faltas, atividades, emissão de documentos e participação em iniciativas de colaboração acadêmica.
2. Professores: Os docentes terão acesso a funcionalidades para lançar notas, controlar atividades, acompanhar o desempenho dos alunos e comunicar-se de forma eficiente com eles, além de integrar-se ao sistema de ranking acadêmico e ao sistema de ajuda colaborativa.
3. Administradores (Coordenação e Direção): O sistema fornecerá ferramentas para os administradores gerenciarem o ciclo acadêmico completo, como matrícula de alunos, controle de documentação, geração de relatórios e monitoramento das funcionalidades de ranking e ajuda colaborativa.
4. Instituições de Ensino: O sistema foi desenvolvido para ser integrado a qualquer instituição de ensino, proporcionando uma gestão mais eficiente e moderna, com uma abordagem focada no aprimoramento do processo educacional.

## Principais Funcionalidades

O sistema será composto por diversas funcionalidades que atendem tanto aos requisitos tradicionais da administração acadêmica quanto a novas necessidades de interação e engajamento dos alunos. As principais funcionalidades incluem:

1. Área do Aluno

Acompanhamento de notas e faltas: Os alunos terão acesso às suas notas e faltas



em tempo real, permitindo maior controle sobre seu desempenho.

Emissão de certificados e documentos: O aluno poderá solicitar e emitir certificados acadêmicos, histórico escolar e outras documentações acadêmicas diretamente pela plataforma.

Notificações e avisos: A plataforma enviará notificações importantes sobre datas de provas, atividades, matrícula e outros eventos acadêmicos.

## 2. Ranking Acadêmico

Classificação de alunos: O sistema calcula e exibirá o ranking dos 30 melhores alunos de cada curso, considerando o desempenho geral em todas as disciplinas.

Premiações e vagas para estágios: Os melhores alunos serão premiados com benefícios, como vouchers de estágio e reconhecimento, incentivando a busca pela excelência acadêmica.

## 3. Sistema de Ajuda Colaborativa

Solicitação de ajuda: Alunos com dificuldades em alguma disciplina podem solicitar ajuda diretamente pelo sistema.

Conexão com colegas qualificados: O sistema identifica os alunos com melhor desempenho nas disciplinas e os conectará com os solicitantes de ajuda.

Horas complementares: Os alunos que colaborarem com o auxílio de colegas receberão horas complementares, incentivando a colaboração e o aprendizado mútuo.

## Gestão Acadêmica para Professores e Administradores

Lançamento de notas e atividades: Professores poderão registrar notas, atividades, faltas e outras informações relevantes.

Relatórios detalhados de desempenho: O sistema gerará relatórios sobre o desempenho de alunos e turmas, facilitando o acompanhamento por parte dos professores e coordenadores.

Gestão de documentos acadêmicos: Administradores poderão gerar e fornecer documentos como atestados, certificados, matrículas e outros.

Gestão de Disciplinas e Atividades

Criação e controle de disciplinas: Professores e administradores poderão criar e gerenciar disciplinas, turmas e atividades acadêmicas.

Planejamento acadêmico: O sistema permitirá a organização das atividades acadêmicas, com datas, prazos e detalhes sobre provas, seminários e tarefas.

## Tecnologias a Serem Utilizadas

Para garantir que o sistema seja robusto, seguro e fácil de usar, ele será desenvolvido utilizando tecnologias modernas, escaláveis e de alto desempenho.

Algumas das principais tecnologias que serão utilizadas incluem:

Frontend (Interface do Usuário)

ReactJS ou Angular: Frameworks JavaScript para a construção de interfaces dinâmicas e responsivas, oferecendo uma experiência de usuário fluida.

HTML5/CSS3: Para a construção da estrutura e o estilo da interface.

Tailwind CSS: Framework para design responsivo e personalizável, garantindo que o sistema seja acessível de qualquer dispositivo.

Backend (Servidor e Lógica de Negócio)

Node.js com Express: Ambiente e framework para criar uma API eficiente e escalável, garantindo o desempenho e a estabilidade do sistema.

Python/Django (alternativa): Framework robusto para o desenvolvimento rápido e seguro de aplicativos web, com funcionalidades como autenticação, gerenciamento de banco de dados e geração de relatórios.

## Banco de Dados

PostgreSQL ou MySQL: Bancos de dados relacionais, garantindo integridade e segurança dos dados acadêmicos.

MongoDB (alternativa): Banco de dados NoSQL para armazenar dados não estruturados, como documentos acadêmicos, caso necessário.

## Autenticação e Segurança

JWT (JSON Web Tokens): Para garantir a autenticação e a gestão de sessões de usuários de forma segura.

OAuth2: Para a integração com sistemas externos e garantir que os dados do usuário sejam acessados apenas por aqueles autorizados.

SSL/TLS: Criptografia para garantir a segurança das transações de dados, como senhas e informações sensíveis.

## Infraestrutura e Deploy

Docker: Para containerização e consistência de ambientes entre desenvolvimento, testes e produção.

AWS ou Microsoft Azure: Plataformas de nuvem para hospedagem escalável e de alta disponibilidade.

CI/CD (Continuous Integration/Continuous Deployment): Pipeline automatizado para facilitar testes contínuos e deploy de novas versões do sistema.

## Funcionalidades Adicionais

Firebase: Para envio de notificações push e gestão de autenticação, caso necessário.

Elasticsearch: Para permitir buscas rápidas e eficientes em grandes volumes de dados acadêmicos.

Benefícios Esperados

Para os alunos: O sistema proporcionará uma melhor organização das suas atividades acadêmicas, além de incentivar a excelência através do ranking acadêmico e das recompensas por colaboração.

Para os professores: Maior eficiência no gerenciamento de notas, faltas e relatórios, além de facilitar o acompanhamento de alunos e a interação com eles.

00

Para as instituições de ensino: Uma plataforma moderna e centralizada, que permite uma gestão acadêmica eficiente e que incentiva a melhoria do desempenho e o aprendizado colaborativo entre os alunos.

## **5. REQUISITOS**

### **5.1. Requisitos Funcionais**

RF01 - Cadastro de Alunos: O sistema deve permitir o cadastro e gerenciamento dos dados pessoais e acadêmicos dos alunos.

RF02 - Consulta de Notas: O aluno deve poder visualizar suas notas de todas as disciplinas cursadas.

RF03 - Registro de Faltas: O sistema deve permitir o registro e acompanhamento das faltas dos alunos por disciplina.

RF04 - Emissão de Certificados: O sistema deve emitir certificados acadêmicos oficiais para os alunos aprovados.

RF05 - Envio de Atestados Online: O aluno deve poder enviar atestados médicos diretamente pela plataforma para justificar faltas.



RF06 - Lançamento de Notas: Professores devem poder lançar e atualizar notas das avaliações dos alunos.

RF07 - Controle de Disciplinas: O sistema deve gerenciar a criação, alteração e exclusão das disciplinas oferecidas.

RF08 - Envio de Avisos: Coordenadores devem poder enviar comunicados e avisos para grupos específicos ou toda a instituição.

RF09 - Acesso a Documentações: Alunos devem ter acesso a documentos como histórico escolar, declarações e regulamentos.

RF10 - Sistema de Ranking Acadêmico: O sistema deve calcular e exibir o ranking dos 30 melhores alunos de cada curso.

RF11 - Atualização Automática do Ranking: O ranking deve ser atualizado automaticamente após a inserção de novas notas.

RF12 - Sistema de Ajuda Colaborativa: O sistema deve permitir que alunos solicitem ajuda e que colegas respondam às dúvidas.

RF13 - Cadastro de Professores e Coordenadores: O sistema deve permitir o cadastro e gerenciamento dos dados dos professores e coordenadores.

RF14 - Acompanhamento de Atividades: Alunos devem poder visualizar as atividades e prazos de entrega de cada disciplina.

RF15 - Notificações Automáticas: O sistema deve enviar notificações automáticas para alunos sobre avisos, pendências e atualizações.

## **5.2. Requisitos Não Funcionais**

RNF01 – Desempenho:

O sistema deve responder às solicitações do usuário em até 2 segundos para operações comuns como login, visualização de notas e envio de documentos.

RNF02 – Escalabilidade:

A plataforma deve ser capaz de suportar até 10.000 usuários simultâneos, com possibilidade de escalabilidade horizontal em ambiente de nuvem.

RNF03 – Usabilidade:

O sistema deve possuir uma interface intuitiva e responsiva, acessível via dispositivos móveis e desktops, com design compatível com padrões de acessibilidade (WCAG 2.1 nível AA).

RNF04 – Disponibilidade:

O sistema deve estar disponível 24 horas por dia, 7 dias por semana, com tempo de inatividade máximo de 2 horas mensais para manutenção programada.

#### RNF05 – Segurança:

Todas as informações trocadas entre o cliente e o servidor devem ser criptografadas usando SSL/TLS, garantindo a confidencialidade dos dados.

#### RNF06 – Autenticação:

O sistema deve implementar autenticação segura por meio de JWT (JSON Web Tokens), com validade de sessão ajustável e logout automático após 30 minutos de inatividade.

#### RNF07 – Backup:

Deve ser realizado backup automático diário de todos os dados do sistema, com retenção mínima de 30 dias.

#### RNF08 – Compatibilidade:

O sistema deve ser compatível com os navegadores mais utilizados: Google Chrome, Mozilla Firefox, Microsoft Edge e Safari, nas suas duas últimas versões.

#### RNF09 – Confiabilidade:

O sistema deve manter uma taxa mínima de 99,5% de uptime mensal, garantindo alta confiabilidade operacional.

#### RNF10 – Portabilidade:

O sistema deve ser multiplataforma, podendo ser acessado via navegadores web em dispositivos com sistemas operacionais Windows, macOS, Android e iOS.

#### RNF11 – Auditabilidade:

Todas as ações críticas dos usuários (cadastros, alterações, exclusões, acessos) devem ser registradas em logs de auditoria, com data, hora e identificação do usuário.

#### RNF12 – Internacionalização:

O sistema deve permitir a futura adaptação para múltiplos idiomas, com separação

dos textos da interface do código-fonte (i18n).

**RNF13 – Modularidade:**

A arquitetura do sistema deve ser modular, permitindo a manutenção e evolução de componentes de forma independente (ex: módulo de ranking, módulo de ajuda, etc.).

**RNF14 – Tempo de Recuperação de Falhas (RTO):**

Em caso de falhas, o sistema deve ser restaurado em no máximo 2 horas (Recovery Time Objective).

**RNF15 – Capacidade de Armazenamento:**

O sistema deve suportar, inicialmente, até 2 TB de dados, com possibilidade de expansão conforme crescimento do uso.

### **5.3. Regras de Negócio**

**RF01 – Cadastro de Alunos**

O aluno só pode ser cadastrado se informar CPF válido, e-mail único e não estiver previamente registrado no sistema.

#### RF02 – Consulta de Notas

O aluno pode consultar apenas suas próprias notas. Professores e coordenadores só podem visualizar notas dos alunos das turmas às quais estão vinculados.

#### RF03 – Registro de Faltas

Faltas só podem ser registradas por professores das disciplinas correspondentes e com justificativa obrigatória após o limite de 25% de ausência.

#### RF04 – Emissão de Certificados

A emissão de certificados só será permitida para alunos com status de "aprovado" e situação acadêmica regularizada.

#### RF05 – Envio de Atestados Online

O atestado deve ser enviado em PDF (até 5MB), com nome do aluno, nome e CRM do médico, data e tempo de afastamento. O sistema deve notificar a coordenação automaticamente.

#### RF06 – Lançamento de Notas

Notas podem ser lançadas ou alteradas apenas por professores até a data de fechamento definida pelo calendário acadêmico. Após isso, apenas a coordenação pode autorizar alterações.

#### RF07 – Controle de Disciplinas

Disciplinas não podem ser excluídas se já houver alunos matriculados ou registros de notas vinculados.

#### RF08 – Envio de Avisos

Apenas coordenadores e administradores podem enviar avisos para toda a instituição. Professores podem enviar avisos apenas para suas turmas.

#### RF09 – Acesso a Documentações

Documentos oficiais só estarão disponíveis para alunos regularmente

matriculados. A geração de históricos requer que todas as notas estejam lançadas.

#### RF10 – Sistema de Ranking Acadêmico

O ranking será calculado com base na média ponderada das disciplinas, considerando apenas alunos ativos e com ao menos 75% de frequência.

#### RF11 – Atualização Automática do Ranking

O ranking será recalculado automaticamente após qualquer novo lançamento de nota. Atualizações manuais não serão permitidas.

#### RF12 – Sistema de Ajuda Colaborativa

Apenas alunos com nota igual ou superior a 8,0 em uma disciplina podem se candidatar como ajudantes. As solicitações são aceitas mediante aprovação do sistema.

#### RF13 – Cadastro de Professores e Coordenadores

Professores e coordenadores devem possuir CPF válido, e-mail institucional e não podem ter cadastro duplicado.

#### RF14 – Acompanhamento de Atividades

Atividades entregues fora do prazo só poderão ser aceitas mediante justificativa e aprovação da coordenação.

#### RF15 – Notificações Automáticas

Notificações de prazos, pendências e atualizações devem ser enviadas automaticamente ao menos 1 vez por dia, às 08h.

#### RNF01 – Desempenho

O sistema deve processar operações como login, visualização de notas e envio de atestados em até 2 segundos.

#### RNF02 – Escalabilidade

O sistema deve ser capaz de escalar automaticamente com base na demanda, sem comprometer o tempo de resposta.

#### RNF03 – Usabilidade

O sistema deve seguir boas práticas de UX/UI, com menus acessíveis, botões com feedback visual e suporte a teclado para navegação.

#### RNF04 – Disponibilidade

O sistema deve estar disponível 24/7, com notificação automática em caso de queda e previsão de retorno.

#### RNF05 – Segurança

Todos os dados sensíveis (senha, CPF, notas) devem ser criptografados em trânsito e em repouso.



#### RNF06 – Autenticação

O token de autenticação (JWT) terá validade de 1 hora. Após 30 minutos de inatividade, o usuário será desconectado automaticamente.

#### RNF07 – Backup

Backups serão realizados diariamente às 3h da manhã e armazenados por no mínimo 30 dias em ambiente seguro e criptografado.

#### RNF08 – Compatibilidade

O sistema deve renderizar corretamente nos principais navegadores e em diferentes resoluções de tela.

#### RNF09 – Confiabilidade

O sistema deve garantir um tempo de atividade (uptime) de no mínimo 99,5% por mês, com sistema de monitoramento ativo.

#### RNF10 – Portabilidade

O sistema deve funcionar plenamente em navegadores de dispositivos Android, iOS, Windows e macOS, sem necessidade de instalação de software adicional.

#### RNF11 – Auditabilidade

Toda ação de cadastro, alteração ou exclusão deve ser registrada em log com data, hora e identificador do usuário responsável.

#### RNF12 – Internacionalização

Textos da interface devem estar separados do código e organizados em arquivos de idioma para fácil tradução.

#### RNF13 – Modularidade

Funcionalidades como ranking, avisos e ajuda colaborativa devem ser implementadas como módulos independentes e reaproveitáveis.

#### RNF14 – Tempo de Recuperação de Falhas (RTO)

Em caso de falha, o sistema deve ser restaurado em no máximo 2 horas, com comunicação imediata aos usuários afetados.

#### RNF15 – Capacidade de Armazenamento

O sistema deve exibir alerta aos administradores quando o uso do armazenamento atingir 85% da capacidade.

## **6. DIAGRAMAS UML**

### **6.1. Diagrama de Casos de Uso**

#### **6.1.1. Resumo**

O diagrama de caso de uso é uma representação visual fundamental na fase de

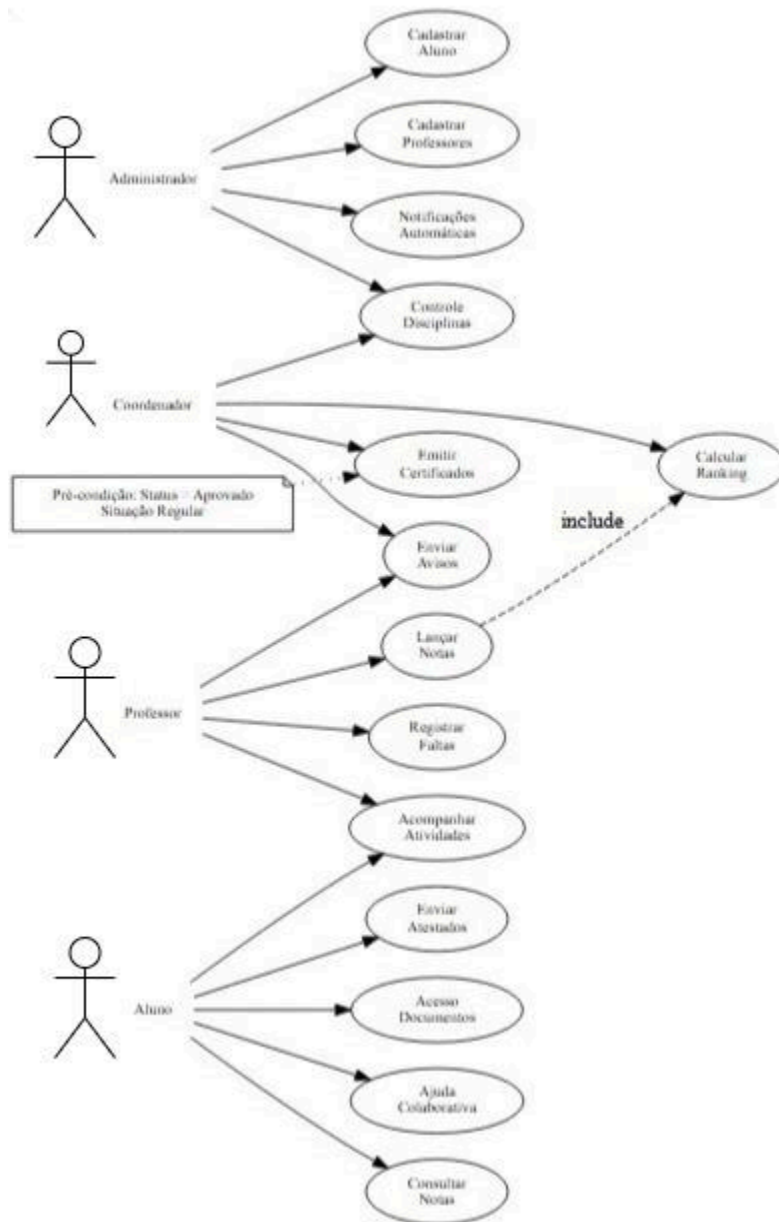
análise de requisitos de um sistema. Ele tem como objetivo descrever as funcionalidades que o sistema deve oferecer, vistas sob a perspectiva dos usuários finais ou de outros sistemas que interagem com ele, chamados de atores. A principal finalidade desse diagrama é comunicar de forma clara e simples o comportamento esperado do sistema, servindo como um elo entre os usuários (ou clientes) e a equipe técnica de desenvolvimento.

Os principais elementos do diagrama são os atores, que representam os usuários ou sistemas externos; os casos de uso, que simbolizam as funcionalidades ou serviços prestados pelo sistema; e os relacionamentos, que mostram como os atores interagem com os casos de uso e como os próprios casos de uso se relacionam entre si (por meio de dependências como include e extend).

Por exemplo, em um sistema acadêmico, casos de uso podem incluir "Cadastrar Aluno", "Lançar Notas", "Consultar Faltas", ou "Gerar Certificado". Os atores seriam "Aluno", "Professor", "Coordenador" e "Administrador". O relacionamento entre "Lançar Notas" e "Atualizar Ranking" pode ser representado como uma relação include, pois o ranking só é atualizado após o lançamento de notas.

No contexto do sistema que o grupo está desenvolvendo, o diagrama de caso de uso pode ser utilizado para modelar e validar os requisitos funcionais, garantindo que todas as interações essenciais (como o uso do ranking acadêmico, o envio de atestados online e a ajuda colaborativa) estejam contempladas. Ele facilita o entendimento entre os stakeholders e serve de base para o detalhamento posterior em diagramas mais técnicos, como diagramas de sequência e de classes.

#### **6.1.2. Desenho/imagem do diagrama**



### 6.1.3. Descrição dos Casos de Uso

#### Caso de Uso 01 — Cadastrar

Ator(es): Administrador

Descrição: Permite o cadastro e gerenciamento dos dados pessoais e acadêmicos dos alunos no sistema.

Pré-condição: O aluno não deve estar previamente registrado; CPF e e-mail devem ser válidos e únicos. Fluxo Principal:

1. Administrador acessa o módulo de cadastro de alunos.

2. Informa os dados obrigatórios (nome, CPF, e-mail, curso, matrícula etc.).
3. O sistema valida o CPF e verifica a duplicidade.
4. O sistema grava os dados no banco e confirma o cadastro. Fluxo Alternativo:
  - Se o CPF ou e-mail já estiver cadastrado, o sistema exibe mensagem de erro. Pós-condição: Aluno é registrado e pode acessar o sistema com suas credenciais.

## Caso de Uso 02 — Consultar

Notas Ator(es): Aluno

Descrição: O aluno visualiza suas notas das disciplinas cursadas. Pré-condição: O aluno deve estar autenticado no sistema.

Fluxo Principal:

1. Aluno acessa o menu “Notas”.
  2. O sistema exibe todas as disciplinas matriculadas e respectivas notas. Fluxo Alternativo:
    - Caso não haja notas lançadas, o sistema informa “Notas ainda não disponíveis”.
- Pós-condição: As notas são exibidas corretamente na tela do aluno.

## Caso de Uso 03 — Registrar

Faltas Ator(es): Professor

Descrição: Permite o registro de faltas dos alunos por disciplina. Pré-condição: O professor deve estar vinculado à disciplina.

Fluxo Principal:

1. O professor seleciona a turma e a aula.
2. Marca as presenças e faltas dos alunos.
3. O sistema grava as informações. Fluxo Alternativo:
  - Caso o aluno ultrapasse 25% de faltas, o sistema exige justificativa obrigatória. Pós-condição: As faltas ficam registradas e disponíveis para consulta.

## Caso de Uso 04 — Emitir

Certificados Ator(es):

Coordenador

Descrição: Gera certificados acadêmicos oficiais para alunos aprovados.

Pré-condição: Aluno deve ter status de “aprovado” e situação acadêmica regularizada.

Fluxo Principal:

1. Coordenador acessa a área de certificados.
2. Seleciona o aluno ou turma.
3. O sistema valida status acadêmico e gera o certificado.
4. O certificado é emitido em formato PDF. Fluxo Alternativo:

– Se o aluno não estiver regular, o sistema bloqueia a emissão.

Pós-condição: Certificado disponível para download e arquivado no sistema.

Caso de Uso 05 — Enviar

Atestados Ator(es): Aluno

Descrição: O aluno envia atestado médico online para justificar faltas.

Pré-condição: O aluno deve estar autenticado e possuir atestado em formato PDF até 5MB. Fluxo Principal:

1. Aluno acessa o módulo de justificativas.
2. Anexa o arquivo PDF e preencha os dados (nome do médico, CRM, tempo de afastamento).
3. O sistema valida o formato e tamanho.
4. O atestado é enviado e a coordenação é notificada. Fluxo Alternativo:

– Caso o arquivo seja inválido, o sistema solicita novo envio.

Pós-condição: O atestado é registrado e enviado à coordenação para análise.

Caso de Uso 06 — Lançar Notas

Ator(es): Professor

Descrição: Permite o lançamento e atualização das notas dos alunos.

Pré-condição: O professor deve estar vinculado à disciplina e dentro do prazo de lançamento. Fluxo Principal:

1. Professor acessa o módulo de notas.
2. Seleciona turma e avaliação.
3. Insere ou atualize as notas.



4. O sistema grava as notas e atualiza automaticamente o ranking. Fluxo Alternativo:
  - Caso o prazo de lançamento tenha expirado, somente a coordenação poderá autorizar alterações.Pós-condição: As notas ficam registradas e o ranking é calculado.

Caso de Uso 07 — Controle de Disciplinas  
Ator(es): Coordenador, Administrador

Descrição: Gerencia a criação, edição e exclusão de disciplinas. Pré-condição: O usuário deve possuir permissão administrativa. Fluxo Principal:

1. Usuário acessa o módulo de disciplinas.
2. Cadastrar ou editar informações da disciplina.
3. O sistema salva as alterações. Fluxo

Alternativo:

- Disciplinas com alunos matriculados não podem ser excluídas. Pós-condição: Disciplinas atualizadas e disponíveis no sistema.

Caso de Uso 08 — Enviar Avisos  
Ator(es): Professor, Coordenador

Descrição: Permite o envio de comunicados a turmas ou a toda a instituição. Pré-condição: O usuário deve estar autenticado.

Fluxo Principal:

1. Usuário cria novo aviso.
2. Define o público-alvo (turma ou geral).
3. Envia o aviso.
4. O sistema notifica os destinatários. Fluxo

Alternativo:

- Professores podem enviar avisos apenas para suas turmas. Pós-condição: Avisos entregues aos usuários selecionados.

## Caso de Uso 09 — Acesso a

Documentos Ator(es): Aluno

Descrição: O aluno acessa documentos acadêmicos, como histórico escolar e declarações. Pré-condição: O aluno deve estar regularmente matriculado.

Fluxo Principal:

1. Aluno acessa o menu “Documentos”.
2. Seleciona o tipo desejado (histórico, declaração etc.).
3. O sistema verifica a elegibilidade e gera o

documento. Fluxo Alternativo:

- Caso existam notas pendentes, o histórico não é gerado.

Pós-condição: Documento disponível para visualização ou download.

## Caso de Uso 10 — Calcular Ranking

Ator(es): Coordenador (automático após lançamento de notas)

Descrição: Calcula e exibe o ranking dos 30 melhores alunos de cada curso. Pré-condição: Notas devem ser lançadas e alunos ativos.

Fluxo Principal:

1. Sistema coleta notas e frequências.
2. Calcula a média ponderada.
3. Atualiza o ranking automaticamente. Fluxo

Alternativo:

- Em caso de erro de nota, o ranking é calculado após correção. Pós-condição: Ranking atualizado e visível para os alunos.

Caso de Uso 11 — Ajuda  
Colaborativa Ator(es): Aluno

Descrição: Permite que alunos solicitem ajuda e colegas respondam dúvidas. Pré-condição: O aluno deve estar autenticado.

Fluxo Principal:

1. Aluno acessa a seção de ajuda.
2. Envia uma dúvida ou se candidata como ajudante.
3. O sistema exibe as respostas e interações.

Fluxo Alternativo:

– Apenas alunos com nota  $\geq 8,0$  podem se registrar como ajudantes.

Pós-condição: Dúvida registrada e visível na comunidade de ajuda.

## Caso de Uso 12 — Cadastrar

Professores Ator(es): Administrador

Descrição: Permite o cadastro de professores e coordenadores.

Pré-condição: CPF e e-mail institucional devem ser válidos e únicos. Fluxo Principal:

1. Administrador acessa o módulo de docentes.
2. Preenche os dados necessários.
3. O sistema valida e registra o cadastro.

Fluxo Alternativo:

– Se houver duplicidade, o sistema exibe mensagem de erro.

Pós-condição: Professor/coordenador registrado no sistema.

## Caso de Uso 13 — Acompanhar

Atividades Ator(es): Aluno, Professor

Descrição: Permite visualizar atividades, prazos e status de entrega. Pré-condição: O usuário deve estar vinculado à disciplina.

Fluxo Principal:

1. Usuário acessa o menu de atividades.
2. O sistema exibe todas as atividades com

prazos e status. Fluxo Alternativo:

– Entregas fora do prazo requerem justificativa e aprovação da coordenação. Pós-condição: Atividades e prazos exibidos corretamente.

#### Caso de Uso 14 — Notificações

Automáticas Ator(es): Sistema (processo automatizado)

Descrição: Envia notificações sobre avisos, pendências e atualizações.

Pré-condição: Eventos geradores de notificações devem existir (novo aviso, atividade, nota etc.). Fluxo Principal:

1. Sistema verifica eventos pendentes.
2. Gera notificações automáticas.
3. Envia aos destinatários diariamente às 8h. Fluxo

Alternativo:

– Caso o envio falhe, o sistema reprograma a tentativa.

Pós-condição: Usuários notificados sobre atualizações relevantes.

## 6.2. Diagrama de Classe

### 6.2.1. Resumo

O Diagrama de Classes é a estrutura central da Modelagem Orientada a Objetos (MOO) e um dos diagramas mais importantes da Linguagem de Modelagem Unificada (UML) [1]. Sua finalidade primária é fornecer uma representação visual e estática da estrutura de um sistema, descrevendo as classes, seus atributos, métodos e os relacionamentos entre elas. Ele serve como um blueprint para a implementação do software, facilitando a comunicação entre desenvolvedores e stakeholders [2].

Os elementos principais de um diagrama de classes são:

1. **Classes:** Representadas por retângulos divididos em três seções: nome, atributos (variáveis que armazenam o estado do objeto) e métodos (funções que definem o comportamento do objeto).
2. **Atributos:** Detalham as características da classe (ex: +nome: string na classe Usuario).
3. **Métodos:** Descrevem as operações que a classe pode realizar (ex: +visualizarNotas() na classe Aluno).

Os relacionamentos definem como as classes interagem:

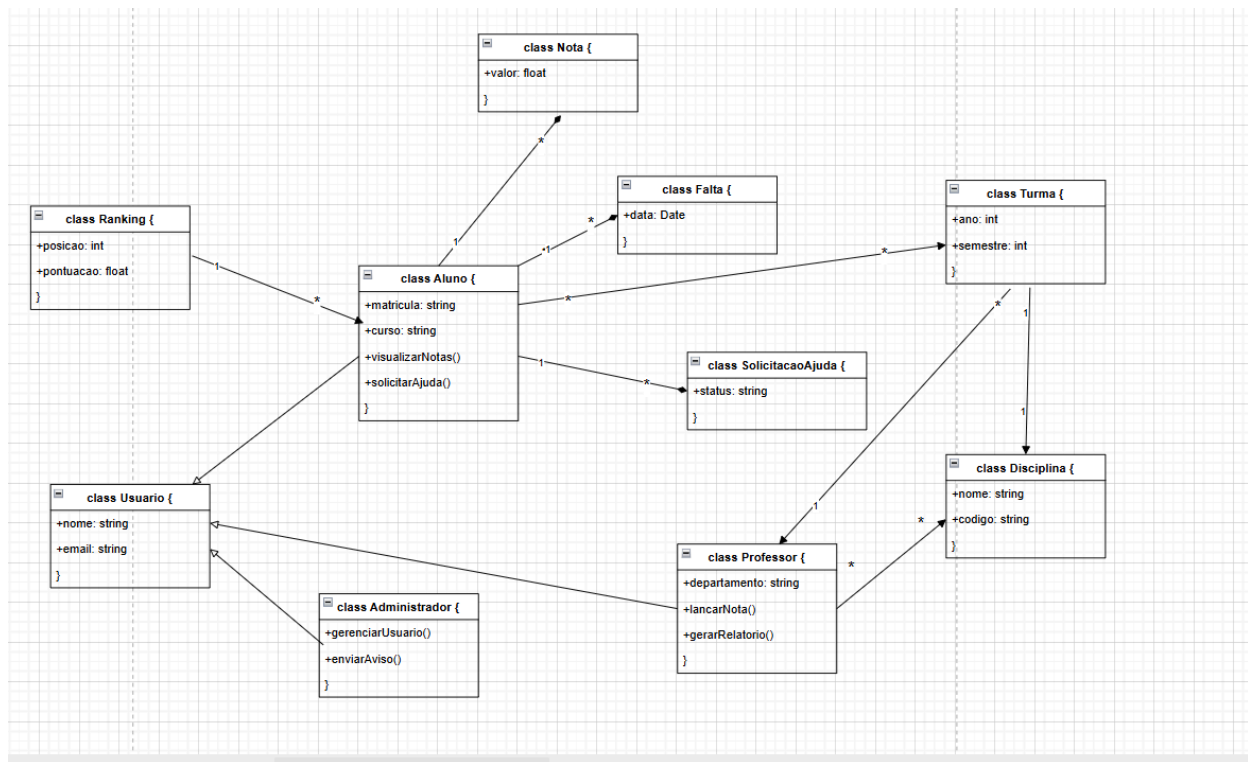
- **Associação:** Conexão geral entre classes (ex: Professor e Disciplina).
- **Agregação:** Tipo de associação "todo-parte" onde as partes podem existir independentemente do todo (ex: um Professor pode existir sem uma Turma específica).
- **Composição:** Tipo forte de agregação, onde a parte não pode existir sem o todo (ex: um Pedido

e seus Itens de Pedido).

- Herança (Generalização): Indica que uma classe (subclasse) herda características de outra (superclasse), promovendo reuso de código (ex: Aluno e Administrador herdam de Usuário). Exemplos de Aplicação incluem modelagem de sistemas de e-commerce (classes Cliente, Produto, Carrinho), sistemas bancários (Conta, Transação) e sistemas acadêmicos.

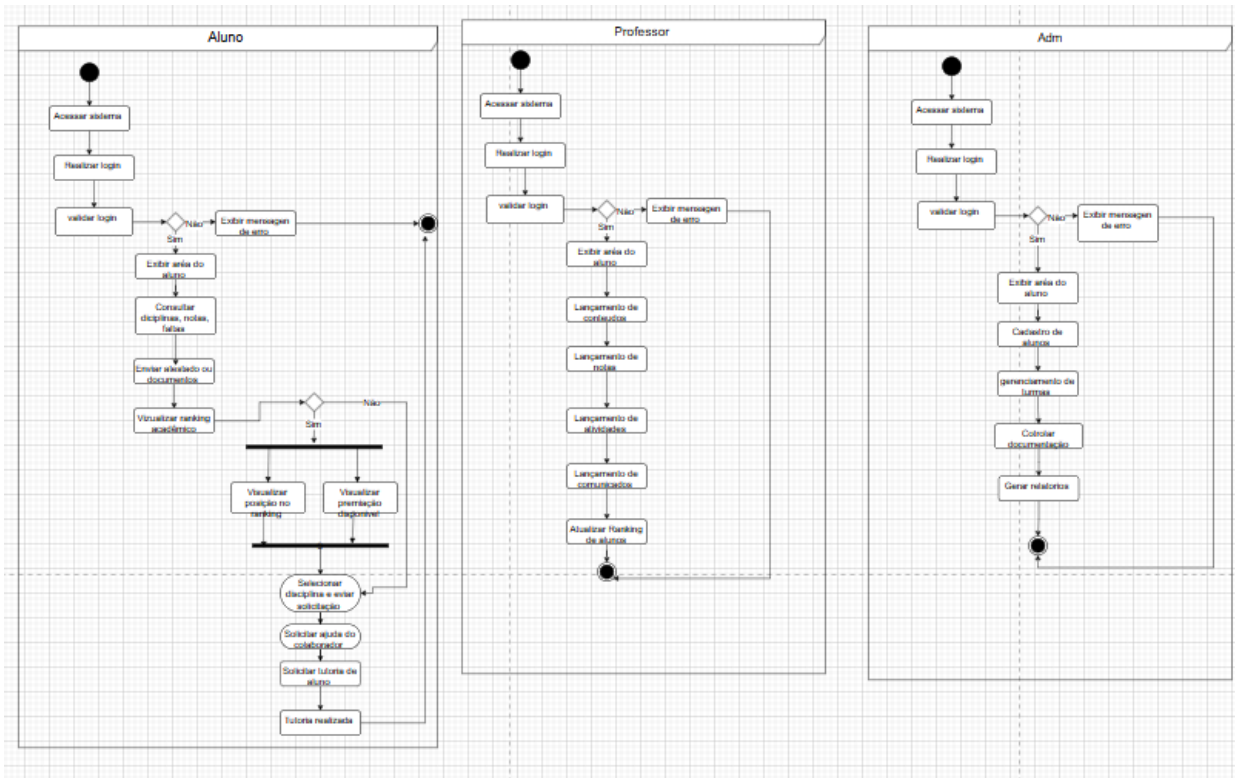
Aplicação no Sistema do Grupo: O diagrama fornecido modela um Sistema Acadêmico ou de Gestão Educacional. Ele utiliza a Herança (linha tracejada) para mostrar que Aluno e Administrador são tipos específicos de Usuário, reutilizando atributos como nome e email. A Associação (linha sólida) define interações como: um Aluno tem Notas e Faltas, e está associado a uma Turma. A composição (losango preenchido) entre Aluno e Ranking sugere que a pontuação de Ranking é intrinsecamente ligada à existência do Aluno. Este diagrama é crucial para definir a estrutura de dados e as funcionalidades (métodos) que o grupo precisará implementar no código.

### **6.2.2. Desenho/Imagem do Diagrama**



### 6.3. Diagrama de Atividades





### 6.3.1. Resumo

O diagrama de atividades elaborado representa o fluxo de ações do aluno dentro de um sistema acadêmico moderno e interativo, voltado para o aprimoramento da experiência universitária. Esse tipo de diagrama, pertencente à UML (Unified Modeling Language), é utilizado para descrever o comportamento dinâmico de um sistema, mostrando a sequência de atividades, decisões e interações entre usuário e sistema de forma clara e lógica.

No diagrama apresentado, cada swimlane representa um cenário de uso do aluno, dividido em três partes: acesso e rotinas acadêmicas, sistema de ranking acadêmico e sistema de ajuda colaborativa. O fluxo inicia sempre com o acesso ao sistema e a validação do login, seguido das ações específicas de cada funcionalidade. São utilizadas atividades (como “Consultar disciplinas”, “Acessar ranking”, “Solicitar ajuda”), decisões (como “Login válido?” ou “Deseja acessar ranking?”), além dos símbolos de início e fim, que delimitam o processo.

O diagrama também evidencia momentos de ramificação e sincronização — quando o aluno escolhe entre visualizar o ranking ou participar da ajuda colaborativa. Esses elementos demonstram forks e joins, responsáveis por representar fluxos alternativos e paralelos de execução.

Esse tipo de modelagem é essencial para visualizar e compreender o funcionamento do sistema, permitindo identificar o caminho percorrido pelo usuário, a lógica de processamento e as interações entre os módulos. No contexto do projeto, o diagrama ajuda o grupo a organizar as funcionalidades principais do sistema acadêmico, assegurando que cada etapa — desde o login até as ações de engajamento e colaboração — seja bem definida, coerente e funcional.

## 6.4. Diagrama de Sequência

### 6.4.1. Resumo

O diagrama de sequência é uma ferramenta da UML (Unified Modeling Language) utilizada para representar a troca de mensagens entre os objetos de um sistema ao longo do tempo. Ele descreve como as interações ocorrem, mostrando a ordem cronológica das chamadas de métodos e respostas entre atores e componentes internos do sistema. Esse tipo de diagrama é essencial para compreender o comportamento dinâmico do software, auxiliando desenvolvedores e analistas a identificar a sequência lógica dos processos.

No sistema **MonitorOn**, o diagrama de sequência demonstra o fluxo de comunicação entre o *Aluno*, o *Sistema Acadêmico* e os módulos internos, como *Ranking Acadêmico*, *Ajuda Colaborativa* e *Gestão de Disciplinas*. Por exemplo, quando um professor lança uma nota, a mensagem é enviada ao módulo de notas, que processa o valor e aciona o módulo de ranking para atualização automática. Da mesma forma, ao solicitar ajuda, o aluno interage com o módulo de colaboração, que busca e retorna os colegas qualificados disponíveis.

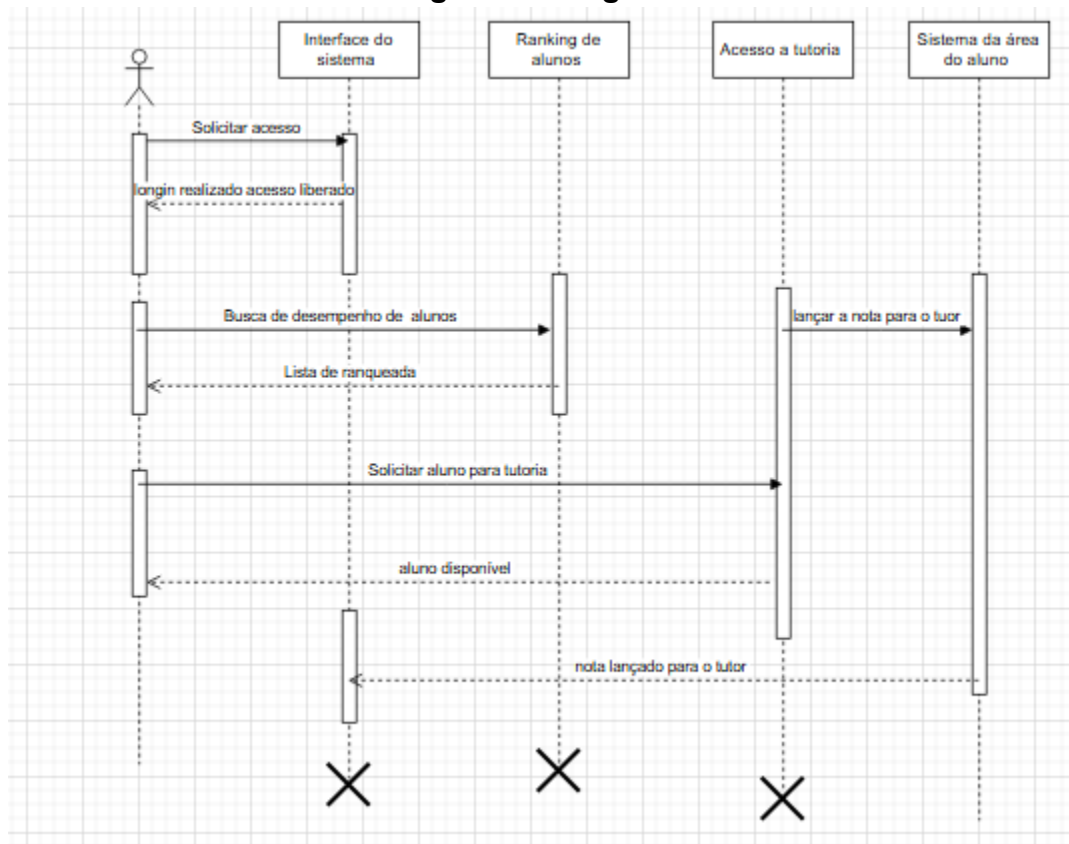
Os principais elementos do diagrama incluem:

- **Atores:** entidades externas que interagem com o sistema (Aluno, Professor, Coordenador).
- **Objetos:** instâncias que participam das trocas de mensagens (Sistema, Módulo de Notas, Banco de Dados, Ranking, etc.).
- **Lifelines (linhas de vida):** indicam a existência dos objetos durante a execução.
- **Mensagens:** representadas por setas, que mostram a comunicação entre os elementos.

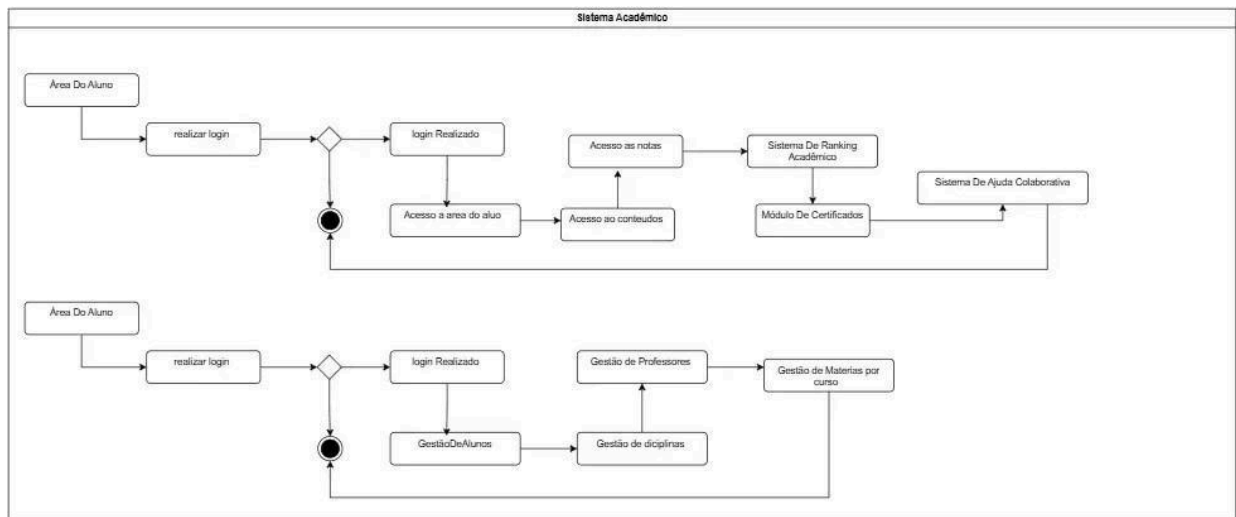
Esse diagrama é útil para garantir que o comportamento esperado de cada caso de uso

esteja correto e completo, além de servir como base para o desenvolvimento do código e para o planejamento de testes de integração do sistema acadêmico.

#### 6.4.2. Desenho/Imagem do Diagrama



## 6.5. Diagrama de Estado (ou Máquina de Estados)



### 6.5.1. Resumo

7. Esse diagrama representa o fluxo interno do Sistema Acadêmico, destacando as principais interações entre o aluno, os módulos administrativos e os sistemas complementares. O modelo está dividido em duas seções principais, que mostram os caminhos distintos de

acesso e

operação. Na parte superior, temos o fluxo voltado ao aluno. Ele inicia o processo ao realizar login na Área do Aluno. Após a autenticação, o aluno pode acessar sua área pessoal, visualizar notas, conteúdos e certificados.

O Sistema de Ranking Acadêmico usa essas informações para calcular e exibir o desempenho dos

estudantes. Em paralelo, o Sistema de Ajuda Colaborativa permite que o aluno solicite ou ofereça auxílio a

colegas. O Módulo de Certificados é acionado quando há necessidade de emissão de certificados acadêmicos.

Essa parte do diagrama reflete a interação contínua entre módulos de engajamento e desempenho.

8. Na parte inferior, o fluxo mostra as ações de gestão administrativa. Após o login, o usuário (professor ou coordenador) acessa o módulo de Gestão de Alunos. Esse módulo se conecta com o de Gestão de Professores e Gestão de Disciplinas. As disciplinas são organizadas e vinculadas às matérias de cada curso. O processo garante que dados acadêmicos e administrativos estejam sincronizados. Os fluxos convergem para um mesmo ponto de controle, representando integração total do sistema.

O uso de decisões e pontos de junção demonstra como o login direciona diferentes perfis de

usuário.

Cada módulo é representado como um componente funcional independente, conectado por fluxos

lógicos.

O diagrama enfatiza a modularidade e integração das funcionalidades. Assim, ele resume de forma clara como o sistema centraliza atividades acadêmicas e administrativas em uma única plataforma digital.

## **8.1. Diagrama de Componentes**

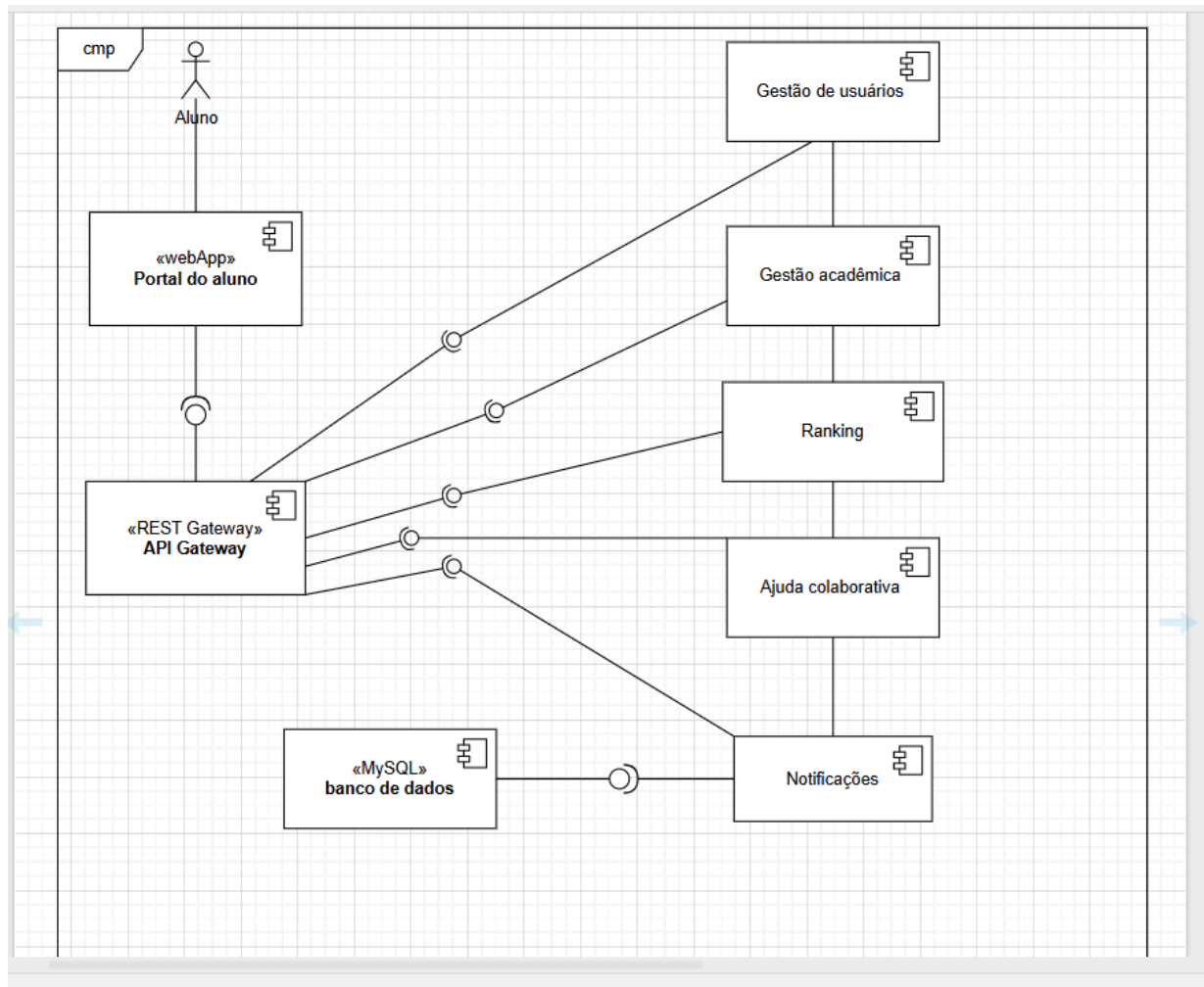
### **8.1.1. Resumo**

O Diagrama de Componentes é como a planta da casa de um sistema de software. Ele mostra as "peças" que formam o sistema (os componentes) e como elas se encaixam e conversam entre si. Sua principal função é visualizar a arquitetura, deixando claro como o software é organizado e quais tecnologias são usadas.

Os elementos chave são os componentes (como uma API, um banco de dados ou a interface do usuário), as interfaces (que mostram o que um componente oferece e o que ele precisa dos outros) e as dependências (as linhas que conectam tudo).

No projeto, o diagrama é perfeito para ilustrar a arquitetura de microserviços. Ele mostra que o Portal do aluno não acessa tudo diretamente, mas sim através de um API Gateway, que organiza o acesso aos diferentes serviços (Gestão de usuários, Ranking, etc.). Também deixa claro que todos esses serviços dependem de um único banco de dados. Para a equipe, isso serve como um mapa, ajudando todos a entenderem a estrutura do sistema e como as partes colaboram

### **8.1.2. Desenho/Imagem do Diagrama**





## 8.2. Diagrama de Objetos

### 8.2.1. Resumo

O diagrama de objetos é uma variação prática do diagrama de classes, usada para representar **instâncias reais das classes** em um momento específico da execução do sistema. Enquanto o diagrama de classes mostra a estrutura genérica, o de objetos mostra exemplos concretos, com **valores atribuídos aos atributos** e **relações específicas entre os objetos**.

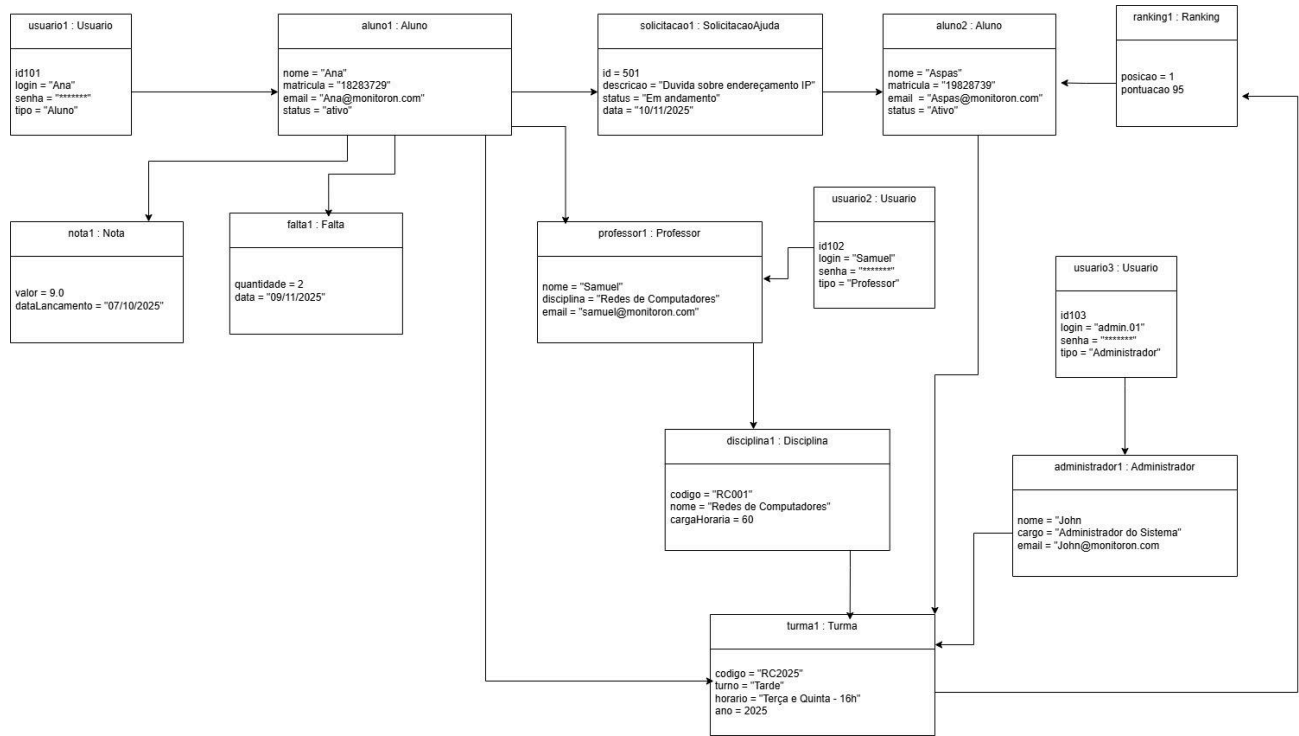
No sistema **MonitorOn**, o diagrama de objetos pode representar, por exemplo, um cenário em que o aluno *João Silva* (objeto da classe Aluno) está vinculado à disciplina *Programação Web* (objeto da classe Disciplina), possuindo uma *NotaFinal = 9.0* e ocupando a *3ª posição* no *Ranking Acadêmico*. Esse diagrama torna mais fácil visualizar a interação entre elementos em tempo de execução e verificar se as relações planejadas no diagrama de classes realmente se concretizam no uso do sistema.

Os principais elementos incluem:

- **Objetos:** mostrados como retângulos sublinhados (ex.: joao:Aluno).
- **Atributos:** exibem valores específicos de cada instância (ex.: notaFinal = 9.0).
- **Links:** indicam associações ativas entre objetos (ex.: Aluno vinculado a Disciplina).

Esse diagrama é essencial para validar a modelagem orientada a objetos e garantir que os relacionamentos definidos nas classes funcionem corretamente na prática. Ele também ajuda na identificação de possíveis ajustes nas estruturas de dados, contribuindo para um sistema mais consistente e eficiente.

## 8.2.2. Desenho/Imagem do Diagrama



## 9. CONCLUSÃO

O desenvolvimento do projeto **MonitorOn** permitiu aplicar na prática os conceitos de análise e modelagem de sistemas, desde a definição de requisitos até a criação de diagramas UML que representam diferentes visões do sistema. Durante o processo, aprendemos a importância da documentação de requisitos como ferramenta de comunicação entre desenvolvedores, usuários e gestores, garantindo clareza e alinhamento de expectativas.

A modelagem UML foi fundamental para entender o comportamento do sistema e visualizar suas interações, auxiliando na organização da lógica de funcionamento e na identificação de melhorias. Entre as maiores dificuldades enfrentadas, destacam-se a definição precisa dos requisitos e a criação dos diagramas, que exigem coerência e integração entre si.

Como resultado, o grupo conseguiu estruturar uma solução robusta, moderna e inovadora, capaz de otimizar a rotina acadêmica e incentivar a colaboração entre os alunos. O projeto proporcionou uma experiência completa de análise de sistemas, consolidando conhecimentos técnicos e práticos que serão essenciais no desenvolvimento de futuros softwares.

## 10. REFERÊNCIAS

<https://www.youtube.com/watch?v=2dSq0Vu1GFo>

<https://www.youtube.com/watch?v=rDidOn6KN9k>

<https://www.youtube.com/watch?v=LeV6RO-6Tn4>

<https://www.youtube.com/watch?v=CWj2F9dflaU>

PRESSMAN, Roger S.; MAXIM, Bruce R. *Engenharia de Software: uma abordagem profissional*. 9. ed. Porto Alegre: AMGH, 2021.

SOMMERVILLE, Ian. *Engenharia de Software*. 10. ed. São Paulo: Pearson Education do Brasil, 2019.

GUEDES, Gilleanes T. A. *UML 2: uma abordagem prática*. 2. ed. São Paulo: Novatec, 2018.

FOWLER, Martin. *UML Essencial: um breve guia para a linguagem padrão de modelagem de objetos*. 3. ed. Porto Alegre: Bookman, 2016.

### Ficha de Autoavaliação do Grupo

- A ficha de autoavaliação deve ser única por grupo e inserida na última página do trabalho.
- O preenchimento deve ser feito em conjunto, de forma justa e responsável.
- O objetivo é registrar o nível de participação de cada integrante no desenvolvimento do projeto.
- Cada critério deve ser avaliado de 0 a 5, onde 0 significa ausência de contribuição e 5 significa contribuição máxima.

Integrantes	Participação nas discussões	Produção de conteúdo escrito	Contribuição nos diagramas UML	Colaboração na organização/formatação	Participação na gravação/apresentação
Samuel Assunção Souza Lisboa	5	5	5	5	5
Tainara dos Anjos da Silva	5	5	5	5	5

João Madeiro Gomes Neto	5	5	5	5	5