



UNIVERSIDADE FEDERAL DO PARANÁ
SETOR DE EDUCAÇÃO PROFISSIONAL E
TECNOLÓGICA CURSO DE TECNOLOGIA EM
ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

Brendon Ferreira de Matos

Matheus Henrique Assumpção dos Reis

Michelly Narita Kuriyama

**SQLab: PLATAFORMA INTEGRADA PARA ENSINO DE
ADMINISTRAÇÃO DE BANCO DE DADOS E GERENCIAMENTO DE
INFORMAÇÃO**

CURITIBA

2024

Brendon Ferreira de Matos
Matheus Henrique Assumpção dos Reis
Michelly Narita Kuriyama

**SQLab: PLATAFORMA INTEGRADA PARA ENSINO DE
ADMINISTRAÇÃO DE BANCO DE DADOS E GERENCIAMENTO DE
INFORMAÇÃO**

Trabalho apresentado como requisito parcial à obtenção do grau de Tecnólogo em Análise e Desenvolvimento de Sistemas no curso de Tecnologia em Análise e Desenvolvimento de Sistemas, Setor de Educação Profissional e Tecnológica da Universidade Federal do Paraná.

Orientador: Prof. Dr. Dieval Guizelini

CURITIBA

2024

***Aos nossos pais, que sempre nos deram
apoio e incentivo para que nossos
sonhos se tornassem realidade.***

A vocês nossa eterna gratidão!

AGRADECIMENTOS

Concluimos o resumo deste curso com muita satisfação e entusiasmo. Esta é uma etapa cheia de desafios e recompensas. Primeiramente, agradecemos a Deus por nos dar força, sabedoria e oportunidade para trilhar esse caminho de aprendizado e crescimento. A vida traz dificuldades e recompensas, e somos gratos por cada momento.

Às nossas famílias, que são os verdadeiros pilares desta jornada, expressamos nossa mais profunda gratidão pelo apoio incondicional, amor e paciência. Suas palavras de incentivo e compreensão foram vitais para me manter firme mesmo nos momentos mais difíceis. Sinceros agradecimentos aos nossos amigos que estiveram ao nosso lado em cada passo do caminho, compartilhando risadas e desafios. Sua amizade tornou esta jornada mais fácil e significativa e cada conquista foi compartilhada.

Agradecemos especialmente aos nossos colegas de curso, cuja cooperação, troca de informações e espírito de equipe foram indispensáveis. Juntos superamos desafios e criamos um ambiente de aprendizagem enriquecedor. Aos professores, nossa eterna gratidão. Suas aulas iam além do conteúdo acadêmico para nos ensinar valores e habilidades para a vida. Em particular, queremos enfatizar a nossa gratidão ao professor orientador Dieval Guizelini. Sua dedicação, paciência e atenção foram fundamentais para a conclusão deste trabalho. Sua orientação útil e atenciosa fez toda a diferença e seremos eternamente gratos.

Por fim, gostaríamos de agradecer a todos que contribuíram de alguma forma para a conclusão deste trabalho. Essa conquista é resultado de muitos esforços e colaborações, e cada aprendizado e lembrança é guardado com amor e gratidão.

“Nada na vida deve ser temido, somente compreendido. Agora é hora de compreender mais para temer menos”.

Marie Curie

RESUMO

O ensino de banco de dados frequentemente enfrenta desafios significativos, com aulas predominantemente teóricas e escassas oportunidades práticas para os alunos. A falta de ferramentas adequadas para práticas efetivas e a dificuldade de acesso a ambientes de desenvolvimento são obstáculos recorrentes. Em resposta a essas limitações, a equipe desenvolveu o SQLab, uma plataforma projetada para revolucionar a experiência de aprendizado em administração de banco de dados. O SQLab apresenta uma interface web dividida em duas interfaces distintas: para professores, oferece a capacidade de configurar bancos de dados específicos para exercícios, gerenciar turmas, controlar notas e monitorar o desempenho dos alunos. Para os alunos, elimina a necessidade de configurações complexas, permitindo a realização de atividades práticas diretamente via web, acessível por meio de qualquer dispositivo conectado à internet. Utilizando PHP com Laravel para o backend e React para a interface web, o SQLab garante uma base sólida e moderna. Além disso, a implementação de funcionalidades avançadas, como sistemas de pontuação e rankings, busca aumentar a motivação dos usuários. Versões móveis baseadas em React Native estão sendo desenvolvidas para oferecer acessibilidade e interatividade através de dispositivos móveis. Com o SQLab, espera-se não apenas facilitar o ensino e aprendizado de SQL, mas também proporcionar um ambiente educacional mais engajador e acessível. Ao oferecer ferramentas avançadas para professores e uma plataforma intuitiva para alunos, o SQLab promete melhorar significativamente a experiência de estudantes e educadores no campo de administração de banco de dados.

Palavras-chave: Banco de dados. Aprendizado. SQL. Tecnologia educacional. Ensino interativo.

ABSTRACT

Teaching database administration often faces significant challenges, with predominantly theoretical classes and few practical opportunities for students. The lack of adequate tools for effective practices and difficulty accessing development environments are recurring obstacles. In response to these limitations, the team developed SQLab, a platform designed to revolutionize the learning experience in database administration. SQLab features a web interface divided into two distinct environments: for teachers, it offers the ability to configure databases specific to exercises, manage classes, control grades, and monitor student performance. For students, it eliminates the need for complex setups, enabling practical activities directly via the web, accessible on any internet-connected device. Using PHP with Laravel for the backend and React for the web interface, SQLab ensures a robust and modern foundation. Additionally, advanced features such as scoring systems and rankings aim to enhance user motivation. Mobile versions based on React Native are being developed to provide accessibility and interactivity on mobile devices. SQLab aims not only to facilitate SQL teaching and learning but also to provide a more engaging and accessible educational environment. By offering advanced tools for teachers and an intuitive platform for students, SQLab promises to significantly enhance the experience of students and educators in the field of database administration.

Keywords: Database. Practical learning. SQL. Educational technology. Interactive teaching.

SUMÁRIO

1 INTRODUÇÃO	1
1.1 PROBLEMA	2
1.2.1 GERAIS	3
1.3 JUSTIFICATIVA	4
2 FUNDAMENTAÇÃO TEÓRICA	6
2.1 AMBIENTE DE APRENDIZAGEM EM SQL PARA ESTUDANTES	6
2.2 METODOLOGIAS PEDAGÓGICAS PARA O ENSINO DE SQL	7
2.2 SOFTWARES SIMILARES	8
2.2.1 FARMAALG	8
2.2.2 COURSERA	9
3 MATERIAL E MÉTODOS	11
3.1 METODOLOGIA DE DESENVOLVIMENTO	11
3.1.1 UML	11
3.1.2 KANBAN	12
3.2 FERRAMENTAS DE DESENVOLVIMENTO	13
3.2.1 REACT	13
3.2.2 PHP	14
3.2.3 LARAVEL	15
3.2.4 POSTGRESQL	15
3.2.5 DRAW.IO	16
3.2.6 EXCALIDRAW	17
4 APRESENTAÇÃO DO SISTEMA	19
4.1 MODELAGEM DO PROJETO	20
4.1.1 DIAGRAMA DE CASO DE USO	20
4.1.2 TABELA DE REQUISITOS	20
4.1.3 DIAGRAMA DE CLASSES	20

4.1.4 DIAGRAMA DE SEQUÊNCIA.....	21
4.2 ARQUITETURA DO SISTEMA.....	21
4.3 FUNCIONALIDADES.....	22
5 REFERÊNCIAS	41
6 APÊNDICES	43
APÊNDICE A – DIAGRAMA DE CASO DE USO	43
APÊNDICE B – TABELA DE REQUISITOS	44
APÊNDICE C – ESPECIFICAÇÕES DE CASO DE USO	46
APÊNDICE D – DIAGRAMA DE CLASSE DE ANÁLISE	105
APÊNDICE E – DIAGRAMA DE SEQUÊNCIA DE ANÁLISE	106
APÊNDICE F – MODELO LÓGICO DE DADOS	124

1 INTRODUÇÃO

O ensino de linguagens de programação, como a SQL (Structured Query Language), enfrenta desafios significativos na criação de experiências de aprendizado práticas e interativas para os alunos. Tradicionalmente, as aulas teóricas são complementadas por exercícios práticos utilizando sistemas de gerenciamento de bancos de dados como MySQL e PostgreSQL. No entanto, a oportunidade para os alunos praticarem autonomamente e aplicarem seus conhecimentos em contextos reais é limitada. Esta limitação se deve, em parte, à escassez de recursos educacionais acessíveis que permitam aos alunos desenvolver habilidades práticas de SQL de maneira eficaz (Clark, 1994; Kirschner et al., 2006).

A abordagem convencional de atribuir atividades para serem realizadas em casa enfrenta diversos obstáculos. Em primeiro lugar, a acessibilidade é uma preocupação central: nem todos os alunos têm acesso aos recursos necessários para configurar e operar ambientes de aprendizado adequados. A preparação de ambientes de desenvolvimento pode ser complexa e exigir conhecimentos técnicos adicionais, o que pode desencorajar muitos estudantes iniciantes. Além disso, as ferramentas disponíveis frequentemente não são projetadas especificamente para o ensino de programação, limitando-se a formatos de entrega de tarefas que não proporcionam uma interação dinâmica e imediata com o conteúdo de SQL (Siemens, 2005).

Em contrapartida, avanços recentes na tecnologia educacional oferecem oportunidades promissoras para superar esses desafios. O presente trabalho de conclusão de curso propõe o desenvolvimento de uma plataforma de aprendizado inovadora, focada em SQL, que permite aos alunos praticar habilidades de consulta e manipulação de dados em um ambiente virtual interativo. Esta aplicação não só oferecerá acesso a recursos educacionais estruturados e atualizados, mas também permitirá que os educadores monitorem o progresso dos alunos de forma mais eficaz e personalizada.

Ao integrar tecnologia de ponta com metodologias pedagógicas comprovadas, a plataforma proposta tem o potencial de revolucionar o ensino de SQL. Estudantes de tecnologia e programação poderão beneficiar-se significativamente, adquirindo

habilidades práticas essenciais para o mercado de trabalho moderno. Além disso, professores terão à disposição ferramentas poderosas para reforçar o aprendizado da matéria, oferecer feedback individualizado e acompanhar o desenvolvimento de seus alunos de maneira mais detalhada.

1.1 PROBLEMA

Uma das principais dificuldades do ensino de SQL reside na lacuna entre a teoria apresentada em sala de aula e a aplicação prática necessária no mercado de trabalho. As aulas frequentemente se limitam a exposições teóricas, com poucas oportunidades para os alunos desenvolverem habilidades práticas efetivas em consultas e manipulação de dados reais (Clark, 1994; Kirschner et al., 2006).

A acessibilidade a recursos para estudos da área é um dos pontos a serem destacados dentro deste assunto. Muitos alunos enfrentam obstáculos para configurar e utilizar ambientes de desenvolvimento apropriados para praticar SQL. A complexidade técnica envolvida na instalação de sistemas de gerenciamento de banco de dados como MySQL ou PostgreSQL pode desencorajar estudantes menos familiarizados com essas tecnologias, impactando negativamente seu aprendizado e participação efetiva nas atividades acadêmicas (Siemens, 2005).

As ferramentas educacionais disponíveis muitas vezes não são projetadas especificamente para o ensino específico de SQL, e não trazem a visibilidade e experiência prática necessária para absorvimento de conteúdo da parte dos alunos. Estas ferramentas frequentemente se limitam a formatos de resposta simplificados, como perguntas de múltipla escolha ou caixas de texto simples, que não proporcionam uma interação dinâmica e imediata com o conteúdo. Além disso, a falta de recursos para acompanhamento e avaliação por parte dos professores dificulta a monitoração do progresso dos alunos e a identificação precisa de áreas que necessitam de reforço (Clark, 1994). Esses desafios têm um impacto significativo na motivação dos alunos e na eficiência do ensino. A falta de experiências práticas e interativas pode diminuir o interesse dos estudantes pela disciplina e pela área de tecnologia como um todo.

Para os professores, a dificuldade em gerenciar turmas numerosas e avaliar o desempenho dos alunos de maneira eficaz pode resultar em uma sobrecarga de

trabalho e na dificuldade de fornecer suporte individualizado adequado a cada estudante. Diante dessas dificuldades, há uma necessidade urgente de soluções inovadoras que superem as barreiras existentes no ensino de SQL. Plataformas educacionais específicas, que integrem tecnologias avançadas com metodologias pedagógicas eficazes, têm o potencial de transformar a forma como o SQL é ensinado e aprendido.

1.2 OBJETIVOS

1.2.1 GERAIS

Desenvolver e implementar uma plataforma educacional inovadora focada no ensino de SQL, visando oferecer uma experiência de aprendizado prática, interativa e acessível para alunos de tecnologia. A plataforma será projetada para superar os desafios atuais no ensino de SQL, proporcionando atividades práticas, feedback imediato e ferramentas avançadas para professores acompanharem o progresso dos mesmos.

1.2.2 OBJETIVOS ESPECÍFICOS

- A) Definir Requisitos Funcionais e Não Funcionais: Documentar requisitos detalhados, incluindo funcionalidades específicas da plataforma (como execução de consultas SQL em tempo real) e requisitos não funcionais (como desempenho e segurança).
- B) Projetar e Desenvolver Interfaces Intuitivas: Criar interfaces de usuário intuitivas e acessíveis, seguindo princípios de User Experience (UX), para facilitar a navegação e utilização por alunos e professores.
- C) Implementar Funcionalidades de SQL Interativas: Desenvolver funcionalidades que permitam aos alunos escrever e executar consultas SQL diretamente na plataforma, manipular dados e visualizar resultados de consultas em tempo real.
- D) Integrar Ferramentas de Feedback e Avaliação: Implementar ferramentas

que permitam aos professores avaliar o progresso dos alunos, analisar desempenhos individuais e coletivos, e fornecer feedback personalizado sobre atividades realizadas.

- E) Implementar Ferramentas de Acompanhamento Acadêmico: Desenvolver funcionalidades que permitam aos professores acompanhar o desempenho dos alunos ao longo do curso, identificar áreas de dificuldade e oferecer suporte personalizado conforme necessário.

1.3 JUSTIFICATIVA

A justificativa para o desenvolvimento desta plataforma educacional baseia-se em desafios significativos enfrentados no ensino atual de SQL, especialmente em ambientes acadêmicos de tecnologia. Atualmente, os métodos tradicionais de ensino frequentemente limitam-se a aulas teóricas, com poucas oportunidades práticas para os alunos aplicarem seus conhecimentos em bancos de dados reais. Esta abordagem resulta em uma lacuna entre o entendimento teórico e a aplicação prática da linguagem SQL, essencial para o gerenciamento de bancos de dados.

Muitas vezes as ferramentas disponíveis para atividades práticas muitas vezes são inadequadas, oferecendo apenas exercícios simplificados ou não integrados com ambientes de banco de dados reais. Isso dificulta a capacidade dos alunos de desenvolverem habilidades práticas necessárias no mercado de trabalho e limita o engajamento e a motivação dos estudantes no aprendizado da disciplina.

A implementação desta plataforma visa superar esses desafios ao proporcionar uma experiência de aprendizado prático e interativo. Permitirá aos alunos não apenas aprenderem a teoria por trás das consultas SQL, mas também praticarem diretamente em um ambiente simulado de banco de dados, onde poderão escrever consultas, manipular dados e observar os resultados em tempo real. Isso não só aumentará a compreensão e a confiança dos alunos na aplicação da SQL, mas também preparará melhor para desafios futuros na carreira profissional.

Além dos benefícios para os alunos, a plataforma fornecerá ferramentas avançadas para os professores acompanharem o progresso dos estudantes, identificarem áreas de dificuldade e fornecerem suporte personalizado. Essa

abordagem não apenas melhora a eficácia do ensino, mas também aumenta o envolvimento dos alunos ao proporcionar um feedback imediato e relevante sobre seu desempenho.

Portanto, o desenvolvimento desta plataforma educacional não apenas endereça as deficiências atuais no ensino de SQL, mas também promove um ambiente de aprendizado mais eficaz, engajador e preparatório para os desafios práticos enfrentados pelos futuros profissionais de tecnologia.

2 FUNDAMENTAÇÃO TEÓRICA

A evolução contínua das tecnologias de informação e comunicação tem redefinido as expectativas e necessidades no campo da educação (Reimers & Schleicher, 2020). Com o aumento da demanda por profissionais qualificados em tecnologia e bancos de dados, é imperativo que os métodos de ensino acompanhem essas mudanças, oferecendo não apenas conhecimento teórico, mas também habilidades práticas e aplicadas.

Os desafios enfrentados no ensino de SQL também incluem a heterogeneidade de experiências e habilidades dos alunos. Muitos estudantes entram em cursos de tecnologia com diferentes níveis de familiaridade com bancos de dados e linguagens de programação, o que torna essencial a adoção de uma abordagem flexível e adaptável (Fisher & Baird, 2011).

Além da necessidade de personalização e adaptação ao perfil diversificado dos alunos, o SQLab também aborda a importância crescente da avaliação formativa e do feedback construtivo. Segundo Hattie & Timperley (2007), o feedback construtivo é fundamental para melhorar o desempenho dos alunos e promover o aprendizado efetivo. Ao fornecer ferramentas que permitem aos professores monitorarem o progresso dos alunos em tempo real, identificar áreas de melhoria e oferecer orientação personalizada, a plataforma não apenas facilita o aprendizado autônomo, mas também fortalece a relação colaborativa entre educadores e estudantes.

2.1 AMBIENTE DE APRENDIZAGEM EM SQL PARA ESTUDANTES

O aprendizado de SQL é essencial para qualquer estudante que pretende trabalhar com gerenciamento de bancos de dados. A compreensão de como projetar e estruturar bancos de dados, por meio da modelagem de dados, é fundamental para o desenvolvimento de soluções eficientes e organizadas. Os alunos devem entender conceitos como tabelas, chaves primárias, chaves estrangeiras e os relacionamentos entre tabelas. A modelagem de dados é a base sobre a qual são construídas as consultas e manipulações subsequentes de dados.

À medida que os alunos avançam no estudo de SQL, as consultas se tornam

cada vez mais complexas. A utilização de junções (INNER JOIN, LEFT JOIN, etc.), subconsultas e agregações (GROUP BY, HAVING) pode ser desafiadora, especialmente para iniciantes. A habilidade de construir consultas complexas é crucial para realizar análises de dados detalhadas e extrair informações significativas a partir de grandes conjuntos de dados. Sem uma prática adequada, os estudantes podem se sentir sobrecarregados e frustrados.

A otimização de consultas é outra área crítica onde os alunos frequentemente encontram dificuldades. Escrever consultas eficientes que utilizem o mínimo de recursos e tenham um desempenho rápido é uma habilidade que se desenvolve com a experiência e a prática constante. A falta de compreensão sobre índices, planos de execução e técnicas de otimização pode levar a consultas que são lentas e consomem muitos recursos, impactando negativamente a performance dos sistemas de banco de dados.

Além das questões técnicas, aspectos como segurança e permissões também são fundamentais no aprendizado de SQL. Saber como conceder e revogar permissões de acesso a tabelas e bancos de dados é crucial para manter a integridade e a segurança dos dados. Erros na configuração de permissões podem resultar em vulnerabilidades de segurança significativas. A capacidade de lidar com erros, exceções e problemas de integridade de dados é essencial para garantir que os sistemas de banco de dados sejam robustos e resilientes.

2.2 METODOLOGIAS PEDAGÓGICAS PARA O ENSINO DE SQL

As metodologias pedagógicas desempenham um papel crucial no ensino de SQL, influenciando diretamente a eficácia e o engajamento dos alunos. Uma abordagem construtivista, onde os alunos constroem seu próprio conhecimento através da interação ativa com o conteúdo, é particularmente eficaz no ensino de linguagens de programação. A aplicação de métodos como a aprendizagem baseada em problemas (PBL) e a aprendizagem ativa pode proporcionar um entendimento mais profundo e duradouro dos conceitos de SQL (Jonassen, 1999).

A aprendizagem baseada em problemas (PBL) envolve a apresentação de problemas reais aos alunos, incentivando-os a encontrar soluções através da

pesquisa e da prática. No contexto do ensino de SQL, isso pode incluir a análise e solução de problemas de bancos de dados reais, permitindo que os alunos apliquem teorias em situações práticas. Esta abordagem não apenas melhora a compreensão dos conceitos, mas também desenvolve habilidades críticas de resolução de problemas e pensamento analítico (Hmelo-Silver, 2004).

A aprendizagem ativa, por outro lado, envolve a participação dos alunos em atividades que promovem a análise, a síntese e a avaliação do conteúdo aprendido. Em SQL, isso pode incluir exercícios interativos, discussões em grupo e simulações de consultas e manipulações de dados. A utilização de ferramentas tecnológicas que oferecem feedback instantâneo pode melhorar significativamente a eficácia da aprendizagem ativa, permitindo que os alunos corrijam erros e consolidem seu entendimento em tempo real (Prince, 2004).

2.2 SOFTWARES SIMILARES

2.2.1 FARMAALG

Desenvolver e implementar uma plataforma educacional inovadora focada no ensino de SQL, visando oferecer uma experiência de aprendizado prática, interativa e acessível para alunos de tecnologia. A plataforma será projetada para superar os desafios atuais no ensino de SQL, proporcionando atividades práticas, feedback imediato e ferramentas avançadas para professores acompanharem o progresso dos alunos.

A ferramenta executa o código submetido pelo aluno utilizando entradas predefinidas e compara os resultados esperados previamente cadastrados. O sistema fornece feedback instantâneo, promovendo uma correção ágil e eficiente para o aluno, o que facilita o aprendizado prático e a retenção de conceitos fundamentais de programação.

Atualmente em sua terceira versão, o FarmaALG é disponibilizado exclusivamente como uma plataforma online acessível via navegador. A primeira versão, também criada por KUTZKE (2015), foi concebida como uma interface web que suportava as linguagens C e Pascal para a gestão online dos códigos-fonte de

exercícios entregues pelos alunos. A interface de entrega de atividades do FarmaALG se assemelha a um editor de texto ou IDE moderna, apresentando exemplos da entrega proposta e dicas exibidas ao longo do desenvolvimento do código.

Um dos diferenciais do FarmaALG, conforme análises de KUTZKE (2015) e MUELLER et al. (2021), é a capacidade de comparar respostas de diferentes alunos ou múltiplas submissões do mesmo aluno. As semelhanças e diferenças são exibidas lado a lado, permitindo uma verificação humana de lógica similar ou plágio. Este recurso é particularmente útil para identificar padrões de aprendizado e possíveis casos de plágio, fornecendo uma ferramenta robusta para professores monitorarem o progresso e a originalidade dos trabalhos dos alunos.

A plataforma continua a evoluir, incorporando feedback dos usuários para melhorar suas funcionalidades e a experiência de aprendizado. A disponibilização exclusiva via navegador facilita o acesso e a utilização, permitindo que alunos e professores utilizem a ferramenta em qualquer lugar e a qualquer momento. Além disso, a interface intuitiva e o feedback instantâneo tornam o processo de aprendizado mais dinâmico e interativo, incentivando os alunos a corrigirem seus erros e melhorarem suas habilidades de programação.

Em resumo, o FarmaALG representa um avanço significativo no ensino de programação, integrando conceitos de aprendizado baseado no erro com tecnologias modernas de desenvolvimento web. Ao proporcionar uma plataforma acessível e eficiente para o aprendizado prático de linguagens de programação, o FarmaALG contribui para a formação de profissionais mais capacitados e preparados para enfrentar os desafios do mercado de tecnologia.

2.2.2 COURSERA

O Coursera é uma das melhores plataformas de ensino online, altamente recomendada para estudantes, funcionários ou gerentes que buscam aprimorar suas habilidades e se manter atualizados nas novidades de suas áreas de trabalho. A plataforma oferece uma ampla gama de cursos gratuitos, tutoriais e outros recursos educativos sobre praticamente qualquer assunto de interesse. Esses cursos são ministrados por professores universitários das melhores instituições ao redor do

mundo, garantindo a alta qualidade do conteúdo disponível.

Uma das principais vantagens do Coursera é a vasta quantidade de cursos voltados para a programação. A plataforma oferece opções que abrangem desde introduções à programação, HTML e CSS, até cursos mais avançados em programação para IoT, linguagem C, Python, Java e muitos outros. Embora os cursos sejam gratuitos, aqueles que desejam obter certificações oficiais pelo Coursera precisam pagar uma taxa. No entanto, a oportunidade de aprender de maneira gratuita e de alta qualidade é uma grande vantagem, especialmente para iniciantes que buscam adquirir novas habilidades ou aprimorar conhecimentos existentes.

Além dos cursos técnicos, o Coursera também disponibiliza uma variedade de conteúdos não técnicos. Estes incluem tópicos como design de produtos, marketing digital e planejamento de impostos, entre muitos outros. A diversidade de assuntos abordados na plataforma permite que os usuários ampliem seus conhecimentos em diferentes áreas, tornando-se mais versáteis e competentes em suas carreiras. Todos os cursos são desenvolvidos com conteúdos relevantes e interessantes, projetados para atender às necessidades de um público global.

O Coursera utiliza metodologias de ensino modernas, incluindo aulas em vídeo, leituras, quizzes e fóruns de discussão, que facilitam o aprendizado interativo e colaborativo. A plataforma também oferece funcionalidades avançadas, como a possibilidade de seguir um cronograma flexível, permitindo que os alunos aprendam no seu próprio ritmo. Embora a plataforma seja em inglês, muitos cursos possuem legendas em vários idiomas, o que amplia o acesso ao conteúdo para uma audiência global. No geral, o Coursera é uma excelente ferramenta para quem deseja expandir seus conhecimentos e habilidades de forma acessível e eficiente.

3 MATERIAL E MÉTODOS

3.1 METODOLOGIA DE DESENVOLVIMENTO

3.1.1 UML

A UML (Unified Modeling Language) é uma linguagem de modelagem padronizada amplamente utilizada na engenharia de software para especificar, visualizar, construir e documentar sistemas complexos. Criada na década de 1990, a UML facilita a compreensão compartilhada entre desenvolvedores e outros stakeholders, promovendo a clareza no design de sistemas. Composta por diagramas estruturais e comportamentais, a UML permite a representação detalhada tanto da organização estática quanto da dinâmica dos sistemas. Os diagramas estruturais, como o diagrama de classes, diagrama de objetos e diagrama de componentes, ajudam a ilustrar a estrutura do sistema, incluindo classes, atributos, operações e os relacionamentos entre as classes. Já os diagramas comportamentais, como o diagrama de casos de uso, diagrama de sequência e diagrama de estados, focam nas interações entre objetos e eventos, destacando como os componentes do sistema interagem e mudam de estado ao longo do tempo (Booch, Rumbaugh & Jacobson, 2005). Oferece uma linguagem comum que reduz ambiguidades, facilita a documentação clara e organizada do sistema e permite a visualização de problemas e soluções de maneira intuitiva. Além disso, a UML auxilia no planejamento e design, ajudando a identificar e resolver problemas potenciais antes que eles se tornem críticos. A reutilização de componentes e padrões de design é outra vantagem importante, promovendo eficiência e qualidade no desenvolvimento de software. Referências notáveis sobre UML, como "The Unified Modeling Language User Guide" de Booch, Rumbaugh e Jacobson (2005), e "UML Distilled: A Brief Guide to the Standard Object Modeling Language" de Martin Fowler (2004), oferecem uma compreensão aprofundada e prática sobre como utilizar essa ferramenta essencial no desenvolvimento de software. A aplicação eficaz da UML contribui para um desenvolvimento mais estruturado, comunicativo e eficiente, refletindo diretamente na qualidade dos sistemas criados.

3.1.2 KANBAN

O Kanban é uma metodologia ágil que se destaca pela simplicidade e eficiência na organização e gestão de atividades, sendo amplamente utilizada no desenvolvimento de sistemas. Originado no sistema de produção da Toyota na década de 1940, o Kanban visa melhorar a produtividade e a eficiência através da visualização do trabalho e do controle do fluxo de tarefas. Utilizando quadros, geralmente divididos em colunas que representam diferentes estágios de um processo (como "A Fazer", "Em Progresso" e "Concluído"), o Kanban permite que equipes de desenvolvimento visualizem facilmente o progresso das atividades, identifiquem gargalos e priorizem tarefas de forma colaborativa e dinâmica.

Os quadros Kanban são ferramentas poderosas para a gestão de projetos, pois oferecem uma visualização clara e imediata do status das tarefas. Isso facilita a comunicação entre os membros da equipe e permite uma rápida adaptação às mudanças de prioridade, melhorando a flexibilidade do projeto. Além disso, ao limitar o número de tarefas em progresso (WIP - Work In Progress), o Kanban ajuda a evitar a sobrecarga de trabalho e a focar na conclusão de tarefas antes de iniciar novas, promovendo um fluxo de trabalho mais eficiente e sustentável. Essa abordagem não apenas aumenta a produtividade, mas também melhora a qualidade do produto final, pois as equipes podem se concentrar em resolver problemas de forma mais eficaz (Kniberg, 2011).

A aplicação do Kanban no desenvolvimento de sistemas proporciona vários benefícios tangíveis. Primeiramente, a transparência é significativamente aumentada, permitindo que todos os membros da equipe tenham uma visão clara do que está sendo feito e do que precisa ser feito. Isso não apenas melhora a coordenação, mas também fortalece a colaboração e o senso de responsabilidade entre os membros da equipe. Em segundo lugar, a capacidade de visualizar e gerenciar o fluxo de trabalho em tempo real ajuda a identificar e mitigar problemas antes que eles se tornem críticos, reduzindo atrasos e melhorando a entrega contínua de valor ao cliente (Anderson, 2010).

Além disso, o Kanban é flexível e pode ser facilmente integrado com outras metodologias ágeis, como Scrum, para otimizar ainda mais a gestão de projetos. Sua

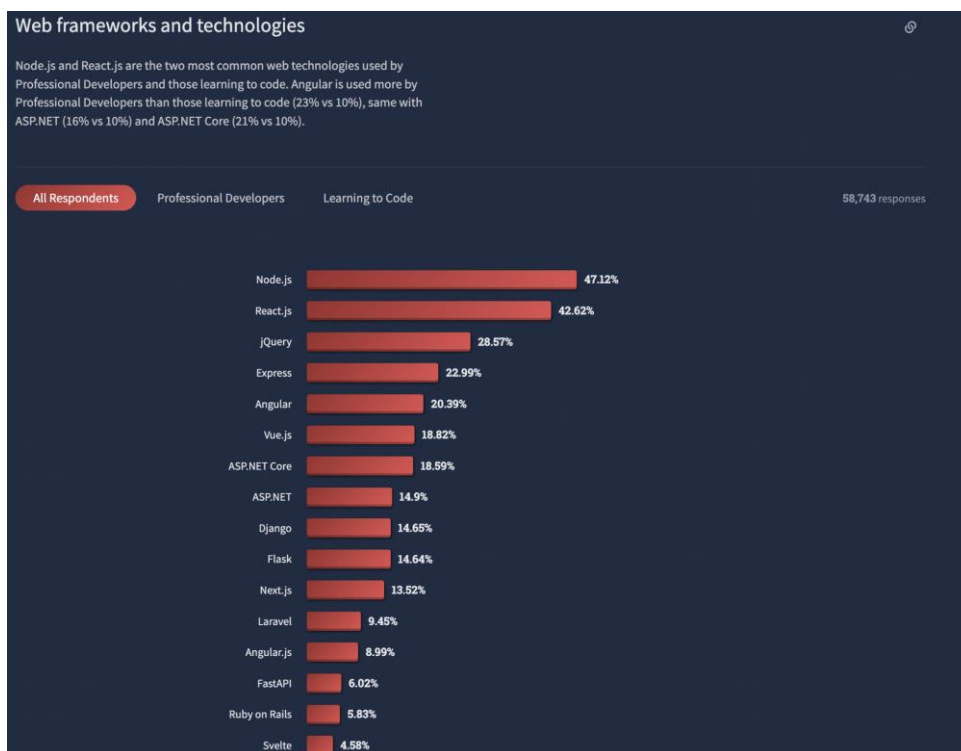
implementação não requer mudanças drásticas nos processos existentes, o que facilita sua adoção gradual e permite melhorias contínuas baseadas em feedback real do fluxo de trabalho. Essa adaptabilidade é crucial no ambiente dinâmico do desenvolvimento de software, onde as necessidades e prioridades podem mudar rapidamente. Em resumo, o Kanban é uma ferramenta valiosa que, ao promover uma gestão visual e colaborativa das atividades, contribui significativamente para a eficiência, a qualidade e a satisfação das equipes e dos clientes no desenvolvimento de sistemas.

3.2 FERRAMENTAS DE DESENVOLVIMENTO

3.2.1 REACT

React é uma biblioteca front-end baseada JavaScript, idealizada pela empresa Facebook (Meta) em 2013. Seu objetivo é criar interfaces para navegadores web. Ou seja, o React renderiza de forma eficiente apenas os componentes necessários conforme os dados mudam. Isso é possível porque ele implementa um processo de virtualização do DOM, onde uma representação ideal do DOM é armazenada na memória do navegador.

Quando há uma atualização dessa representação, o React a compara com o DOM "real" e altera apenas o que foi modificado, otimizando assim os processos de renderização. A estrutura do React é composta por componentes com baixo acoplamento, o que facilita a replicação, manutenção e depuração do código (REACT, 2023). No ano de 2022, o StackOverflow realizou uma comparação entre frameworks e tecnologias web, onde o React ficou apenas atrás do Node.js:



A versatilidade de bibliotecas que podem compor a aplicação React faz com que ele possa se tornar quase um framework, similar ao Vue.js e ao Angular. Para compor essa medida, podemos utilizar as bibliotecas React, Router Dom, para manipulação de rotas, React Query para cacheamento de dados, junto do axios para fazer o tratamento das requisições HTTP via REST.

3.2.2 PHP

PHP é uma linguagem de programação de scripts do lado do servidor, criada por Rasmus Lerdorf em 1994. Ela é amplamente utilizada para o desenvolvimento de aplicações web dinâmicas e interativas.

PHP é especialmente valorizado por sua simplicidade e flexibilidade, permitindo aos desenvolvedores criar páginas web rapidamente. Além disso, sua integração fácil com bancos de dados, como MySQL, e sua ampla comunidade de usuários fazem do PHP uma escolha popular para muitos projetos web. Sua capacidade de embutir código PHP diretamente em HTML torna a linguagem acessível para iniciantes, enquanto suas características avançadas suportam o desenvolvimento de aplicações complexas (PHP, 2023).

3.2.3 LARAVEL

Laravel O framework PHP de código aberto criado por Taylor Otwell em 2011. Ele foi desenvolvido para facilitar o desenvolvimento de aplicações web robustas e bem estruturadas, fornecendo um conjunto de ferramentas e recursos que ajudam a simplificar tarefas comuns, como roteamento, autenticação e gerenciamento de banco de dados.

Laravel adota o padrão de arquitetura MVC (Model-View-Controller), o que promove a separação de preocupações e torna o código mais modular e fácil de manter. Entre os recursos mais apreciados do Laravel estão o Eloquent ORM (para manipulação de banco de dados de forma intuitiva), o Artisan (interface de linha de comando poderosa), e o suporte para testes automatizados. Esses recursos, juntamente com uma documentação extensa e uma comunidade ativa, fazem do Laravel uma das escolhas mais populares para o desenvolvimento de aplicações web em PHP (LARAVEL, 2023).

3.2.4 POSTGRESQL

O PostgreSQL é um sistema de gerenciamento de banco de dados objeto-relacional (ORDBMS) de código aberto que tem se destacado no mercado por sua robustez, flexibilidade e conformidade com os padrões SQL. Lançado inicialmente em 1989 como parte do projeto POSTGRES na Universidade da Califórnia em Berkeley, o PostgreSQL evoluiu significativamente ao longo dos anos, tornando-se uma das escolhas mais populares entre desenvolvedores e empresas de todo o mundo. Sua arquitetura avançada, extensibilidade e forte foco em integridade de dados o tornam uma opção atraente para uma ampla variedade de aplicações, desde pequenas startups até grandes corporações.

Uma das principais razões para a ampla adoção do PostgreSQL é sua aderência rigorosa aos padrões SQL e seu suporte a uma vasta gama de funcionalidades avançadas. Ao contrário de alguns outros bancos de dados que podem sacrificar a conformidade com padrões em favor de funcionalidades proprietárias, o PostgreSQL mantém um compromisso firme com a padronização, o

que facilita a portabilidade e a manutenção do código. Além disso, ele oferece suporte nativo para tipos de dados avançados, como JSON, XML e arrays, bem como para funções complexas, índices personalizados e operações geométricas, tornando-o extremamente versátil e capaz de atender a diversas necessidades de armazenamento e processamento de dados.

Outro fator que contribui para a popularidade do PostgreSQL é sua extensibilidade. Desenvolvedores podem criar suas próprias funções, tipos de dados, operadores e índices personalizados, permitindo que o banco de dados seja adaptado para casos de uso específicos. Essa capacidade de extensão é suportada por uma comunidade ativa e vibrante que continuamente desenvolve novas extensões e melhorias. Ferramentas populares como PostGIS, que adiciona suporte a dados geoespaciais, e PL/pgSQL, uma linguagem procedural similar ao PL/SQL do Oracle, são exemplos de como a comunidade enriquece o ecossistema do PostgreSQL.

A confiabilidade e a integridade dos dados são também pilares fundamentais do PostgreSQL. Ele oferece um modelo de transações ACID (Atomicidade, Consistência, Isolamento e Durabilidade), garantindo que todas as operações sejam completadas com sucesso e que o banco de dados permaneça em um estado consistente mesmo em casos de falha. Além disso, o PostgreSQL implementa controle de concorrência multiversionamento (MVCC), permitindo que múltiplas operações de leitura e escrita ocorram simultaneamente sem conflitos, o que melhora significativamente o desempenho em ambientes de alta concorrência.

Finalmente, o PostgreSQL é uma escolha atraente devido ao seu modelo de licenciamento de código aberto sob a licença PostgreSQL, uma licença permissiva que permite o uso, modificação e distribuição do software sem restrições significativas. Isso não apenas reduz os custos para empresas, mas também proporciona maior controle e flexibilidade sobre suas implementações de banco de dados. Grandes empresas, incluindo Apple, Fujitsu e Cisco, utilizam PostgreSQL em suas operações diárias, validando ainda mais sua confiabilidade e eficácia.

3.2.5 DRAW.IO

O draw.io é uma ferramenta online de diagramação e criação de fluxogramas,

lançada em 2012, que permite aos usuários criar diagramas profissionais de forma intuitiva e colaborativa, sem a necessidade de instalar software adicional. Com draw.io, é possível acessar uma ampla variedade de elementos e formas pré-definidas, incluindo símbolos para diagramas de fluxo, organogramas, wireframes e diagramas UML. Além disso, o sistema permite importar e exportar diagramas em diferentes formatos, como XML, PNG e PDF, facilitando o compartilhamento e a colaboração com outras pessoas.

A natureza online do draw.io, aliada à sua integração com serviços de armazenamento em nuvem, torna-o uma opção conveniente para a criação e compartilhamento de diagramas em diferentes cenários e dispositivos. Os usuários podem acessar seus projetos de qualquer lugar, colaborar em tempo real com colegas e compartilhar facilmente os resultados finais. Isso é particularmente útil para equipes distribuídas geograficamente que precisam trabalhar juntas em tempo real.

A flexibilidade e robustez do draw.io são complementadas por sua acessibilidade. Sendo uma ferramenta gratuita e de código aberto, o draw.io é uma opção atraente para indivíduos e organizações que procuram uma solução eficaz para a visualização e documentação de processos e ideias. Sua interface amigável e a vasta gama de funcionalidades permitem que usuários de todos os níveis de habilidade criem diagramas detalhados e precisos com facilidade.

3.2.6 EXCALIDRAW

Excalidraw é uma ferramenta online de diagramação e esboço que se destaca por sua simplicidade e versatilidade. Lançada inicialmente como um projeto de código aberto, ela permite que usuários criem diagramas e ilustrações de maneira intuitiva e colaborativa, sem a necessidade de instalação de software adicional. Sua interface limpa e minimalista facilita a criação rápida de esboços, fluxogramas, wireframes e outros tipos de diagramas, tornando-a uma escolha popular entre designers, desenvolvedores e equipes de projeto.

Um dos principais atrativos do Excalidraw é a sua abordagem de desenho à mão livre, que dá aos diagramas um visual natural e despretensioso. Isso é especialmente útil em fases iniciais de brainstorming e planejamento, onde a

formalidade excessiva pode ser uma barreira à criatividade. Além disso, Excalidraw suporta colaboração em tempo real, permitindo que várias pessoas trabalhem juntas em um mesmo projeto, o que é essencial para equipes distribuídas e projetos colaborativos.

A integração com plataformas de armazenamento em nuvem como Google Drive e o suporte para exportação de diagramas em formatos como PNG e SVG ampliam a utilidade do Excalidraw, facilitando o compartilhamento e a incorporação de diagramas em outras ferramentas e documentos. A combinação de simplicidade, flexibilidade e colaboração faz do Excalidraw uma ferramenta poderosa para qualquer pessoa que precise criar e compartilhar visualizações claras e eficazes.

4 APRESENTAÇÃO DO SISTEMA

A plataforma SQLab surge como uma resposta à necessidade de proporcionar aos estudantes de tecnologia um ambiente educacional dinâmico e relevante, onde possam aprender os fundamentos teóricos da SQL e aplicá-los de forma prática em cenários simulados e reais. O SQLab foi concebido com a flexibilidade em mente, permitindo que os alunos avancem no aprendizado de acordo com seu ritmo e necessidades específicas, com suporte integrado para ajudar na superação de dificuldades individuais.

O ponto focal do SQLab é a implementação prática de atividades de SQL pelos alunos. Professores podem criar diversas perguntas, incluindo questões de múltipla escolha, respostas dissertativas e, principalmente, perguntas específicas de SQL onde os alunos devem escrever e executar código SQL para obter resultados correspondentes ao esperado. Este aspecto prático é fundamental para que os alunos desenvolvam habilidades aplicáveis no mundo real.

Além de criar as perguntas, os professores podem definir respostas e resultados esperados, possibilitando a correção automática das atividades. O sistema permite que os professores forneçam feedback detalhado, apontando melhorias e corrigindo erros, o que facilita um aprendizado mais direcionado e eficaz.

SQLab oferece suporte para pontuação e gamificação, incentivando os alunos a se engajarem mais profundamente nas atividades através de um sistema de pontos que recompensa a participação e o desempenho. Os alunos podem debater com seus colegas de turma em fóruns de discussão, promovendo um ambiente colaborativo e de troca de conhecimento.

Para apoiar ainda mais o aprendizado, a plataforma oferece dicas que os alunos podem acessar ao responder as atividades. Essas dicas são projetadas para guiar os estudantes sem fornecer respostas diretas, incentivando a resolução de problemas de maneira autônoma. Também incluso e subentendido funcionalidades de cadastro de alunos e turmas, realizados pelos professores, login, funcionalidades para o acompanhamento dos desempenhos, pela parte dos alunos dos professores.

Por fim, a implementação do SQLab não se limita apenas ao ambiente acadêmico; seu impacto potencial se estende à capacidade dos alunos de se tornarem

profissionais bem-sucedidos e adaptáveis no mercado de trabalho digital. Boulos et al. (2020) destacam a importância de plataformas educacionais que preparem os alunos para os desafios práticos e inovadores que enfrentarão em suas carreiras futuras, como é o caso da SQLab ao promover uma educação em SQL alinhada com as demandas da indústria de tecnologia.

4.1 MODELAGEM DO PROJETO

4.1.1 DIAGRAMA DE CASO DE USO

O diagrama de casos de uso é essencial para a compreensão dos requisitos funcionais de um sistema, representando as interações entre atores e os diferentes casos de uso que o sistema deve suportar. Ele proporciona uma visão clara das funcionalidades esperadas pelo usuário final, facilitando o alinhamento entre desenvolvedores e stakeholders. Para uma visualização detalhada deste diagrama, consulte o apêndice A. As especificações completas de cada caso de uso estão disponíveis no APÊNDICE C.

4.1.2 TABELA DE REQUISITOS

A tabela de requisitos é uma ferramenta fundamental na engenharia de requisitos, onde são listadas e detalhadas todas as funcionalidades e características que o sistema deve ter para atender às necessidades dos usuários e stakeholders. Ela proporciona uma base clara e estruturada para o desenvolvimento do sistema, garantindo que todas as especificações sejam consideradas e implementadas de maneira adequada. Consulte o APÊNDICE B para uma visão completa dos requisitos detalhados do sistema.

4.1.3 DIAGRAMA DE CLASSES

O diagrama de classes descreve a estrutura estática do sistema, mostrando as classes, seus atributos, métodos e relacionamentos. Ele é fundamental para o

projeto orientado a objetos, ajudando a definir a arquitetura do software e a visualizar a organização das entidades no sistema. Para uma representação completa deste diagrama, consulte o APÊNDICE D.

4.1.4 DIAGRAMA DE SEQUÊNCIA

O diagrama de sequência ilustra a interação entre objetos em uma sequência específica de eventos, mostrando como os objetos colaboram para realizar uma funcionalidade específica do sistema. Ele é crucial para capturar o comportamento dinâmico do sistema, facilitando o entendimento do fluxo de informações e controle entre os objetos. Para uma visualização detalhada deste diagrama, consulte o APÊNDICE E.

4.2 ARQUITETURA DO SISTEMA

Para o sistema SQLab, a arquitetura está sendo meticulosamente planejada para garantir eficiência, escalabilidade e segurança, fundamentais para suportar suas funcionalidades educacionais avançadas. A abordagem cliente-servidor adotada se divide entre o front end desenvolvido em React e o back end em Laravel, integrados por meio de uma arquitetura de requisições HTTP RESTful.

No front end, o React será implementado para oferecer uma interface de usuário responsiva e dinâmica. A escolha por React não apenas facilita a criação de componentes reutilizáveis e a gestão de estados complexos com Redux, mas também proporciona uma experiência de usuário fluída ao permitir a edição de consultas SQL em tempo real e a visualização imediata dos resultados. A interface intuitiva do SQLab será projetada para guiar os usuários através das atividades de SQL de forma clara e eficiente.

Já no lado do servidor, o Laravel será utilizado para desenvolver o web service que gerencia a lógica de negócios do sistema. Este framework PHP oferece uma estrutura MVC robusta que organiza a aplicação em camadas de modelo, visualização e controle, promovendo assim a separação de responsabilidades e facilitando a manutenção do código. A escolha do Laravel também se deve à sua capacidade de

criar APIs RESTful de maneira eficiente, garantindo uma comunicação segura e escalável entre o front end e o banco de dados PostgreSQL.

A integração entre front end e back end será realizada através de requisições HTTP RESTful, onde o front end enviará requisições para o back end para manipulação de dados, execução de consultas SQL e gestão de usuários. Essa arquitetura não só simplifica a comunicação entre os componentes do sistema, mas também permite que o SQLab ofereça uma experiência de usuário consistente e responsiva, essencial para o aprendizado eficaz de SQL.

Quanto ao banco de dados, o PostgreSQL foi selecionado devido à sua robustez, escalabilidade e suporte avançado para operações SQL complexas. Este SGBD será responsável por armazenar e gerenciar os dados dos usuários, atividades, resultados de consultas e outras informações críticas para o funcionamento do SQLab. A integração entre Laravel e PostgreSQL será facilitada pelo uso de ORM (Object-Relational Mapping), o que simplifica o acesso e a manipulação dos dados de forma eficiente e segura.

Por fim, a arquitetura do SQLab é desenhada para atender aos objetivos educacionais de proporcionar uma plataforma prática e interativa para o ensino e para garantir que o sistema seja escalável, seguro e de alto desempenho. Essa abordagem estruturada oferecerá uma experiência de usuário otimizada e adaptável às necessidades dos alunos e educadores.

4.3 FUNCIONALIDADES

A seção a seguir tem como principal objetivo apresentar as funcionalidades do sistema proposto de forma clara e objetiva. Por meio de um tutorial descritivo, serão apresentados todos os protótipos de telas, junto de uma breve explicação sobre cada componente e fluxos.

O SQLab é dividido em 2 perfis de usuários: professores e alunos, sendo que a primeira ação deve ser de um professor, com seu auto cadastro. A tela é composta de um formulário com quatro campos: “E-mail”, “Matrícula”, “Senha”, “Confirmação de senha”, seguidos de um botão “Registrar”. Ao preencher todos os campos e clicar no botão, o usuário é registrado no banco de dados e o sistema redireciona para a tela

principal.

Auto Cadastro

E-mail

Matrícula

Senha

Confirmação de senha

Registrar

UC01 – Auto cadastro de professor

Na tela principal, o professor consegue ter a visualização do desempenho de cada turma em cada lista de exercício cadastradas, com a porcentagem de quantos alunos já terminaram os exercícios propostos e qual o desempenho geral.

Exercicios Listas Turmas Logout

lista 1 turma A
75% dos alunos terminaram
<desempenho da turma>

lista 1 turma B
50% dos alunos terminaram
<desempenho da turma>

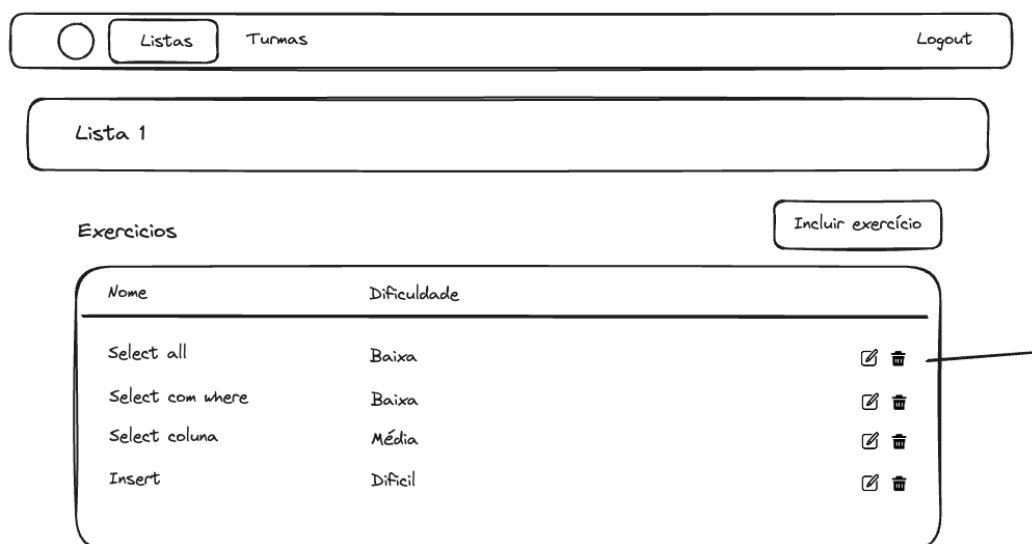
lista 2 turma C
25% dos alunos terminaram
<desempenho da turma>

UC18 – Visualizar relatório de desempenho

A partir da tela principal, é possível navegar para as três partes principais do sistema: tela de exercícios, tela de listas de exercícios e tela de turmas, que serão

descritas nos parágrafos a seguir.

Ao clicar no botão de “exercícios” na navegação superior, uma tela com um botão “Incluir exercício” e uma tabela com todos os exercícios cadastrados é aberta. Ao clicar no botão, a tela de cadastro de novo exercício é aberta. Na tabela, é possível ver cada nome, dificuldade e botões de ação: “editar” e “remover”.



UC10 – Listar exercícios (professor)

As telas de cadastro e edição de exercício seguem o mesmo padrão, sendo que quando a de edição é aberta, os dados registrados no banco de dados já aparecem preenchidos. A tela é composta por três abas: enunciado, ambiente e critérios. Sendo que a aba de enunciado é onde o professor registra qual o enunciado do exercício, na aba de ambiente é detalhado quais as tabelas e quais dados iniciais elas terão e na aba de critérios é onde são registrados as formas de avaliação do exercício.

Exercícios

Listas

Turmas

Logout

Enunciado

Ambiente

Crterios

Título

Enunciado Markdown

Voltar

Salvar

UC11 – Cadastrar exercício, aba de enunciado

Exercícios

Listas

Turmas

Logout

Enunciado

Ambiente

Crterios

Boot

CREATE Table blabla
CREATE Table blabla
CREATE Table blabla
CREATE Table blabla
ALTER TABLE blabla add column blabla

Preview

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
jondate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthymaintenance	numeric

UC11 – Cadastrar exercício, aba de ambiente

UC11 – Cadastrar exercício, aba de critérios

Na tela de listagem também é possível remover um exercício, depois do clique no botão “remover”, o seguinte modal é aberto, confirmando a exclusão.

UC13 – Remover exercício

Na tela de detalhe de exercício é possível listar, cadastrar e remover dicas de um exercício, conforme o professor ache necessário.



UC14 – Listar dicas



UC15 – Remover dicas

Cadastrar dica

Descrição

Cancelar
Cadastrar

UC32 – Cadastrar dicas

Voltando para as telas da página principal, o professor pode visualizar as turmas, clicando no botão “turmas” no menu superior de navegação. É aberta uma tela com uma tabela com “disciplina”, “quantidade de alunos”, “ano” e “semestre”. Junto com os botões de edição/remoção de cada turma, além de dois botões superiores: “incluir turma” para cadastrar novas turmas e “convidar alunos” para registro de novos alunos.

Exercícios
Listas
Turmas
Logout

Lista de turmas

Convidar alunos
Incluir turma

Disciplina	Qtd. Alunos	Ano	Semestre	
BD1	22	2024	1	
BD3	22	2024	1	
BD2	22	2024	1	
DAC	22	2024	1	

UC22 – Listar turmas

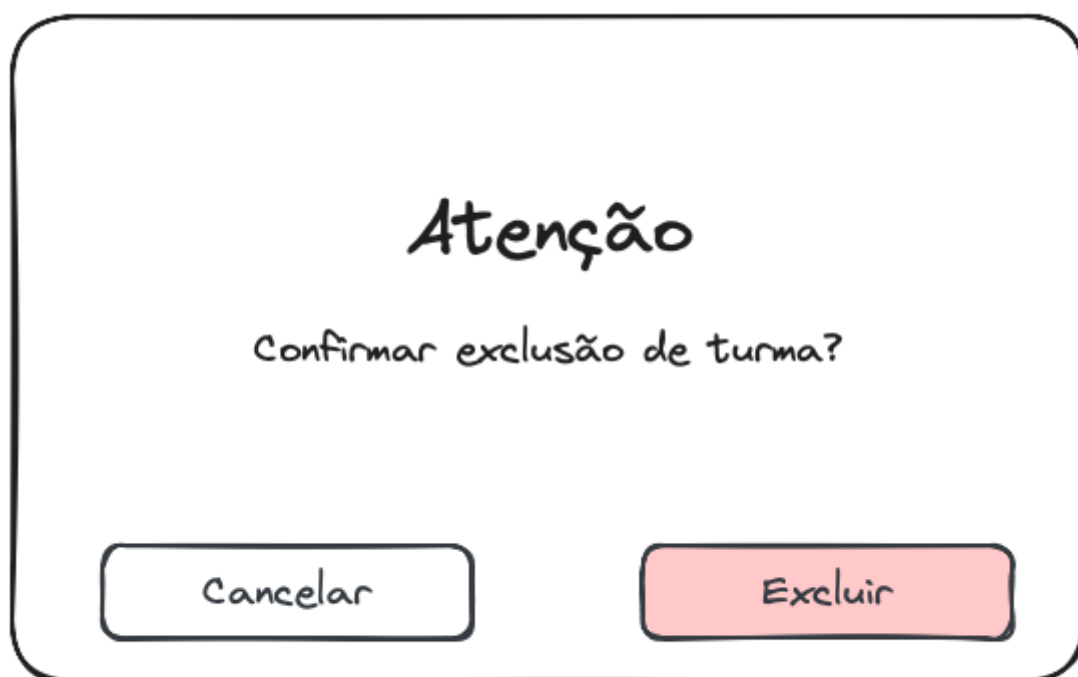
Clicando no botão “incluir turma”, é aberto um modal de cadastro. Com um formulário simples contendo “nome” e “ano”, além de dois botões “cadastrar” e “cancelar”.

The image shows a hand-drawn sketch of a modal form titled "Cadastrar Turma". The form is enclosed in a rounded rectangle. It contains two input fields: the first is labeled "Nome" and the second is labeled "Ano". Below the input fields are two buttons: "Cancelar" and "Cadastrar". The "Cadastrar" button is highlighted with a green background.

UC23 – Cadastrar turmas

Ao clicar no botão de edição da listagem, o mesmo modal de cadastrar turmas é aberto, porém as informações são preenchidas com o registro do banco de dados.

Para o botão de remover uma turma, é aberto um modal de confirmação, antes de efetivamente deletar o registro do banco de dados.



UC25 – Remover turma

Se o botão “convidar alunos” for selecionado, o seguinte modal é aberto, com dois botões principais: “carregar .csv” e “enviar convites”. Ao clicar no botão de CSV, o professor deve fazer o upload de um arquivo que esteja no formato especificado na descrição do modal. Depois de fazer o upload, o professor deve clicar no botão “enviar convites”, a partir disso, e-mails serão enviados para os alunos fazerem o registro de seus dados e senha, a fim de ganharem acesso ao sistema.

Convidar alunos

Adicione um arquivo CSV com os campos: GRR, email e turma no formato CSV

Carregar .CSV

Cancelar Enviar convites

UC26 – Convidar alunos

A última seção do perfil de professor é a de listas de exercícios, que nada mais são onde as turmas se relacionam com os exercícios. Ao clicar no botão “listas de exercício” no menu de navegação superior, é aberta a tela de listagem de listas de exercício. Com um botão superior de “incluir lista”, que abre a tela de cadastro de listas, e uma tabela com todas as listas de exercícios já cadastradas, com “nome”, “dificuldade”, “quantidade de exercícios” e os botões de “atribuir a turma”, “editar” e “excluir”.

☐

Listas

Turmas

Logout

Listas Exercicios

Incluir Lista

Nome	Dificuldade	Quantidade	
Lista 1	Baixa	3	<div>atribuir a turma</div> <div>✎</div> <div>🗑</div>
Lista 2	Baixa	5	<div>atribuir a turma</div> <div>✎</div> <div>🗑</div>

UC27 – Listar listas de exercício

Ao clicar no botão “incluir lista”, é aberto o modal com os campos de “nome” e “exercícios”, que vem preenchido com todos os exercícios cadastrados anteriormente, e o botão “cadastrar”. No campo de exercícios, devem ser selecionados todos os exercícios que vão compor a lista.

The image shows a hand-drawn sketch of a modal form titled "Nova lista". It contains two input fields: "Nome" and "Exercícios". The "Exercícios" field has a dropdown arrow. At the bottom, there are two buttons: "Cancelar" and "Cadastrar".

```
graph TD; Title[Nova lista]; Nome[Nome]; Exercicios[Exercícios]; Cancelar[Cancelar]; Cadastrar[Cadastrar];
```

UC28 – Cadastrar lista de exercício

Ao clicar no botão “editar” na listagem, o mesmo modal do cadastro é aberto, com a diferença é de que os campos aparecem populadas. Para a remoção de listas, o modal de confirmação é aberto com o botão “excluir”, confirmando a deleção do registro.



UC30 – Remover lista de exercício

Ao clicar no botão “atribuir a turma” na listagem, um modal é aberto para que o professor especifique a qual turma a lista será atribuída, ao clicar no botão “atribuir”, a lista fica disponível para alunos da turma resolver.

Atribuir a Turma

Turma

✓

Cancelar

Cadastrar

UC27 – Fluxo alternativo

Agora apresentando as telas do perfil de aluno, a primeira que o aluno deve acessar é a página de registro em uma turma, que é aberta através do convite que o professor enviou via e-mail.

Nessa tela, o aluno deve preencher os campos de “e-mail”, “grr”, “senha” e “confirmação de senha”, após isso, quando clicar no botão de “registrar”, uma conta será criada e o aluno redirecionado para sua página principal com uma sessão ativa.

Auto Cadastro

Turma

E-mail

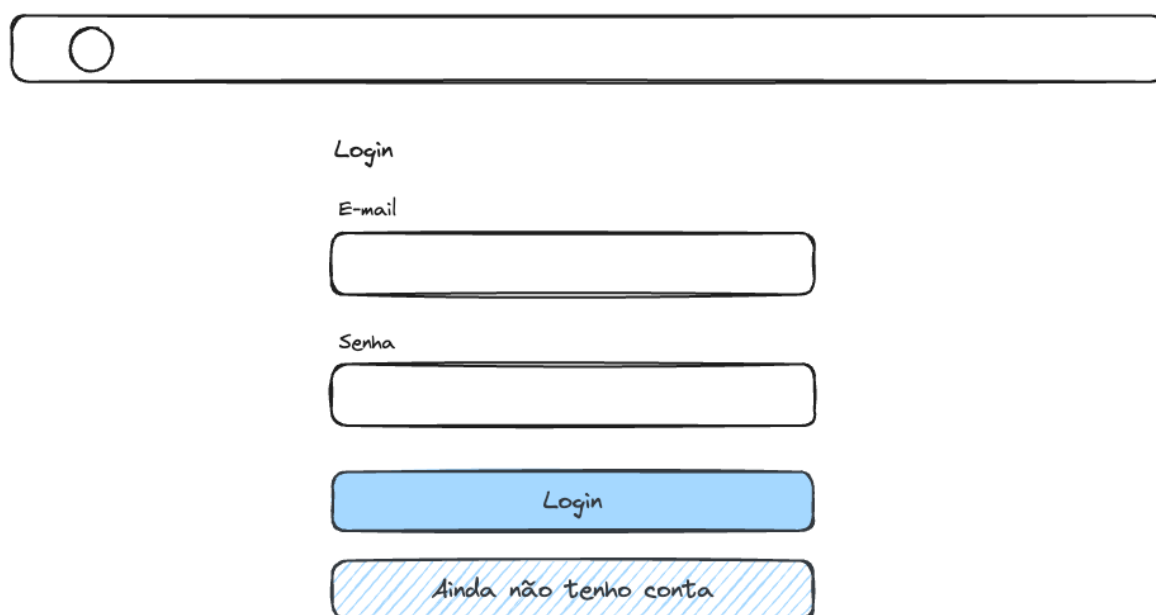
Matrícula

Senha

Confirmação de senha

UC04 – Inscrever em turma

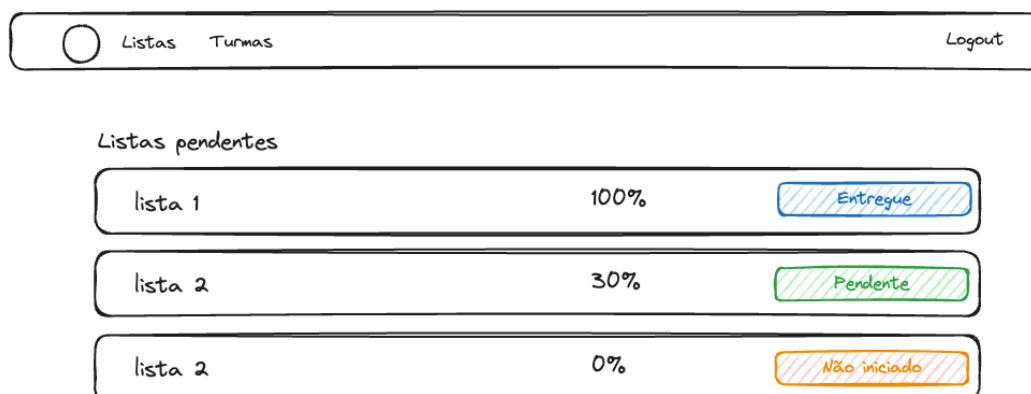
Uma outra forma de tanto o aluno como o professor entrar no sistema, é através do login, que possui um formulário com “e-mail” e “senha”, além de um botão de “login”.



A hand-drawn sketch of a login form. At the top is a horizontal bar with a circle on the left. Below it, the word "Login" is written. Then, "E-mail" is written above a rectangular input field. Below that, "Senha" is written above another rectangular input field. Under the password field is a solid blue button labeled "Login". At the bottom is a button with diagonal hatching labeled "Ainda não tenho conta".

UC02 - Login

Na página principal, o aluno deve visualizar as listas que o professor registrou em sua turma.




A hand-drawn sketch of a student exercise list interface. At the top is a horizontal bar with a circle on the left, followed by the text "Listas Turmas", and "Logout" on the right. Below this bar, the text "Listas pendentes" is written. There are three rows, each representing a list item in a rounded rectangular box. Each row contains the list name, a percentage, and a status button.

Nome da Lista	Porcentagem	Status
lista 1	100%	Entregue
lista 2	30%	Pendente
lista 2	0%	Não iniciado

UC31 – Listar listas de exercício do aluno

Ao clicar no nome de alguma lista que está pendente de entrega ou que ainda não foi iniciada, o sistema abre a tela de resolução da lista. Com uma lista de todos os exercícios daquela lista.


Listas Turmas
Logout

[< Voltar](#)
 Lista 1

Nome	Dificuldade	
Select all	Baixa	Resolvido
Select com where	Baixa	Resolver
Select coluna	Média	Resolver
Insert	Difícil	Resolvido

UC10 – Listar exercícios da lista (aluno)

Ao clicar no nome de algum exercício, a página de resolução de exercício é aberta. A parte superior guarda o nome do exercício atual, assim como uma navegação para voltar a listagem de exercícios ou prosseguir para o próximo. Logo abaixo, é apresentado o enunciado, que é cadastrado anteriormente pelo professor, caso alguma dica esteja atrelada a esse exercício, ela é apresentada abaixo do enunciado. Depois o sistema apresenta a seção de dados/estrutura, são duas abas, que variam de acordo com qual é selecionada, a aba de dados mostra em forma de tabela as informações populadas nas tabelas que são apresentadas na aba de estrutura, onde é mostrado um diagrama de entidade e relacionamento do banco. O campo de “solução” é onde o aluno efetivamente preenche a resposta do exercício. Ao clicar no botão “enviar”, o sistema valida os critérios de aceite, e valida se a resposta é adequada ao problema. Caso seja, o botão “resolvido” é apresentado e caso não, o botão “falha” é apresentado. Por último, o sistema mostra o retorno que a resposta do aluno teve.

☐ Exercícios
 Listas
 Turmas
 Logout

< Voltar

Lista 1

Histórico

Enviar >

Select all

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus quis sem nisi. Nulla lobortis porta dignissim. Nullam tempor quam lorem, at dictum neque euismod scelerisque. Sed id pulvinar elit, eu sodales odio. Phasellus suscipit ornare neque, at feugiat ante aliquet vel. Praesent nisl nibh, pretium vitae mauris eget, euismod dapibus turpis. Sed dignissim venenatis finibus. Cras varius tincidunt velit, sed suscipit diam luctus quis. Aenean tempor, metus eu bibendum pharetra, eros purus commodo mi, et convallis nunc dui nec nisi. Praesent varius lacus ac turpis gravida laoreet vitae sed tortor. Fusce blandit vitae tellus vitae accumsan.

Dica 1

Dica 2

Dados

Estrutura

Alunos	Turmas	Notas

Solução

SELECT * FROM turmas;

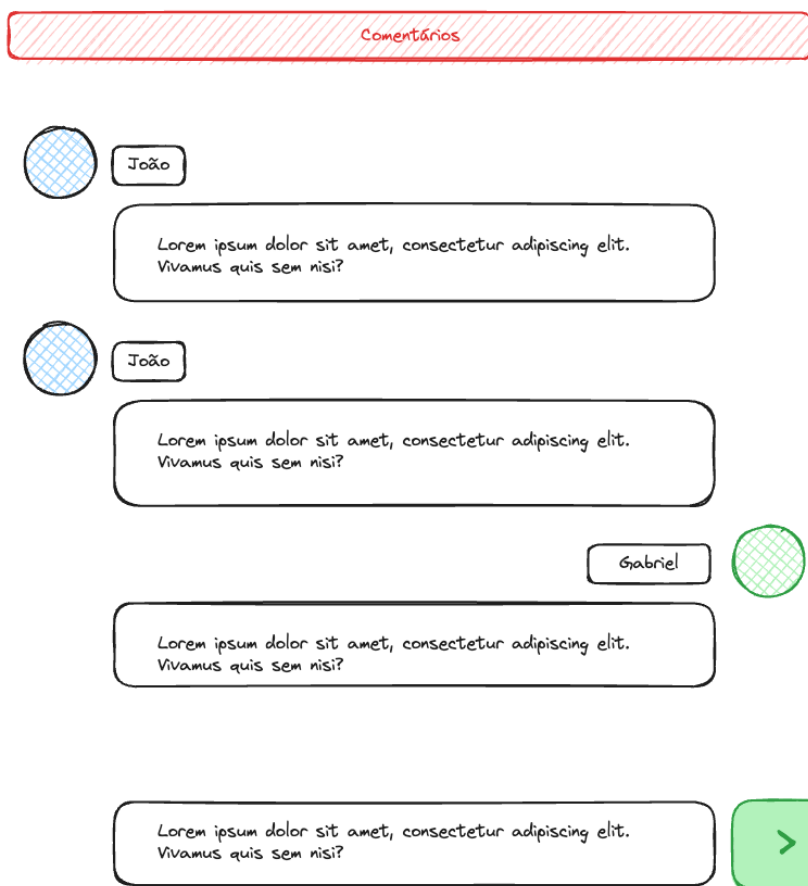
Critérios de aceite

Utilizou o select	Ok	Resolvido
Valida alunos	Não retornou o aluno "João"	Falha

Retorno

UC08 – Resolver exercício

O aluno também tem acesso a seção de feedback (professor) e comentários (alunos) em cada exercício, para ter uma interação com seus colegas. O feedback é o que o professor comentou por aluno sobre a resposta, já os comentários, é a interação entre alunos que pode ocorrer em formato de forum.



UC09 – Ler comentários/UC06 - Editar comentários

Ao clicar no botão “histórico”, o aluno pode visualizar o histórico de tentativas de resolução.

 Exercícios Listas Turmas Logout

< Voltar

Tentativas: Select all

Solução	Data/hora	
Select * rom dev;	25/05/2024 12:30	Falha
Select * from dev;	25/05/2024 12:31	Falha
Select all from dev;	25/05/2024 12:32	Falha
Select * from alunos;	25/05/2024 12:33	Resolvido

UC07 – Visualizar tentativas

Por fim, o aluno também tem acesso as informações que estão cadastradas, acessando o menu de “perfil”.

 Exercícios Listas Turmas Logout

Perfil

E-mail

helloworld@hotmail.com

GRR

GRR20202020

Nome

Hello World da Silva

UC05 – Visualizar perfil

5 REFERÊNCIAS

Clark, R. E. (1994). **Media will never influence learning**. Educational Technology Research and Development, 42(2), 21-29.

Kirschner, P. A., Sweller, J., & Clark, R. E. (2006). **Why minimal guidance during instruction does not work: An analysis of the failure of constructivist, discovery, problem-based, experiential, and inquiry-based teaching**. Educational Psychologist, 41(2), 75-86.

Siemens, G. (2005). **Connectivism: A learning theory for the digital age**. International Journal of Instructional Technology and Distance Learning, 2(1), 3-10.

Boulos, M. N. K., et al. (2020). **Education and training in health informatics and digital health: A global perspective, challenges, and opportunities reviewed**. Yearbook of Medical Informatics, 29(1), 115-123.

Fisher, K., & Baird, D. E. (2011). **Online learning design that fosters student support, self-regulation, and retention**. Campus-Wide Information Systems, 28(5), 329-340.

Hattie, J., & Timperley, H. (2007). **The power of feedback**. Review of Educational Research, 77(1), 81-112.

Reimers, F., & Schleicher, A. (Eds.). (2020). **Teaching for Tomorrow: Learning from the COVID-19 Pandemic**. OECD Publishing.

Jonassen, D. H. (1999). **Constructivist Learning Environments on the Web: Engaging Students in Meaningful Learning**. Educational Technology, 39(3), 35-38.

Hmelo-Silver, C. E. (2004). **Problem-Based Learning: What and How Do Students Learn?**. Educational Psychology Review, 16(3), 235-266.

Prince, M. (2004). **Does Active Learning Work? A Review of the Research**. Journal of Engineering Education, 93(3), 223-231.

Booch, G., Rumbaugh, J., & Jacobson, I. (2005). **The Unified Modeling Language User Guide**. Addison-Wesley.

Fowler, M. (2004). **UML Distilled: A Brief Guide to the Standard Object Modeling Language**. Addison-Wesley.

6 APÊNDICES

APÊNDICE A – DIAGRAMA DE CASO DE USO



APÊNDICE B – TABELA DE REQUISITOS

Identificador	Requisito	Classificação
RF001	O sistema deve permitir que o usuário se identifique através de login.	Geral
RF002	O sistema deve permitir que o usuário visualize o ranking.	Geral
RF003	O sistema deve permitir o auto cadastro de professores.	Professor
RF004	O sistema deve permitir que o professor gerencie alunos.	Professor
RF005	O sistema deve permitir que o professor gerencie turmas.	Professor
RF006	O sistema deve permitir que o professor gerencie listas de exercícios.	Professor
RF007	O sistema deve permitir que o professor visualize relatórios de desempenho dos alunos	Professor
RF008	O sistema deve permitir que o professor forneça feedback aos alunos.	Professor
RF009	O sistema deve permitir que o professor visualize as tentativas de exercícios dos alunos.	Professor
RF010	O sistema deve permitir que o professor gerencie dicas.	Professor
RF011	O sistema deve permitir que o professor gerencie exercícios.	Professor

RF012	O sistema deve permitir que o professor remova comentários.	Professor
RF013	O sistema deve permitir que o professor leia comentários.	Professor
RF014	O sistema deve permitir que o professor atribua listas de exercício a turmas.	Professor
RF015	O sistema deve permitir que o aluno se inscreva em turmas.	Aluno
RF016	O sistema deve permitir que o aluno visualize seu perfil.	Aluno
RF017	O sistema deve permitir que o aluno edite seus comentários.	Aluno
RF018	O sistema deve permitir que o aluno visualize o histórico de suas tentativas.	Aluno
RF019	O sistema deve permitir que o aluno resolva exercícios.	Aluno
RF020	O sistema deve permitir que o aluno cadastre comentários.	Aluno
RF021	O sistema deve permitir que o aluno leia comentários de exercícios.	Aluno

APÊNDICE C – ESPECIFICAÇÕES DE CASO DE USO

UC01 Auto cadastro

Caso de Uso		Auto cadastro
Ator Principal	Professor	
Ator secundário	-	
Resumo	Permite que professor faça o auto cadastro	
Pré-condições	-	
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa a página de autocadastro	
2		Exibe a tela com o formulário: Campos: <ul style="list-style-type: none">– email;– senha;– matrícula; Botões: <ul style="list-style-type: none">– registrar;
3	Preenche os campos exigidos e clica em “registrar”	
4		Faz a validação das informações fornecidas e armazena no banco de dados
5		Salva informações na sessão e redireciona o professor para a página principal.

6	Caso de Uso encerra	
Fluxo de Exceção		
E (2.1) dados obrigatórios não enviados		
Passo	Ação do Ator	Ação do Sistema
1		Informa que os dados obrigatórios não foram enviados
2	Caso de uso retorna para o passo 3 do fluxo principal	

UC02 - Login

Caso de Uso		Login
Ator Principal		Usuário
Ator secundário		-
Resumo		Permite que um usuário faça a autenticação no sistema
Pré-condições		O usuário precisa ter um registro de cadastro
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa a página de login	
2		Exibe a tela com o formulário: Campos: email; senha; Botões: login; "ainda não tenho conta"
3	Preenche os campos exigidos e clica no botão “login”	
4		Faz a validação das informações fornecidas

5		Salva informações na sessão e redireciona o usuário para a página principal
6	Caso de Uso encerra	
Fluxo de Exceção		
E (2.1) email ou senha incorretos		
Passo	Ação do Ator	Ação do Sistema
1		Informa que os dados estão errados
2	Caso de uso retorna para o passo 3 do fluxo principal	
E (2.2) dados obrigatórios não enviados		
Passo	Ação do Ator	Ação do Sistema
1		Informa que os dados obrigatórios não foram enviados
2	Caso de uso retorna para o passo 3 do fluxo principal	

UC03 - Visualizar ranking

Caso de Uso	Visualizar ranking
Ator Principal	Usuário
Ator secundário	-
Resumo	Permite que usuários visualizem o ranking dos

	alunos	
Pré-condições	Usuário precisa estar autenticado e pelo menos uma turma estar cadastrada	
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Seleciona o menu “ranking”	
2		Exibe a tela com as turmas que o usuário pode ler
3	Seleciona de qual turma deseja acessar	
4		Exibe a tela com o ranking da turma: Tabela: Nome do aluno Posição geral Quantos acertos teve
5	Caso de Uso encerra	

UC04 - Inscrever em turma

Caso de Uso		Inscrever em turma
Ator Principal	Aluno	
Ator secundário	-	
Resumo	Permite que o aluno se inscreva em uma turma	
Pré-condições	O aluno deve ter acessado via link de cadastro enviado anteriormente pelo professor	
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Abre o link de cadastro	
2		Exibe um formulário confirmando a inscrição: Campos: GRR; email; senha; Botões: Salvar
3		Salva informações na sessão e redireciona o aluno para a página principal
4	Caso de Uso encerra	

Fluxo de Exceção		
E (2.1) GRR não corresponde ao do link		
Passo	Ação do Ator	Ação do Sistema
1		Informa que o GRR informado no formulário não corresponde ao do link enviado.
2	Caso de uso retorna para o passo 2 do fluxo principal	
E (2.2) dados obrigatórios não enviados		
Passo	Ação do Ator	Ação do Sistema
1		Informa que os dados obrigatórios não foram enviados
2	Caso de uso retorna para o passo 2 do fluxo principal	

UC05 - Visualizar perfil

Caso de Uso		Visualizar perfil
Ator Principal		Aluno
Ator secundário		-
Resumo		Permite que o aluno veja suas informações cadastradas
Pré-condições		Aluno precisa estar autenticado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa a página de “perfil”	
2		Carrega informações cadastradas: email; GRR; nome;
3	Caso de Uso encerra	

UC06 - Editar comentário

Caso de Uso		Editar comentário
Ator Principal		Aluno
Ator secundário		-
Resumo		Permite que o aluno edite seus comentários em um exercício
Pré-condições		Aluno precisa estar autenticado e com pelo menos um comentário cadastrado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa a página de um exercício	
2		Carrega a seção de comentários do exercício, uma lista com: Campos: nome do aluno; comentário em texto; data do comentário; Botões: editar;
3	Clica no botão de “editar” do comentário desejado	
4		Troca o campo de texto para campo editável:

		comentário em texto; Exibe um botão para salvar a ação: salvar;
5	Altera o conteúdo do comentário e clica em “salvar”	
6		As informações são salvas na base de dados e o campo volta para exibição normal
7	Caso de Uso encerra	
Fluxo de Exceção		
E (2.1) comentário sem conteúdo		
Passo	Ação do Ator	Ação do Sistema
1		Informa que os dados obrigatórios não foram enviados
2	Caso de uso volta para o passo 5 do fluxo principal	

UC07 - Visualizar histórico de tentativas

Caso de Uso	Visualizar histórico de tentativas
Ator Principal	Aluno
Ator secundário	-
Resumo	Permite que o aluno visualize suas as tentativas de exercícios
Pré-condições	Aluno deve estar autenticado e com pelo menos um exercício resolvido

Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa página de “histórico”	
2		Carrega uma lista com todas as tentativas do aluno Campos: exercício; lista; nota; Botões: visualizar
3	Caso de Uso encerra	
Fluxo Alternativo		
A (2.1) Visualizar resposta		
Passo	Ação do Ator	Ação do Sistema
1	Clica no botão “visualizar”	
2		Sistema abre um modal detalhando qual foi a resposta do aluno no exercício
3	Caso de uso encerra	

UC08 - Resolver exercícios

Caso de Uso		Resolver exercícios
Ator Principal		Aluno
Ator secundário		-
Resumo		Permite que o aluno resolva exercícios de uma lista
Pré-condições		Aluno deve estar autenticado e com pelo menos uma lista de exercícios pendente de resolução
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa a página de um exercício de uma lista	
2		Carrega na página: Campos: enunciado; resultado desejado; resposta; Botões: enviar;
3	Preenche o campo “resposta” com a resposta do exercício e clica no botão “resolver”	
4		Salva informações na base de dados

4	Caso de Uso encerra	
Fluxo de Exceção		
E (2.1) o campo de resposta não foi preenchido		
Passo	Ação do Ator	Ação do Sistema
1		Exibe mensagem que o campo “resposta” deve ser preenchido
2	Caso de uso volta para o passo 3 do fluxo principal	
Fluxo Alternativo		
A (2.1) adicionar comentário		
Passo	Ação do Ator	Ação do Sistema
1	Acessa a seção de comentários	
2	Adiciona um comentário no exercício	
3		Salva informações no banco de dados
4	Caso de uso encerra	

UC09 - Ler comentários

Caso de Uso		Ler comentários
Ator Principal		Aluno
Ator secundário		-
Resumo		Permite que o aluno leia os comentários de um exercício
Pré-condições		Aluno deve estar autenticado e com pelo menos um exercício cadastrado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa a seção de comentários na página de um exercício	
2		Carrega lista de comentários com os campos: nome do aluno; comentário; data de cadastro;
3	Caso de Uso encerra	
Fluxo Alternativo		
A (2.1) adicionar comentário		
Passo	Ação do Ator	Ação do Sistema

1	Acessa a seção de comentários	
2	Adiciona um comentário no exercício	
3		Salva informações no banco de dados
4	Caso de uso encerra	

UC10 - Listar exercícios

Caso de Uso		Listar exercícios
Ator Principal		Professor
Ator secundário		-
Resumo		Permite que o professor visualize uma lista de exercícios
Pré-condições		Professor deve estar autenticado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa página de “exercícios”	
2		Carrega uma tela com botão para cadastrar novo exercício e uma lista de contendo: Campos: nome; tipo do exercício; Botões: editar; remover;
3	Caso de Uso encerra	
Fluxo Alternativo		
A (2.1) Clica no botão “adicionar”		

Passo	Ação do Ator	Ação do Sistema
1		Abre tela de novo exercício
2	Caso de uso encerra	

UC11 - Cadastrar exercícios

Caso de Uso		Cadastrar exercícios
Ator Principal		Professor
Ator secundário		-
Resumo		Permite que o professor cadastre novos exercícios
Pré-condições		Professor deve estar autenticado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa página de cadastro exercício	
2		Carrega na tela um formulário Campos: enunciado; resultado esperado; modelagem do banco; dificuldade; Botões: cadastrar;
3	Clica no botão “cadastrar”	
4		Valida informações e salva registro no banco de dados

5	Caso de Uso encerra	
Fluxo de Exceção		
E (2.1) informações obrigatórias não enviadas		
Passo	Ação do Ator	Ação do Sistema
1		Exibe na tela a informação de que os campos obrigatórios não foram enviados
2	Caso de uso retorna para o passo 2 do fluxo principal	
Fluxo Alternativo		
A (2.1) Cadastrar dicas		
Passo	Ação do Ator	Ação do Sistema
1	Clica no botão “adicionar dica”	
2		Abre um modal com o campo de descrição e um botão “registrar”
3	Clica no botão “registrar”	
4		Salva informação no banco de dados
5	Caso de uso encerra	

UC12 - Editar exercícios

Caso de Uso		Editar exercícios
Ator Principal		Professor
Ator secundário		-
Resumo		Permite que o professor edite um exercício
Pré-condições		Professor deve estar autenticado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa página de edição de exercício	
2		Carrega na tela um formulário com informações do banco de dados Campos: enunciado; resultado esperado; modelagem do banco; Botões: salvar;
3	Clica no botão “salvar”	
4		Valida informações e salva registro no banco de dados

5	Caso de Uso encerra	
Fluxo de Exceção		
E (2.1) informações obrigatórias não enviadas		
Passo	Ação do Ator	Ação do Sistema
1		Exibe na tela a informação de que os campos obrigatórios não foram enviados
2	Caso de uso retorna para o passo 2 do fluxo principal	
Fluxo Alternativo		
A (2.1) Remover comentários		
Passo	Ação do Ator	Ação do Sistema
1	Acessa a listagem de comentários do exercício	
2		Lista os comentários cadastrados para aquele exercício Campos: nome do aluno; comentário; data de cadastro; Botões: excluir;
3	Clica no botão “excluir”	

4	Caso de uso encerra	
A (2.2) Editar dicas		
Passo	Ação do Ator	Ação do Sistema
1	Acessa a listagem de dicas do exercício e seleciona o botão “editar” de alguma	
2		Abre um modal com o campo de descrição e um botão “salvar”
3	Clica no botão “salvar”	
4		A informação é salva no banco de dados
5	Caso de uso encerra	
A (2.3) Cadastrar dicas		
Passo	Ação do Ator	Ação do Sistema
1	Acessa a listagem de dicas do exercício e seleciona o botão “cadastrar dica”	
2		Abre um modal com o campo de descrição e um botão “cadastrar”
3	Clica no botão “cadastrar”	
4		A informação é salva no banco de dados
5	Caso de uso encerra	

UC13 - Remover exercícios

Caso de Uso		Remover exercícios	
Ator Principal		Professor	
Ator secundário		-	
Resumo		Permite que o professor remova exercícios	
Pré-condições		Professor deve estar autenticado	
Fluxo Principal			
Passo	Ação do Ator		Ação do Sistema
1	Acessar listagem de exercícios		
2	Clicar no botão “remover” de algum registro		
3			Valida se exercício não pertence a nenhuma lista e exibe modal de confirmação da exclusão
4	Confirma a exclusão		
5			Registro é excluído da base de dados
6	Caso de Uso encerra		
Fluxo de Exceção			
E (2.1) O exercício excluído pertence a uma lista			

Passo	Ação do Ator	Ação do Sistema
1		Exibe a informação de que o exercício pertence a uma lista e não pode ser excluído enquanto ela existir
2	Caso de uso volta para passo 1 do fluxo principal	
Fluxo Alternativo		
A (2.1) Clica no botão de fechar o modal		
Passo	Ação do Ator	Ação do Sistema
1	Clica no botão para fechar o modal de confirmação	
2		Modal é fechado
3	Caso de uso encerra	

UC14 - Listar dicas

Caso de Uso	Listar dicas
Ator Principal	Professor
Ator secundário	-
Resumo	Permite que o professor visualize as dicas de um exercício
Pré-condições	Professor deve estar logado e com pelo menos um exercício cadastrado

Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessar seção “dicas” na tela de detalhe de um exercício	
2		Carrega lista com as dicas cadastradas no exercício, com os botões: editar; remover;
3	Caso de Uso encerra	

UC15 - Remover dicas

Caso de Uso		Remover dicas
Ator Principal		Professor
Ator secundário		-
Resumo		Permite que um professor remova dicas de um exercício
Pré-condições		Professor deve estar autenticado e na tela de detalhe de um exercício
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Clica no botão “remover” de alguma dica de um exercício	
2		Exibe modal de confirmação da exclusão com o botão “excluir”
3	Clica no botão “excluir”	
4		Registro é excluído da base de dados
5	Caso de Uso encerra	
Fluxo Alternativo		
A (2.1) Clica no botão de fechar o modal		

Passo	Ação do Ator	Ação do Sistema
1	Clica no botão para fechar o modal de confirmação	
2		Modal é fechado
3	Caso de uso encerra	

UC16 - Visualizar tentativas de exercícios

Caso de Uso		Visualizar tentativas de exercícios
Ator Principal		Professor
Ator secundário		-
Resumo		Permite que o professor visualize as tentativas de resolução dos exercícios
Pré-condições		Professor deve estar autenticado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa página de tentativas de resolução de exercícios	
2		Exibe uma listagem dos exercícios com tentativas Campos: nome do exercício; quantidade de tentativas; porcentagem de acerto;
3	Clica no nome de algum exercício da listagem	
4		Exibe listagem com detalhes das tentativas Campos:

		enunciado; resposta; feedback;
5	Caso de Uso encerra	

UC17 - Fornecer feedback

Caso de Uso		Fornecer feedback
Ator Principal		Professor
Ator secundário		-
Resumo		Permite que o professor forneça feedback nas tentativas de resolução
Pré-condições		Professor deve estar autenticado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa detalhe de tentativa de exercício	
2		Exibe as informações da tentativa Campos: enunciado (leitura); resposta (leitura); feedback (leitura e escrita);
3	Clica no campo de feedback	
4		O campo de feedback se torna editável e ao lado dele é exibido um botão “salvar”
5	Preenche o campo de feedback e clica no botão “salvar”	

6		Valida campo e salva informações no banco de dados
7	Caso de Uso encerra	
Fluxo de Exceção		
E (2.1) campo de feedback não preenchido		
Passo	Ação do Ator	Ação do Sistema
1		Informa que o campo de feedback é obrigatório
2	Caso de uso volta para passo 4 do fluxo principal	

UC18 - Visualizar relatório de desempenho

Caso de Uso		Visualizar relatório de desempenho
Ator Principal		Professor
Ator secundário		-
Resumo		Permite que o professor acesse uma tela com informações mais gerais sobre o desempenho dos alunos
Pré-condições		Professor deve estar autenticado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa página principal	
2		Exibe informações sobre listas de exercícios Campos: nome da lista; turma; porcentagem de acerto; número médio de tentativas/aluno; porcentagem da turma que já concluiu;
3	Caso de Uso encerra	

UC19 - Listar alunos

Caso de Uso		Listar alunos
Ator Principal		Professor
Ator secundário		-
Resumo		Permite que professor visualize os alunos registrados
Pré-condições		Professor deve estar autenticado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa tela de alunos	
2		Exibe uma listagem de alunos Campos: nome; GRR; turma;
3	Caso de Uso encerra	
Fluxo Alternativo		
A (2.1) Professor clica no botão de adicionar aluno		
Passo	Ação do Ator	Ação do Sistema
1	Clica no botão “cadastrar aluno”	

2		Abre tela de cadastro de aluno
3	Caso de uso encerra	

UC20 - Cadastrar alunos

Caso de Uso		Cadastrar alunos
Ator Principal		Professor
Ator secundário		-
Resumo		Permite que o professor cadastre novos alunos
Pré-condições		Professor deve estar autenticado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa tela de cadastro de alunos	
2		Exibe um formulário que recebe um arquivo CSV para registro dos alunos Campos: email; GRR; Botões: salvar;
3	Clica no botão “salvar”	
4		Valida informações e salva registros na base de dados
5	Caso de Uso encerra	

Fluxo de Exceção		
E (2.1) Arquivo CSV com informações faltando		
Passo	Ação do Ator	Ação do Sistema
1		Exibe modal informando que o formato dos dados no arquivo CSV contém dados inválidos
2	Caso de uso retorna para passo 2 do fluxo principal	
Fluxo Alternativo		
A (2.1) Professor adiciona aluno por aluno		
Passo	Ação do Ator	Ação do Sistema
1	Clica no botão “cadastrar aluno individualmente”	
2		Exibe um formulário para registro unitário do aluno Campos: email; GRR; Botões: salvar;
3	Clica no botão “salvar”	
4		Valida informações e

		salva registros na base de dados
5	Caso de uso encerra	

UC21 - Remover alunos

Caso de Uso		Remover alunos
Ator Principal	Professor	
Ator secundário	-	
Resumo	Permite que o professor delete alunos	
Pré-condições	Professor deve estar autenticado	
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Clica no botão “remover” em algum aluno na listagem de alunos	
2		Exibe modal de confirmação de exclusão, dizendo que todas as tentativas dos alunos serão perdidas Botões: confirmar; cancelar;
3	Clica no botão “confirmar”	
4		Informação é excluída da base de dados
5	Caso de Uso encerra	
Fluxo Alternativo		

A (2.1) Clica no botão de fechar o modal		
Passo	Ação do Ator	Ação do Sistema
1	Clica no botão para fechar o modal de confirmação	
2		Modal é fechado
3	Caso de uso encerra	

UC22 - Listar turmas

Caso de Uso		Listar turmas
Ator Principal		Professor
Ator secundário		-
Resumo		Permite que o professor visualize as turmas registradas
Pré-condições		Professor deve estar autenticado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa página de “turmas”	
2		Exibe uma listagem com as turmas registradas Campos: nome; ano; quantidade de alunos; Botões: editar; remover;
3	Caso de Uso encerra	
Fluxo Alternativo		
A (2.1) Clica no botão “cadastrar turma”		

Passo	Ação do Ator	Ação do Sistema
1	Clica no botão “cadastrar turma”	
2		Abre tela de registro de nova turma
3	Caso de uso encerra	

UC23 - Cadastrar turmas

Caso de Uso		Cadastrar turmas
Ator Principal		Professor
Ator secundário		-
Resumo		Permite que o professor cadastre uma nova turma
Pré-condições		Professor deve estar autenticado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa página de cadastro de nova turma	
2		Exibe um formulário com Campos: nome; ano; Botões: cadastrar;
3	Clica no botão “cadastrar”	
4		Valida informações e salva informações no banco de dados
5	Caso de Uso encerra	
Fluxo de Exceção		

E (2.1) campos obrigatórios não enviados		
Passo	Ação do Ator	Ação do Sistema
1		Exibe mensagem pedindo as informações obrigatórias
2	Caso de uso retorna para passo 2 do fluxo principal	

UC24 - Editar turmas

Caso de Uso		Editar turmas
Ator Principal		Professor
Ator secundário		-
Resumo		Permite que o professor edite turmas
Pré-condições		Professor deve estar autenticado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa página de detalhe de uma turma	
2		Carrega na tela um formulário com informações do banco de dados Campos: nome; ano; Botões: salvar; adicionar alunos;
3	Clica no botão “salvar”	
4		Valida informações e salva informações no banco de dados

5	Caso de Uso encerra	
Fluxo de Exceção		
E (2.1) campos obrigatórios não enviados		
Passo	Ação do Ator	Ação do Sistema
1		Exibe mensagem pedindo as informações obrigatórias
2	Caso de uso retorna para passo 2 do fluxo principal	
Fluxo Alternativo		
A (2.1) clica no botão “adicionar alunos”		
Passo	Ação do Ator	Ação do Sistema
1		Abre tela de enviar link de convite para alunos
2	Caso de uso encerra	

UC25 - Remover turmas

Caso de Uso	Remover turmas
Ator Principal	Professor
Ator secundário	-
Resumo	Permite que professor remove turmas cadastradas
Pré-condições	

Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Clica no botão “remover” em alguma turma na listagem de turmas	
2		Exibe modal de confirmação de exclusão, dizendo que todas as informações linkadas a turma serão perdidas Botões: confirmar; cancelar;
3	Clica no botão “confirmar”	
4		Informação é excluída da base de dados
5	Caso de Uso encerra	
Fluxo Alternativo		
A (2.1) Clica no botão de fechar o modal		
Passo	Ação do Ator	Ação do Sistema
1	Clica no botão para fechar o modal de confirmação	
2		Modal é fechado
3	Caso de uso encerra	

UC26 - Enviar link de convite para ingressar

Caso de Uso		Enviar link de convite para ingressar
Ator Principal		Professor
Ator secundário		-
Resumo		Permite que o professor envie um link de ingresso em turma para um aluno
Pré-condições		Professor precisa estar autenticado e pelo menos ter uma turma cadastrada
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Clicar no botão “convidar aluno”	
2		Exibe um formulário que recebe um arquivo CSV com Campos: GRR; email; turma; Botão: convidar;
3		Envia uma mensagem para o email recebido com o link de convite
4	Caso de Uso encerra	

Fluxo de Exceção		
E (2.1) campos GRR ou email inválidos		
Passo	Ação do Ator	Ação do Sistema
1		Informa que os dados são inválidos
2	Caso de uso retorna para passo 2 do fluxo principal	
E (2.2) campos não enviados		
Passo	Ação do Ator	
1		Informa que os dados precisam ser enviados
2	Caso de uso retorna para passo 2 do fluxo principal	
Fluxo Alternativo		
A (2.1) Professor adiciona aluno por aluno		
Passo	Ação do Ator	Ação do Sistema
1	Clica no botão “cadastrar aluno individualmente”	
2		Exibe um formulário para registro unitário do aluno Campos: email; GRR; Botões:

		salvar;
3	Clica no botão “salvar”	
4		Valida informações e salva registros na base de dados
5	Caso de uso encerra	

UC27 - Listar listas de exercícios

Caso de Uso		Listar listas de exercícios
Ator Principal		Professor
Ator secundário		-
Resumo		Permite que o professor cadastre listas de exercícios
Pré-condições		Professor deve estar autenticado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa página de “listas de exercícios”	
2		Exibe uma listagem das listas cadastradas Campos: nome; quantidade de exercícios; dificuldade; Botões: editar; remover; atribuir a turma;
4	Caso de Uso encerra	
Fluxo Alternativo		

A (2.1) Clica no botão “cadastrar lista”		
Passo	Ação do Ator	Ação do Sistema
1	Clica no botão “cadastrar lista”	
2		Abre tela de registro de nova lista
3	Caso de uso encerra	
A (2.2) Clica no botão “atribuir a turma”		
Passo	Ação do Ator	Ação do Sistema
1	Clica no botão “atribuir a turma”	
2		Abre um modal com um campo para o professor escolher a qual turma quer atribuir à lista. Botões: salvar;
3	Clica no botão “salvar”	
4		As informações são salvas no banco de dados.
5	Caso de uso encerra	

UC28 - Cadastrar listas de exercícios

Caso de Uso		Cadastrar listas de exercícios
Ator Principal		Professor
Ator secundário		-
Resumo		Permite que o professor cadastre uma lista de exercício
Pré-condições		Professor deve estar autenticado e pelo menos um exercício deve estar registrado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa página de cadastro de nova lista de exercícios	
2		Exibe um formulário com Campos: exercícios; Botões: cadastrar;
3	Clica no botão “cadastrar”	
4		Valida informações e salva informações no banco de dados
5	Caso de Uso encerra	

Fluxo de Exceção		
E (2.1) campos obrigatórios não enviados		
Passo	Ação do Ator	Ação do Sistema
1		Exibe mensagem pedindo as informações obrigatórias
2	Caso de uso retorna para passo 2 do fluxo principal	

UC29 - Editar listas de exercícios

Caso de Uso		Editar listas de exercícios
Ator Principal		Professor
Ator secundário		-
Resumo		Permite que o professor edite uma lista de exercício
Pré-condições		Professor deve estar autenticado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa página de detalhe de uma lista	
2		Carrega na tela um formulário com informações do banco de dados Campos: exercícios; Botões: salvar;
3	Clica no botão “salvar”	
4		Valida informações e salva informações no banco de dados
5	Caso de Uso encerra	

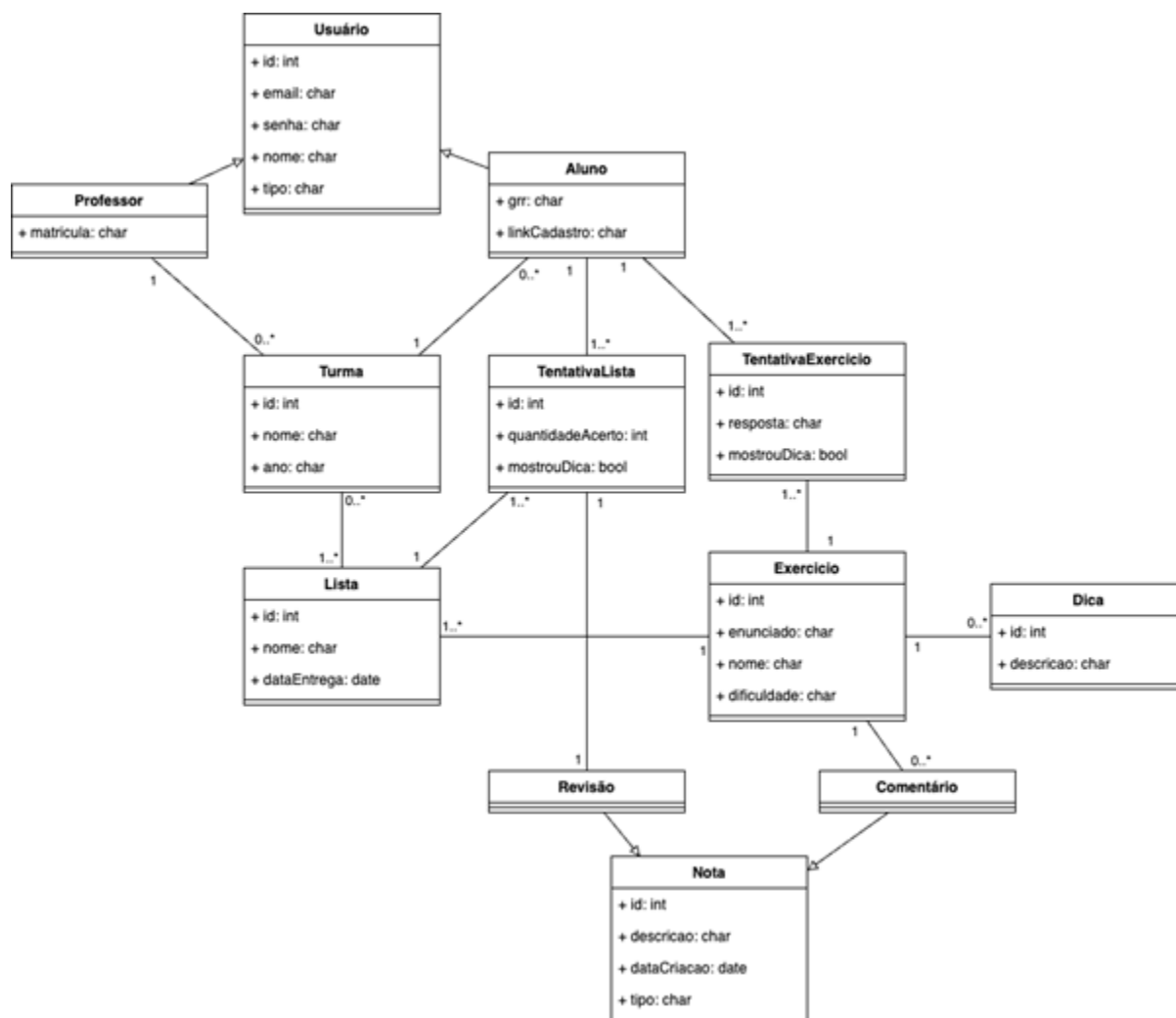
Fluxo de Exceção		
E (2.1) campos obrigatórios não enviados		
Passo	Ação do Ator	Ação do Sistema
1		Exibe mensagem pedindo as informações obrigatórias
2	Caso de uso retorna para passo 2 do fluxo principal	

UC30 - Remover listas de exercícios

Caso de Uso		Remover listas de exercícios
Ator Principal		Professor
Ator secundário		-
Resumo		Permite que o professor remova listas de exercícios
Pré-condições		Professor deve estar autenticado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Clica no botão “remover” em alguma lista da listagem de listas	
2		Exibe modal de confirmação de exclusão Botões: confirmar; cancelar;
3	Clica no botão “confirmar”	
4		Valida informações e deleta registro do banco de dados
5	Caso de Uso encerra	
Fluxo de Exceção		

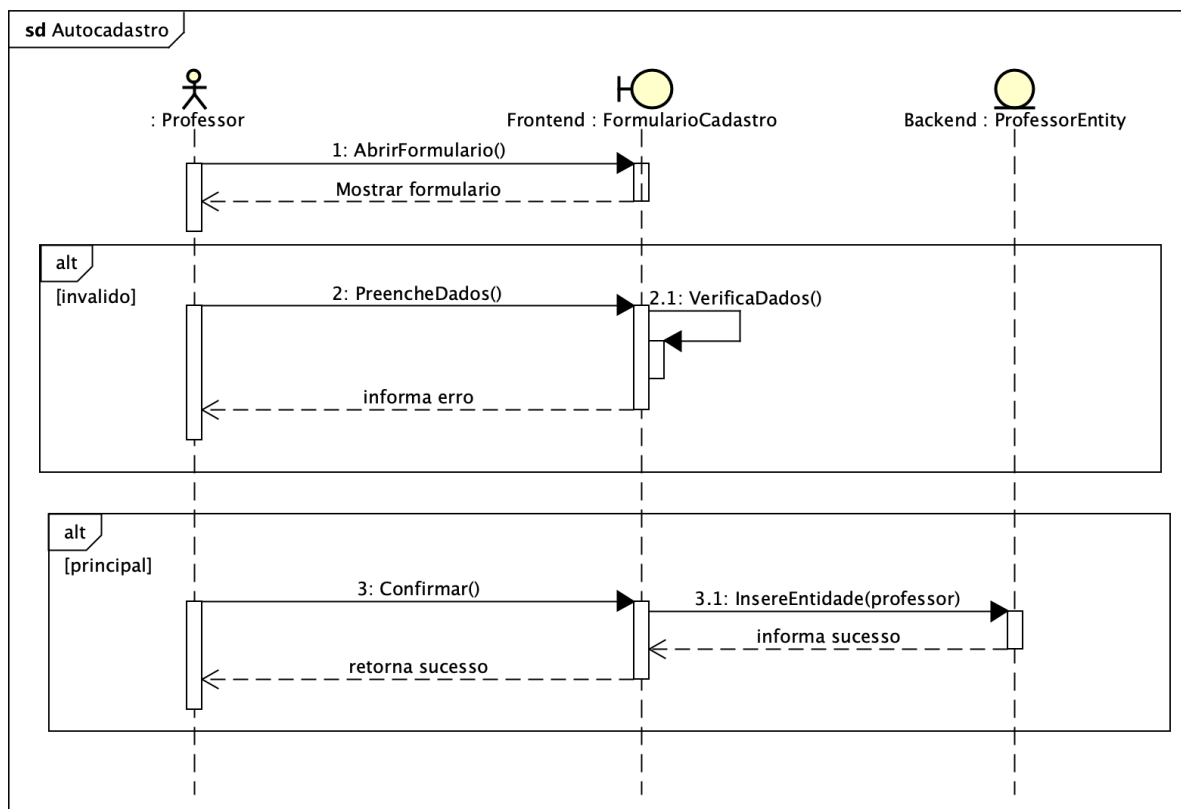
E (2.1) a lista já está atrelada a alguma turma		
Passo	Ação do Ator	Ação do Sistema
1		Exibe informação dizendo que a lista já está sendo usada por uma turma
2	Caso de Uso encerra	
Fluxo Alternativo		
A (2.1) Clica no botão de fechar o modal		
Passo	Ação do Ator	Ação do Sistema
1	Clica no botão para fechar o modal de confirmação	
2		Modal é fechado
3	Caso de uso encerra	

APÊNDICE D – DIAGRAMA DE CLASSE DE ANÁLISE

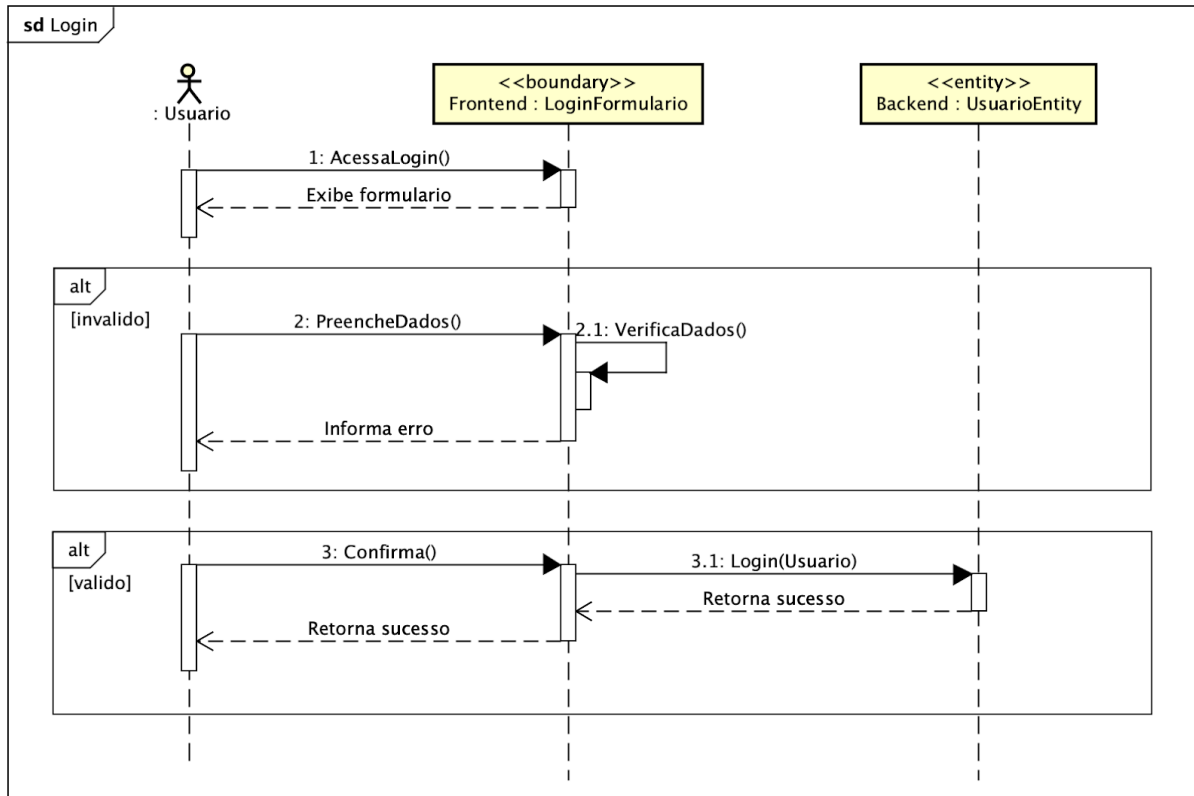


APÊNDICE E – DIAGRAMA DE SEQUÊNCIA DE ANÁLISE

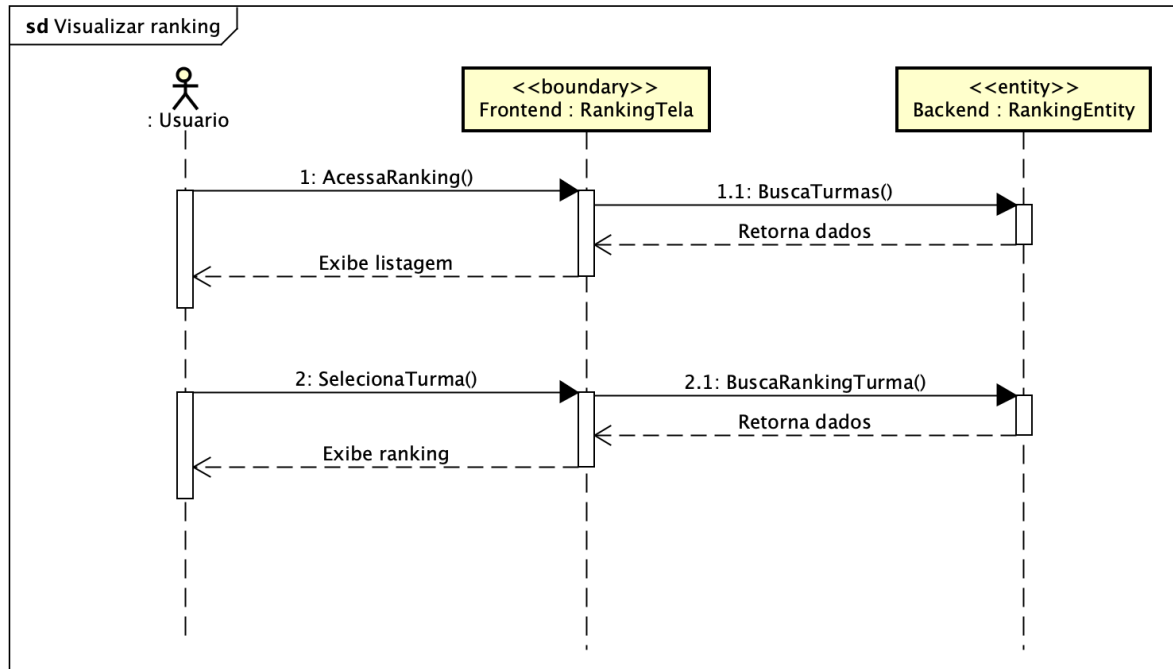
UC01 – Autocadastro



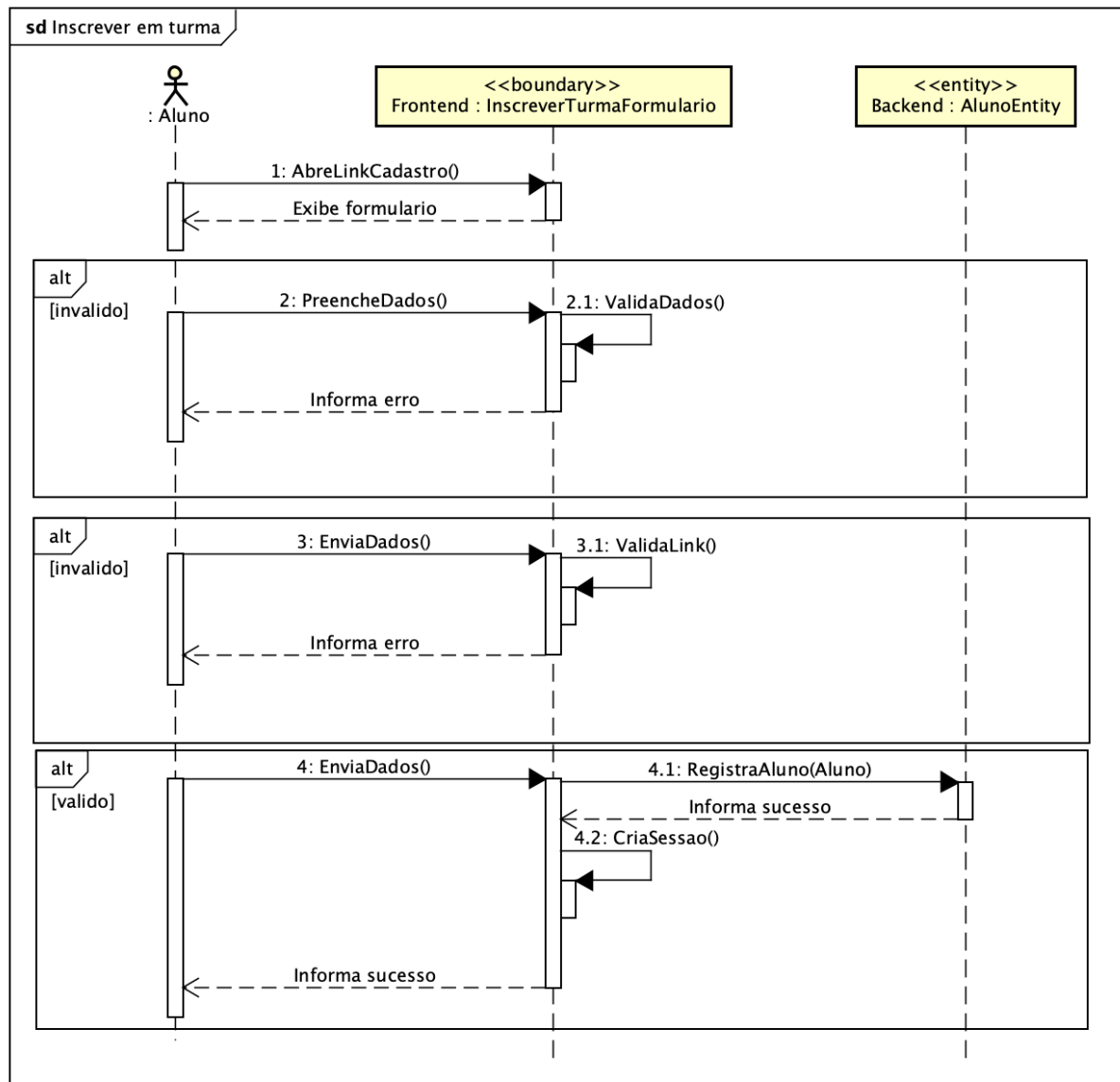
UC02 - Login



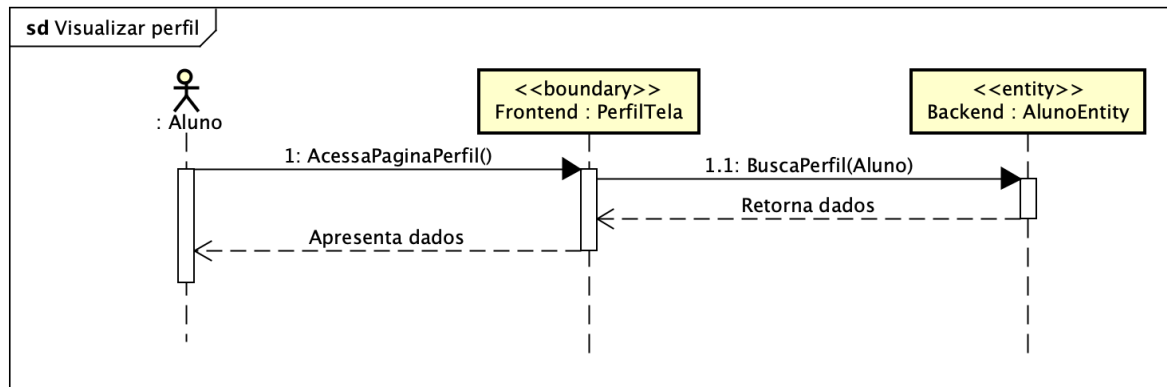
UC03 - Visualizar ranking



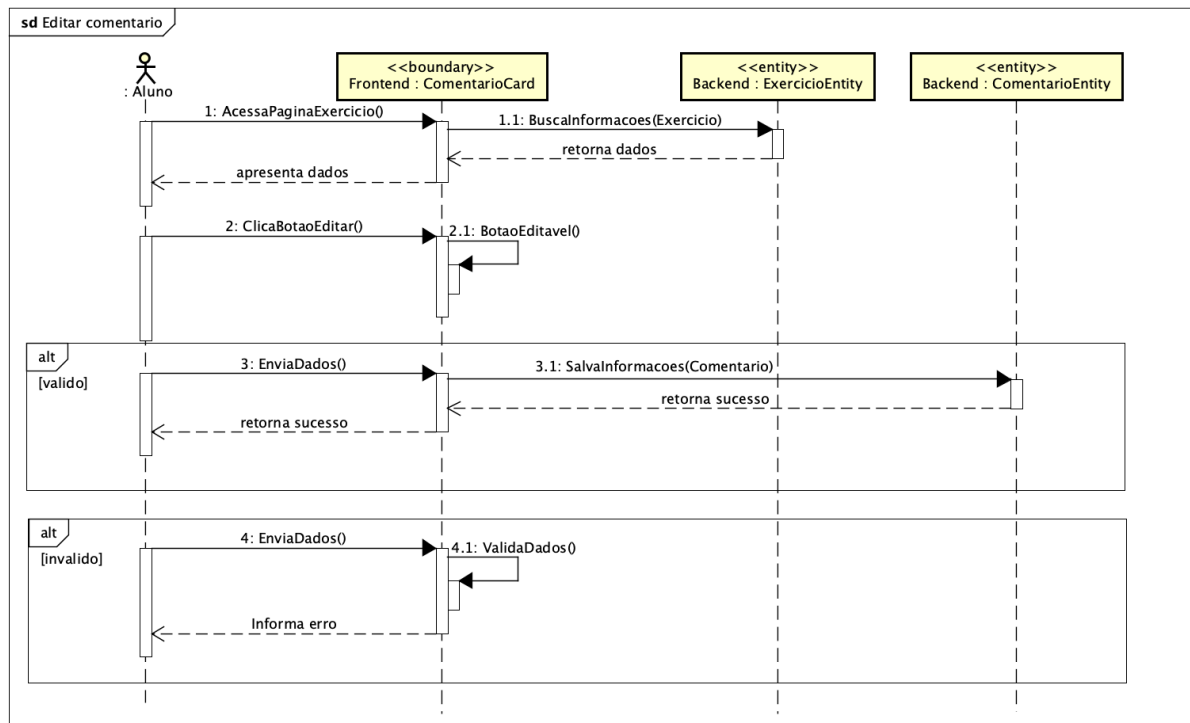
UC04 - Inscrever em turma



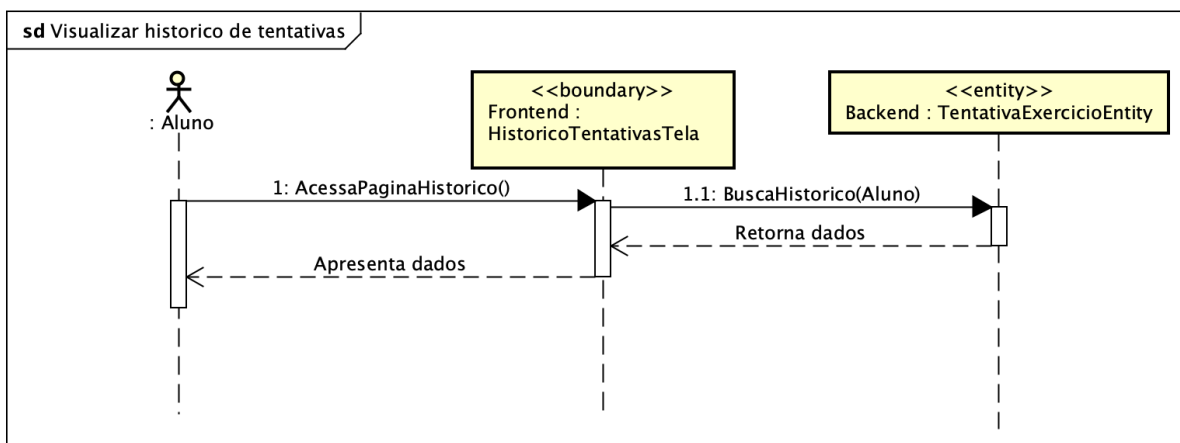
UC05 - Visualizar perfil



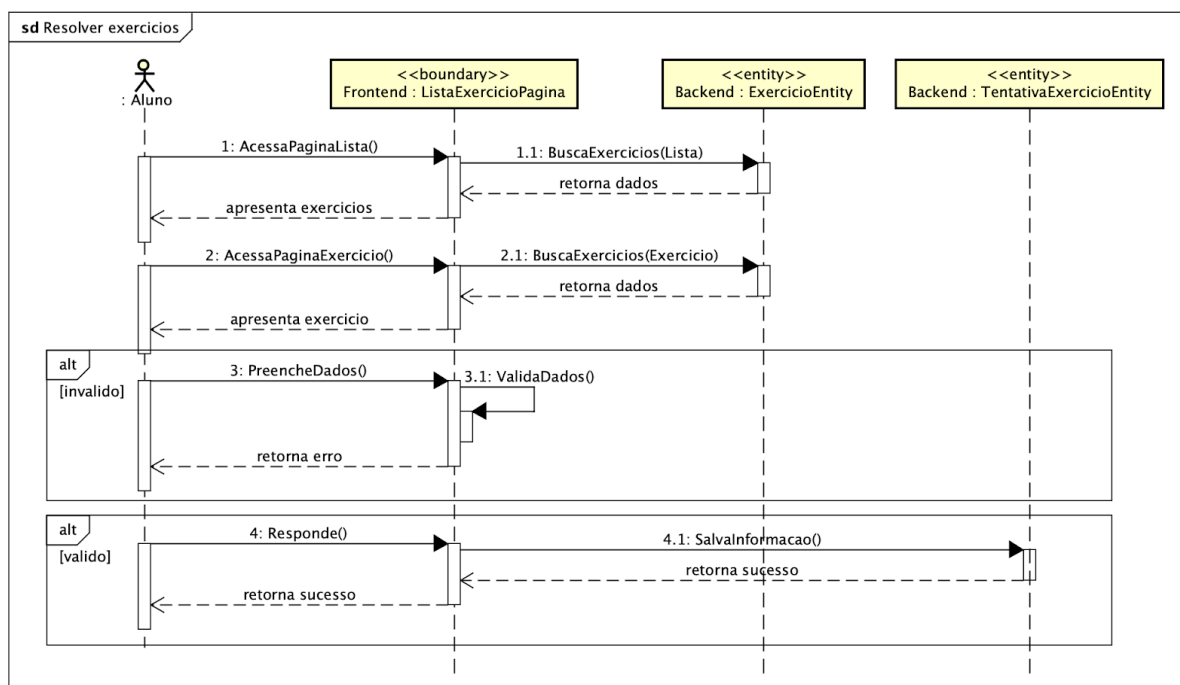
UC06 - Editar comentário



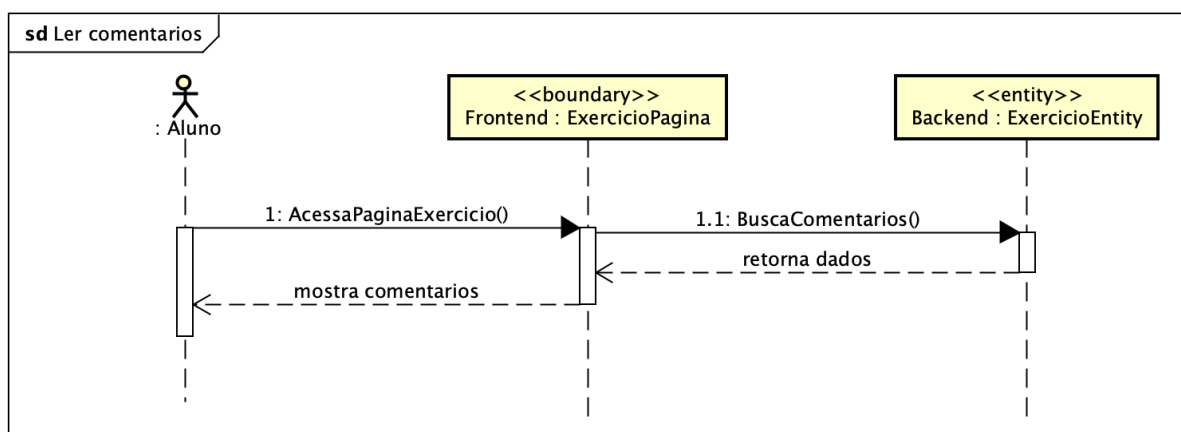
UC07 - Visualizar histórico de tentativas



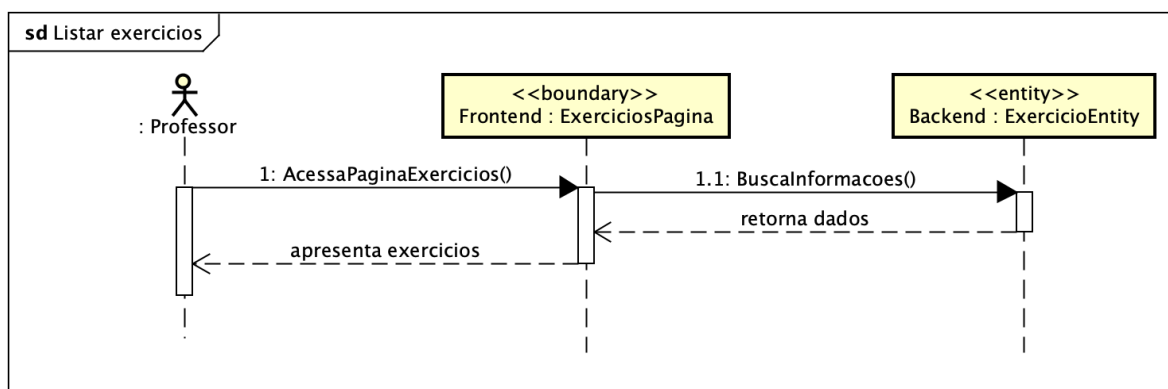
UC08 - Resolver exercícios



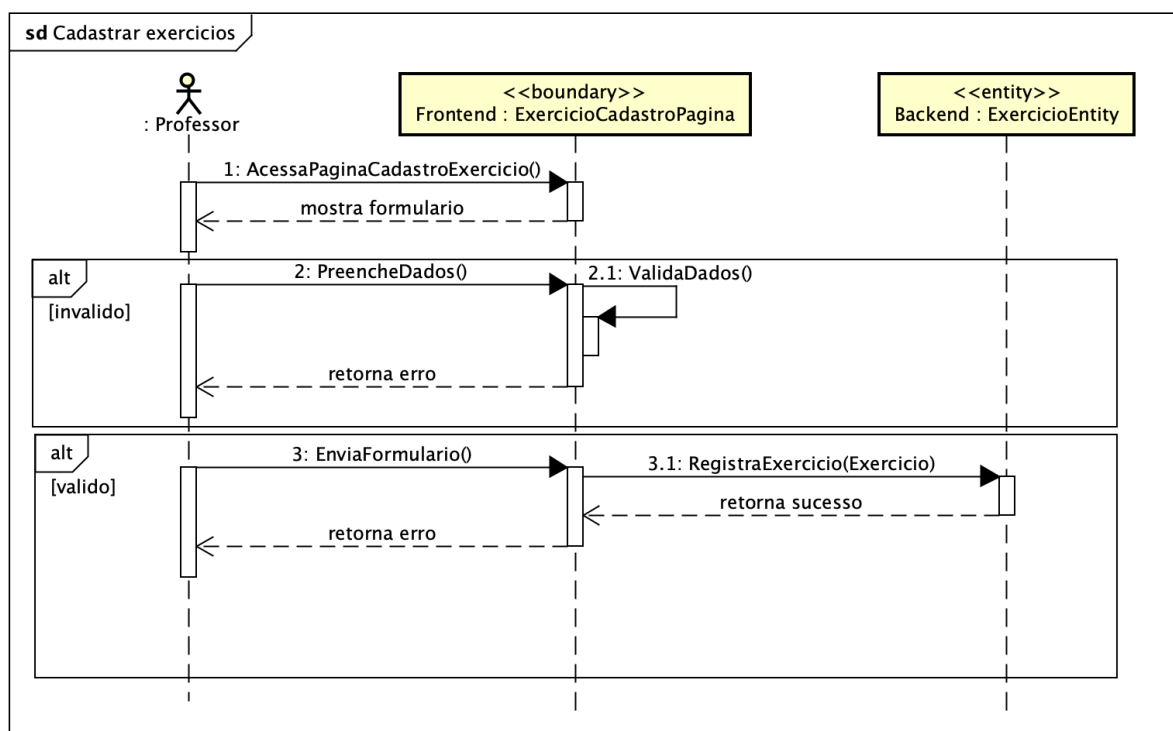
UC09 - Ler comentários



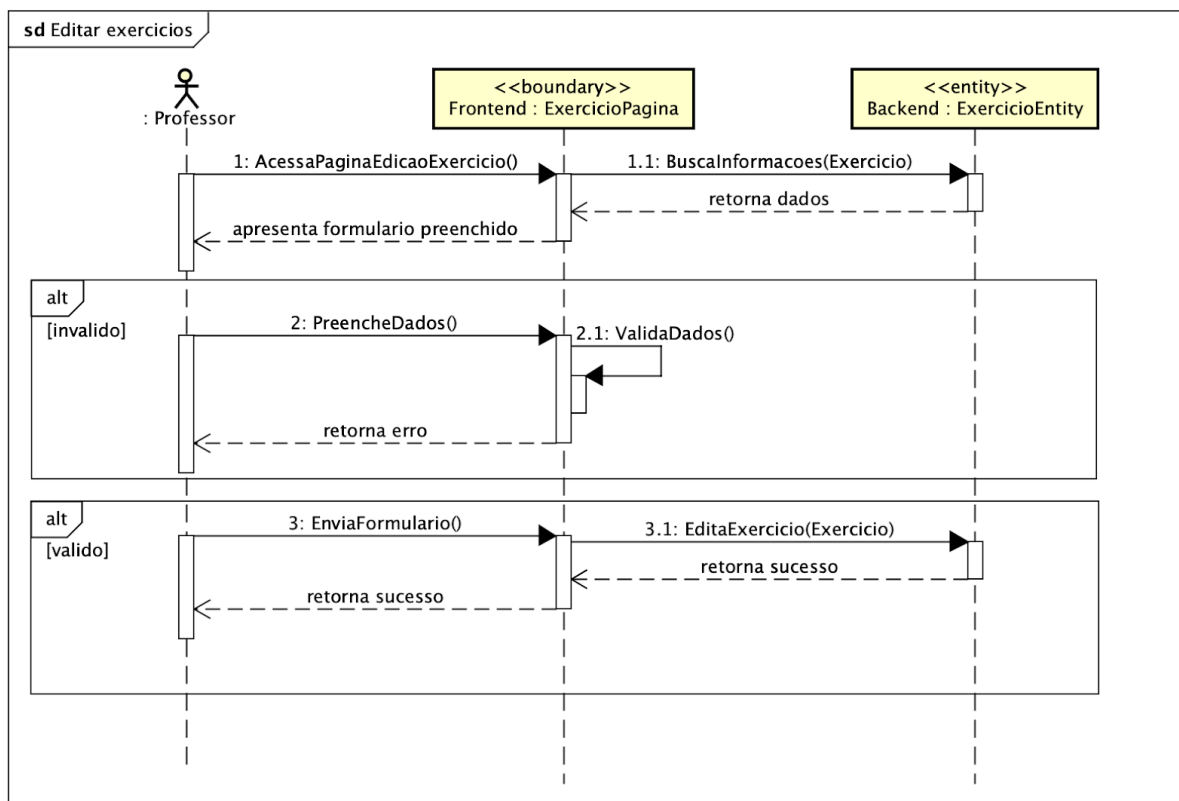
UC10 - Listar exercícios



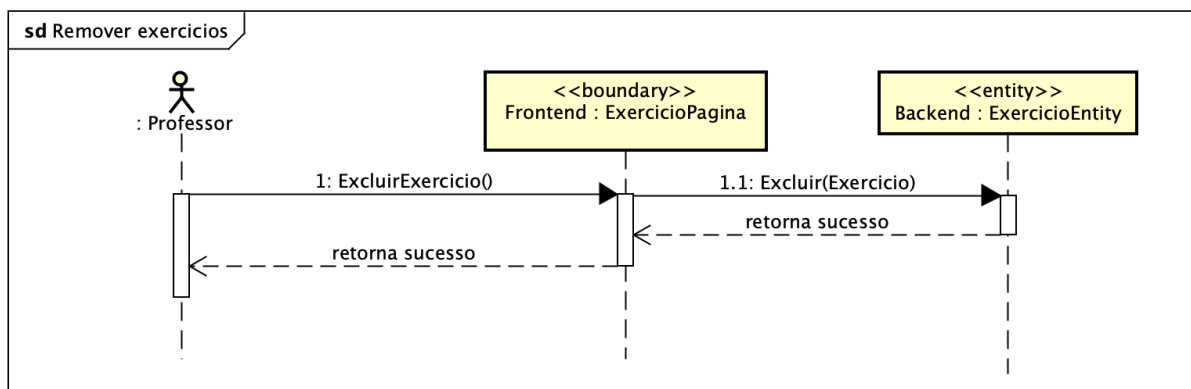
UC11 - Cadastrar exercícios



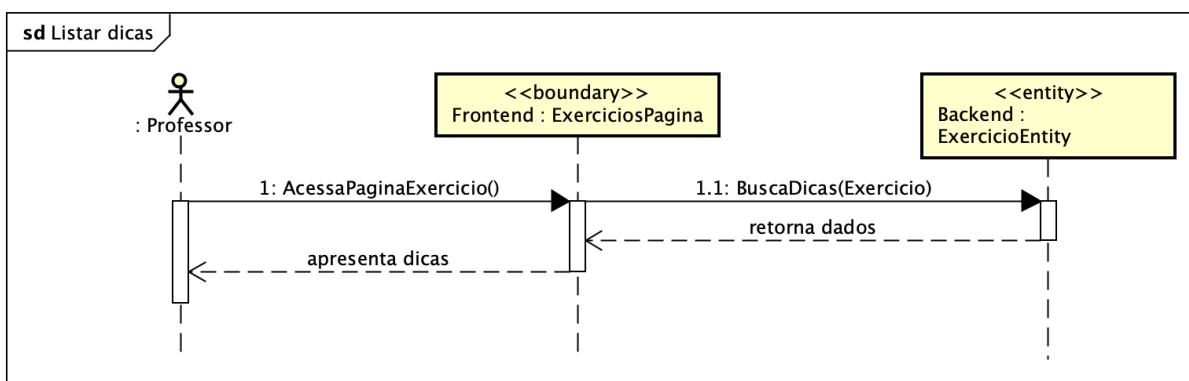
UC12 - Editar exercícios



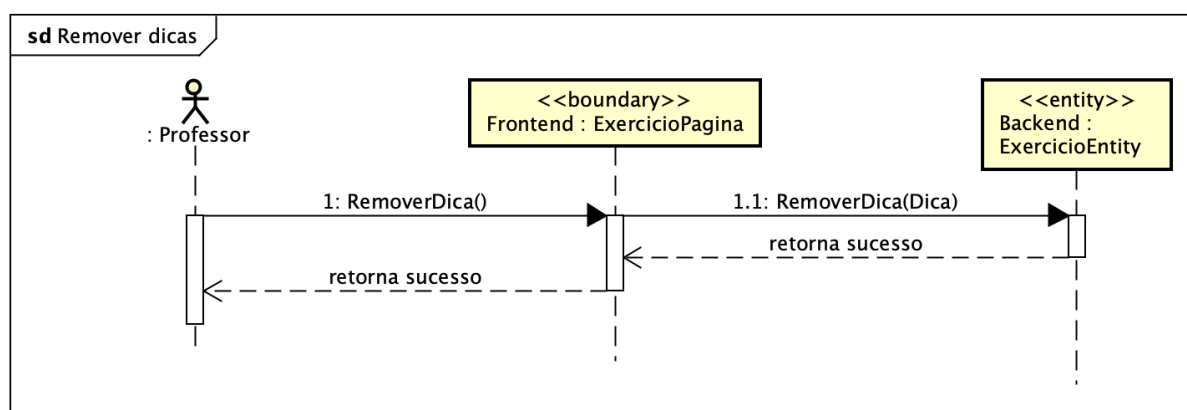
UC13 - Remover exercícios



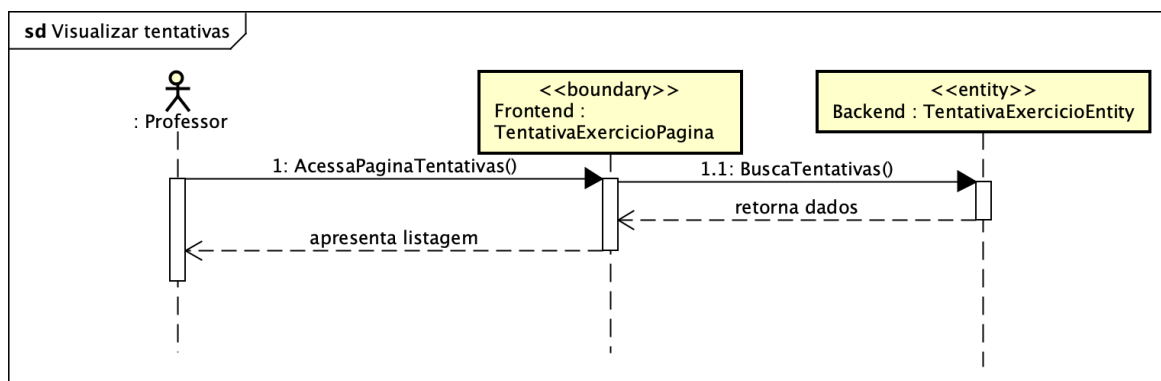
UC14 - Listar dicas



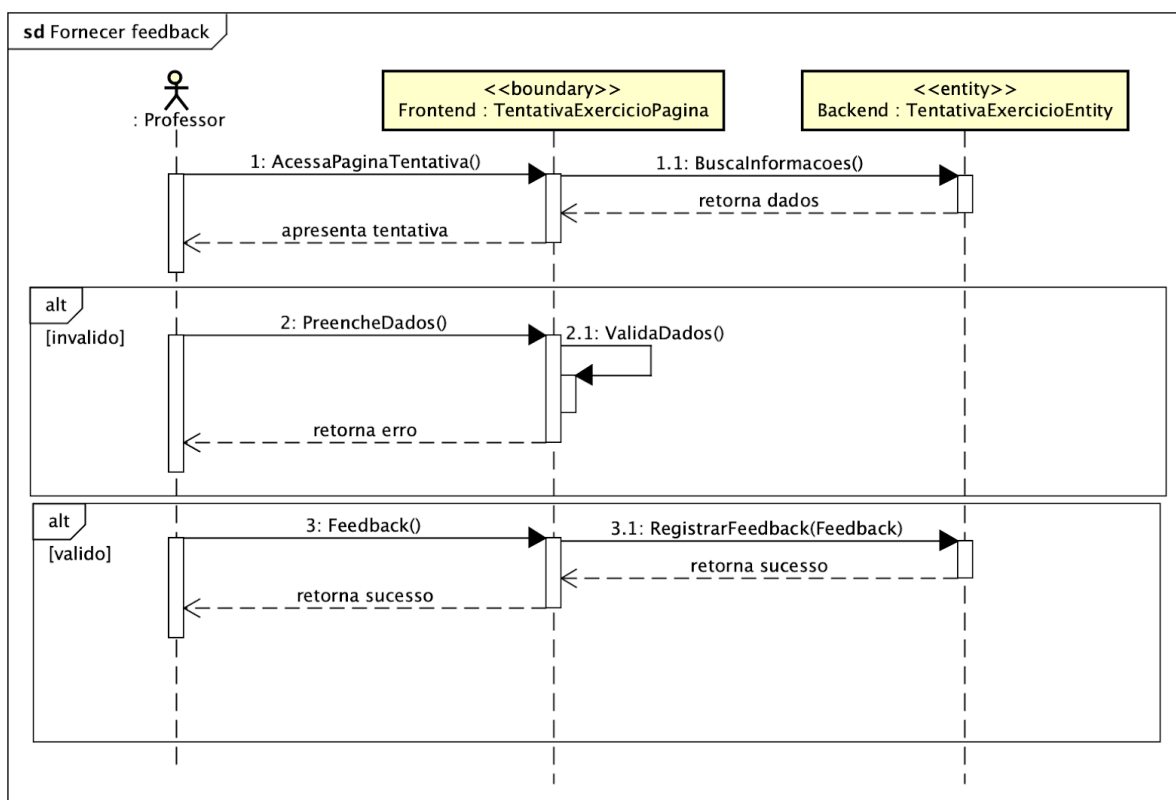
UC15 - Remover dicas



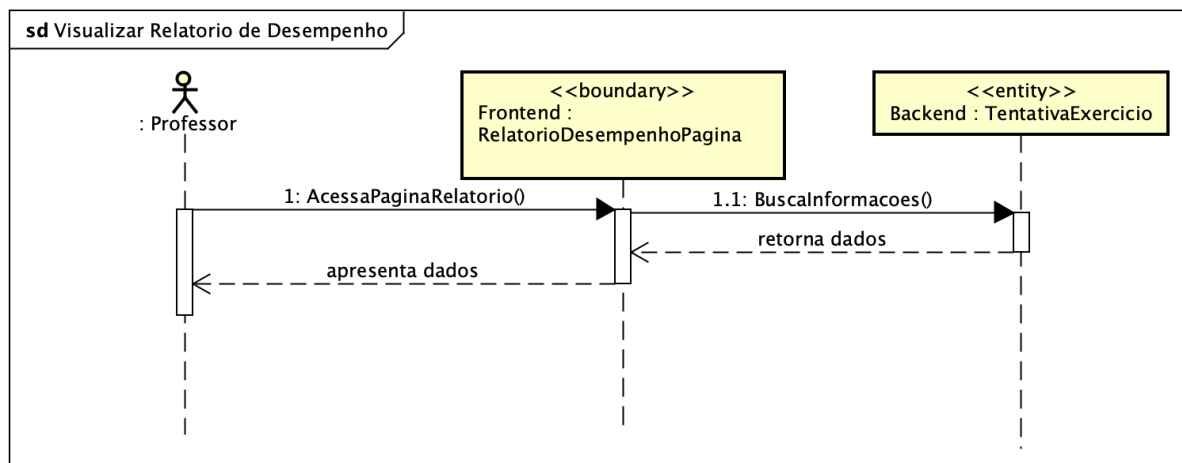
UC16 - Visualizar tentativas de exercícios



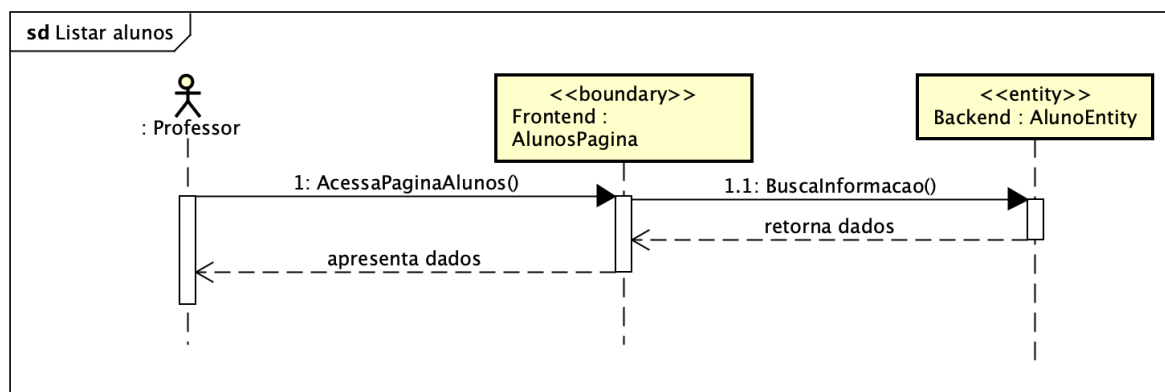
UC17 - Fornecer feedback



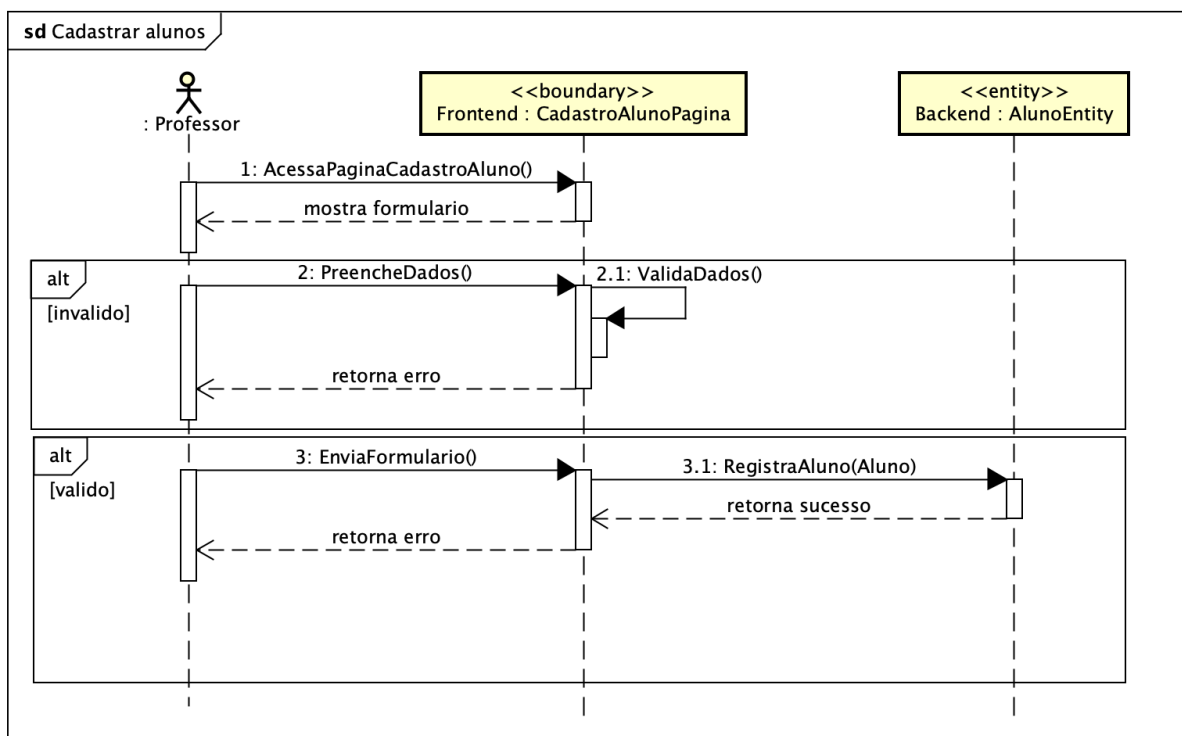
UC18 - Visualizar relatório de desempenho



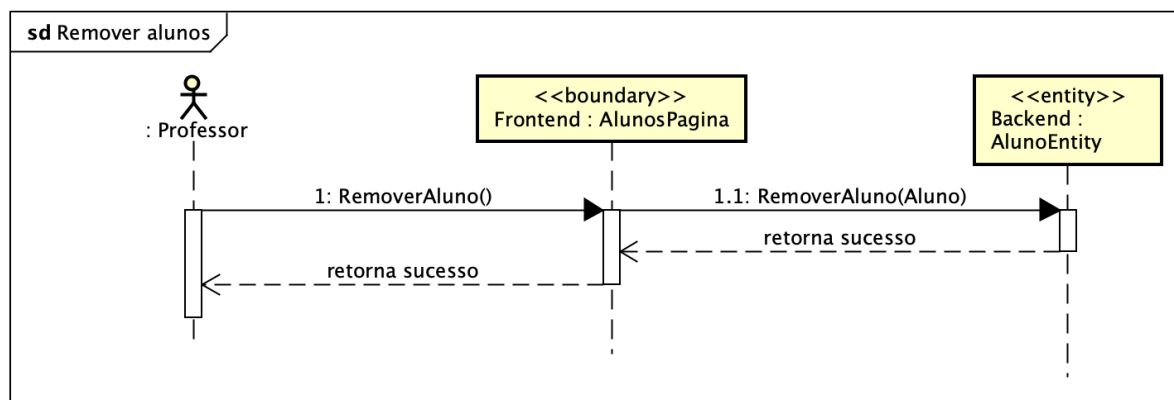
UC19 - Listar alunos



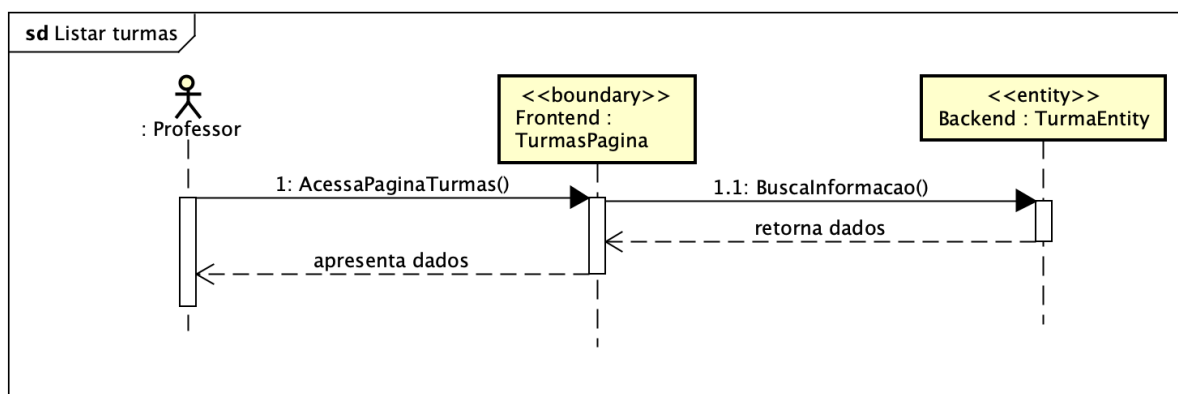
UC20 - Cadastrar alunos



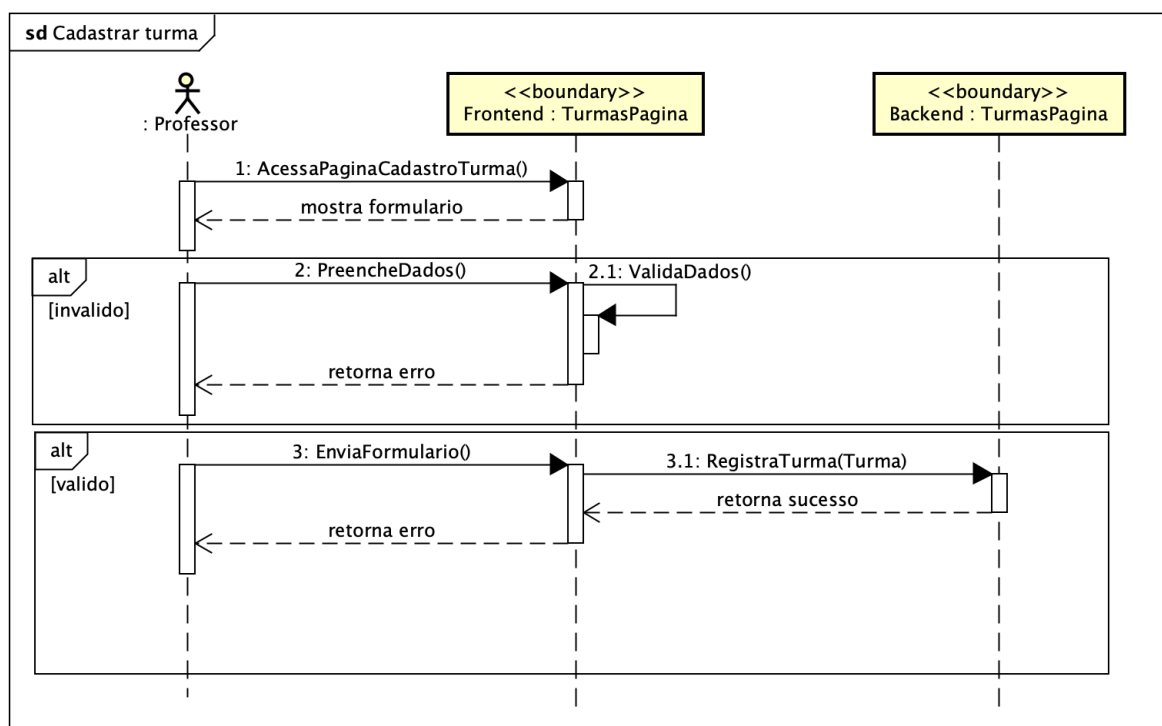
UC21 - Remover alunos



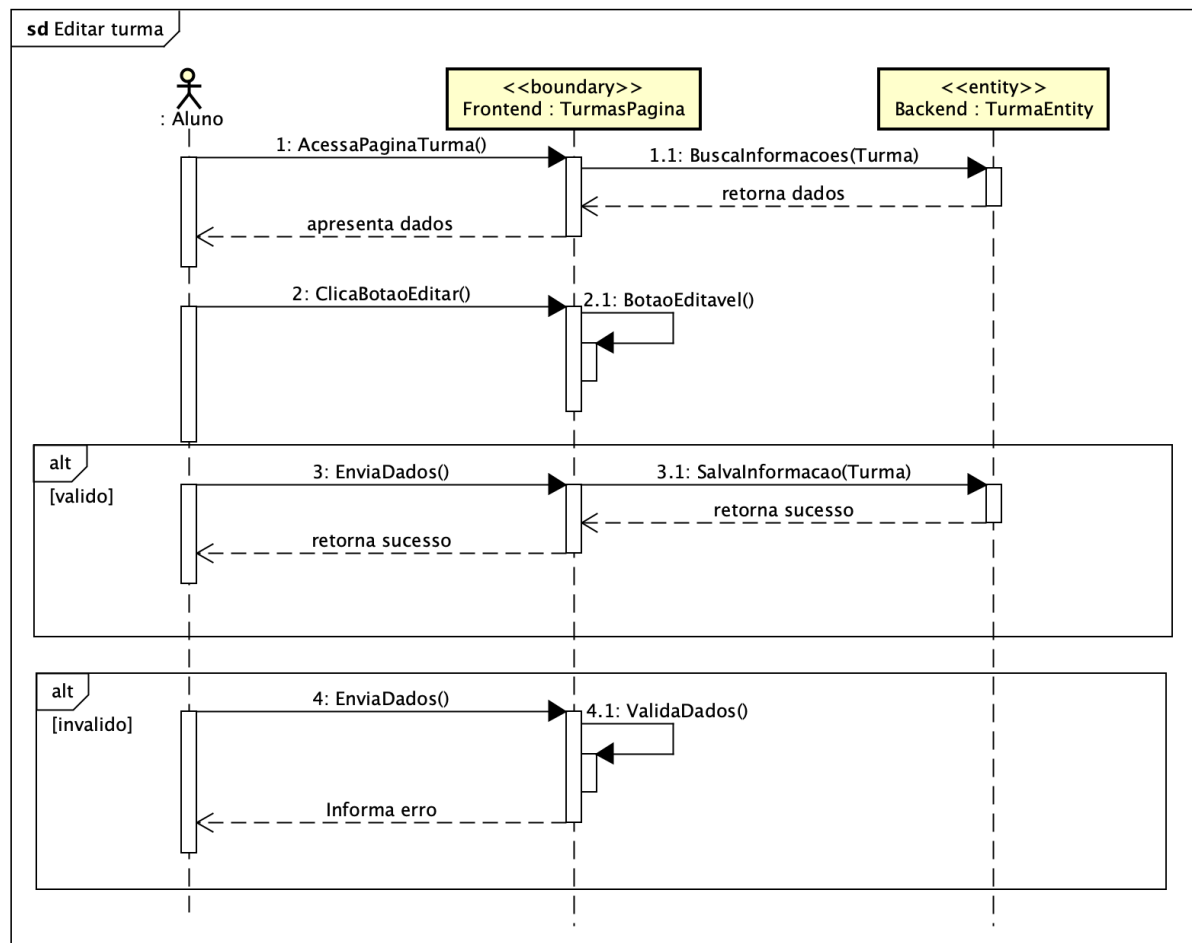
UC22 - Listar turmas



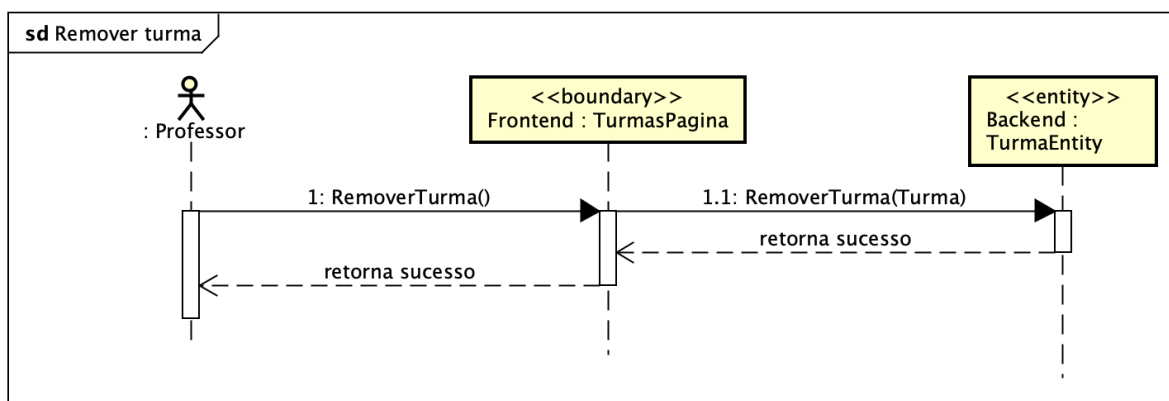
UC23 - Cadastrar turmas



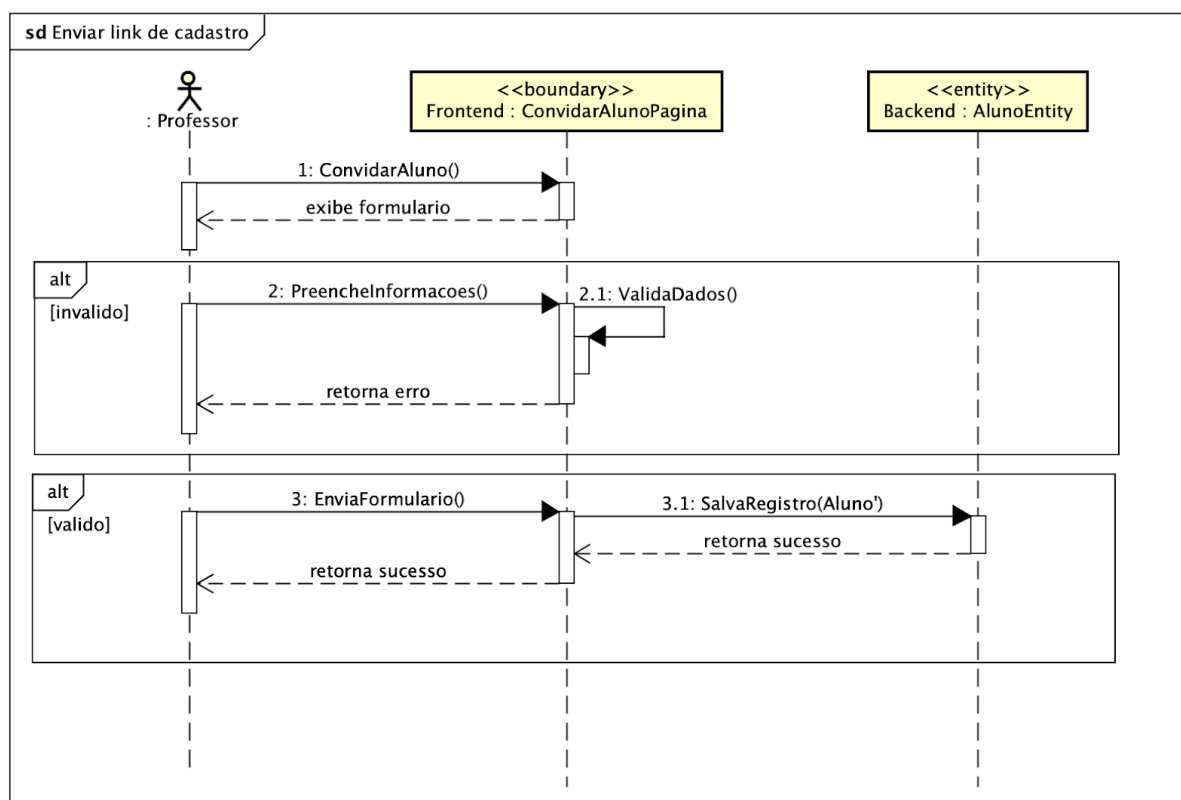
UC24 - Editar turmas



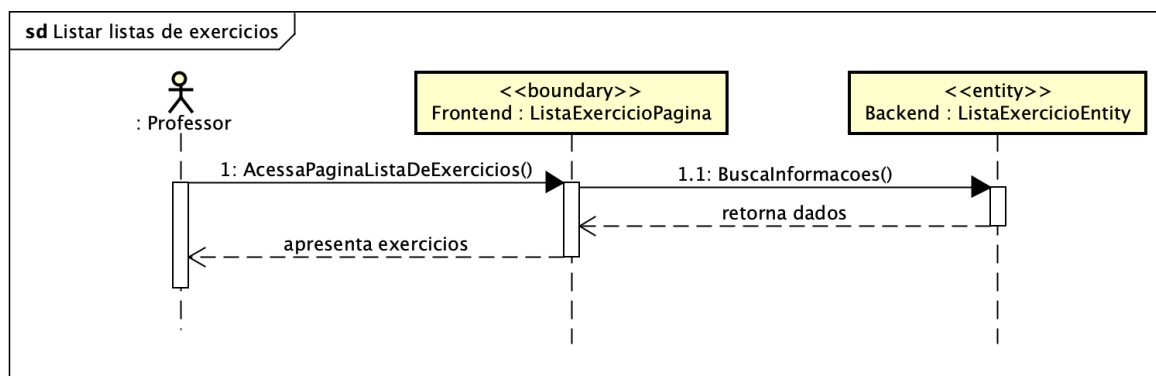
UC25 - Remover turmas



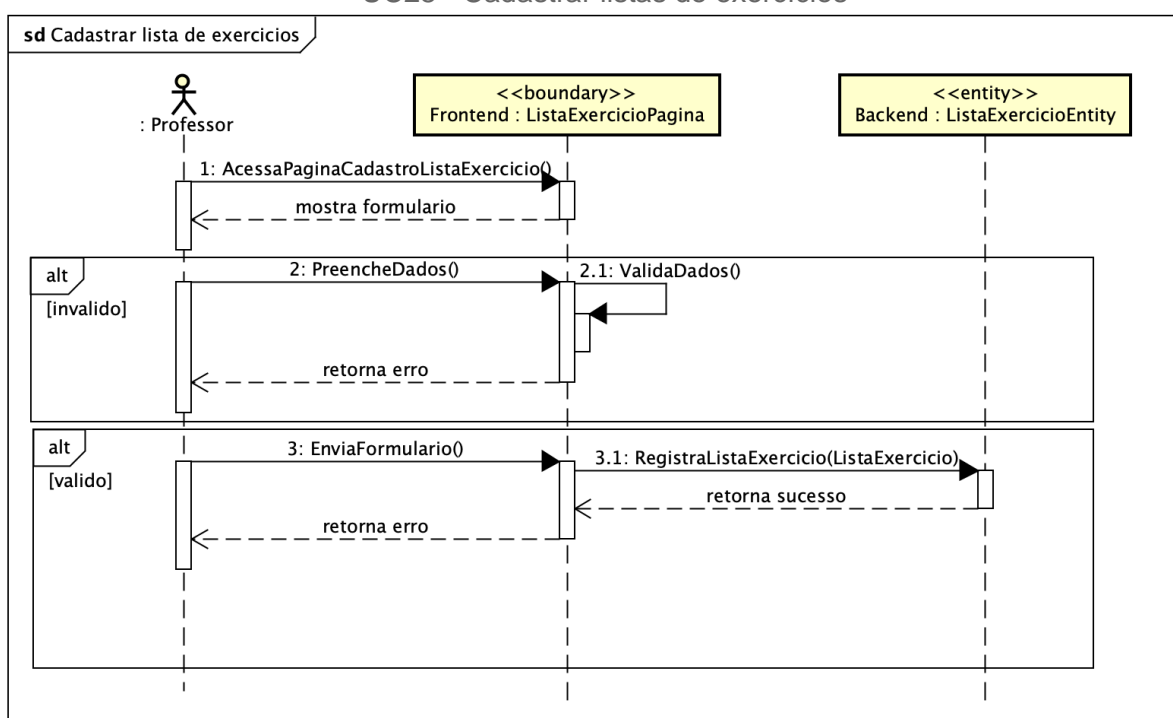
UC26 - Enviar link de convite para ingressar



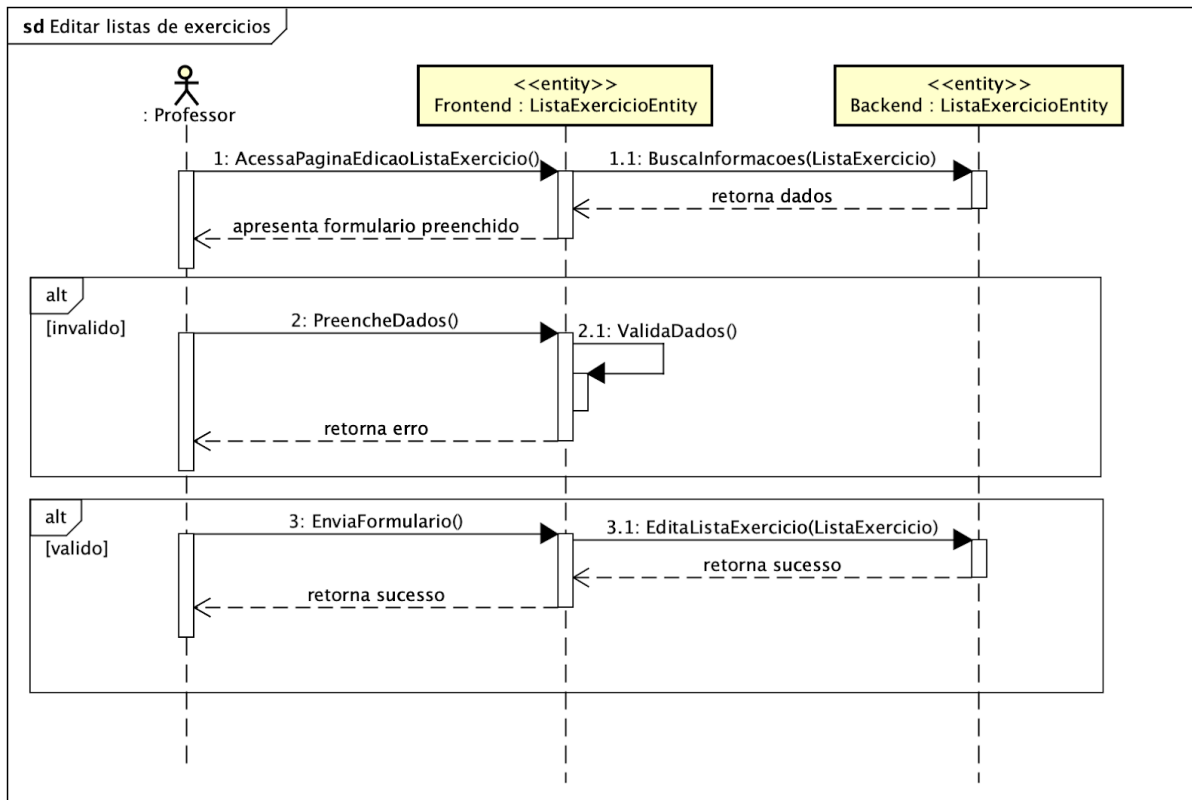
UC27 - Listar listas de exercícios



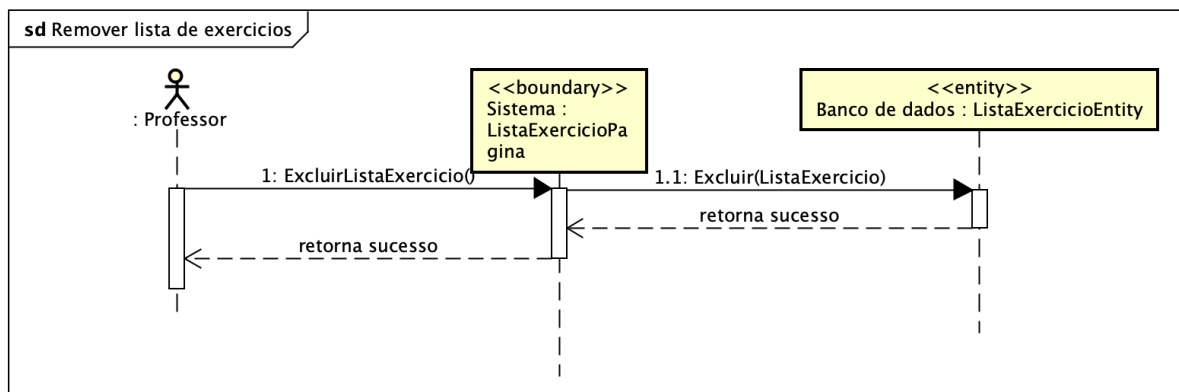
UC28 - Cadastrar listas de exercícios



UC29 - Editar listas de exercícios



UC30 - Remover listas de exercícios



APÊNDICE F – MODELO LÓGICO DE DADOS

