

UNIVERSIDADE FEDERAL DO PARANÁ

FERNANDO LUIS DUNAISKI BRUGINSKI
ROGÉRIO BAYER GHIZI

FarmaALG: IMPLEMENTAÇÃO MOBILE E ATUALIZAÇÃO DA INTERFACE WEB
INSERINDO CONCEITOS DE GAMIFICAÇÃO

CURITIBA

2023



UNIVERSIDADE FEDERAL DO PARANÁ
SETOR DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
Rua Alcides Vieira Arcoverde 1225, -- Bairro Jardim das Américas, Curitiba/PR, CEP 81520-260
Telefone: 3360-5000 - <http://www.ufpr.br/>

Ata de Reunião

TERMO DE APROVAÇÃO

FERNANDO LUIS DUNAISKI BRUGINSKI
ROGÉRIO BAYER GHIZI

FARMAALG: IMPLEMENTAÇÃO MOBILE E ATUALIZAÇÃO DA INTERFACE WEB INSERINDO CONCEITOS DE GAMIFICAÇÃO

Monografia aprovada como requisito parcial à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas, do Setor de Educação Profissional e Tecnológica da Universidade Federal do Paraná.

Prof. Dr. Alexander Robert Kutzke
Orientador – SEPT/UFPR

Prof. Dr. Mario de Paula Soares Filho
SEPT/UFPR

Prof. Dr. Paulo Eduardo Sobreira Moraes
SEPT/UFPR

Curitiba, 30 de junho de 2023.



Documento assinado eletronicamente por **MARIO DE PAULA SOARES FILHO, PROFESSOR ENSINO BAS/TEC/TECNOL**, em 30/06/2023, às 15:52, conforme art. 1º, III, "b", da Lei 11.419/2006.



Documento assinado eletronicamente por **ALEXANDER ROBERT KUTZKE, PROFESSOR DO MAGISTERIO SUPERIOR**, em 30/06/2023, às 15:53, conforme art. 1º, III, "b", da Lei 11.419/2006.



Documento assinado eletronicamente por **PAULO EDUARDO SOBREIRA MORAES, PROFESSOR DO MAGISTERIO SUPERIOR**, em 30/06/2023, às 15:53, conforme art. 1º, III, "b", da Lei 11.419/2006.



A autenticidade do documento pode ser conferida [aqui](#) informando o código verificador **5718020** e o código CRC **A9998CA8**.

UNIVERSIDADE FEDERAL DO PARANÁ

FERNANDO LUIS DUNAISKI BRUGINSKI

ROGÉRIO BAYER GHIZI

FarmaALG: IMPLEMENTAÇÃO MOBILE E ATUALIZAÇÃO DA INTERFACE WEB
INSERINDO CONCEITOS DE GAMIFICAÇÃO

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. Alexander Robert Kutzke

CURITIBA

2023

RESUMO

Durante a pandemia da COVID-19, o uso de ferramentas educacionais online aumentou entre estudantes e docentes. No entanto, a falta de motivação e o uso inadequado dessas ferramentas resultaram em altas taxas de evasão universitária. Para abordar essa questão, a equipe decidiu analisar as dificuldades enfrentadas pelos alunos de computação, com o objetivo de identificar as ferramentas mais utilizadas e propor melhorias para otimizar a experiência de aprendizado. A aplicação escolhida para a monografia foi o FarmaALG, uma plataforma amplamente adotada pelos alunos de computação da UFPR, com potencial de escalabilidade após sua recente refatoração em Java e junto ao framework React. O principal objetivo foi transformar o ambiente do FarmaALG em uma experiência ainda mais personalizável e gamificada, buscando aumentar a motivação dos usuários por meio de técnicas de jogos, como a implementação de mecanismos de pontuação, medalhas e rankings. Para atingir esse objetivo, foram incorporadas funcionalidades de gamificação na plataforma web e criadas duas versões exclusivas para dispositivos móveis, uma para alunos e outra para professores, utilizando o framework React Native com Expo. Essa abordagem buscou democratizar o acesso ao sistema, permitindo interação por meio de dispositivos móveis, como smartphones. O processo de implementação e desenvolvimento seguiu uma adaptação da metodologia Scrum, fornecendo direcionamento claro para as atividades e promovendo uma abordagem ágil. Dessa forma, espera-se que os desenvolvimentos da equipe dentro do ecossistema do FarmaALG resultem em um ambiente de aprendizado mais envolvente, acessível e motivador, contribuindo para a redução dos índices de evasão e melhorando a experiência dos estudantes de computação da universidade.

Palavras-chave: Gamificação. Motivação. Educação a distância. Software Educacional. Desenvolvimento *Mobile*.

ABSTRACT

During the COVID-19 pandemic, the use of online educational tools increased among students and teachers. However, lack of motivation and inappropriate use of these tools resulted in high university dropout rates. To address this issue, the team decided to analyze the difficulties faced by computer science students with the aim of identifying the most used tools and proposing improvements to optimize the learning experience. The chosen Application for the thesis was FarmaALG, a platform widely adopted by UFPR computer science students, with scalability potential after its recent refactoring in Java and React framework. The main objective was to transform the FarmaALG environment into an even more customizable and gamified experience, aiming to increase user motivation through game techniques, such as the implementation of scoring mechanisms, medals, and rankings. To achieve this goal, gamification features were incorporated into the web platform, and two exclusive versions were created for mobile devices, one for students and one for teachers, using the React Native framework with Expo. This Approach aimed to democratize access to the system, allowing interaction through mobile devices such as smartphones. The implementation and development process followed an adaptation of the Scrum methodology, providing clear guidance for activities and promoting an agile Approach. Therefore, it is expected that the team's developments within the FarmaALG ecosystem will result in a more engaging, accessible, and motivating learning environment, contributing to the reduction of dropout rates and improving the experience of university computer science students.

Keywords: Gamification. Motivation. Distance education. Educational Software.
Mobile Development.

LISTA DE FIGURAS

Figura 1 - Home do FarmaALG v1	20
Figura 2 - Tela de comparação de exercícios do FarmaALG v1	21
Figura 3 - FarmaALG v20200603.1.....	21
Figura 4 - Ranking. FarmaALG v20200603.1.....	22
Figura 5 - FarmaALG v3	23
Figura 6 - Khan Academy	25
Figura 7 - Khan Academy Mobile (2022).....	25
Figura 8 - Geekie Games (2022).....	26
Figura 9 - Geekie Games Mobile	27
Figura 10 - Tela inicial do Duolingo.....	28
Figura 11 - Duolingo Mobile	29
Figura 12 - Kanban de Desenvolvimento	31
Figura 13 - Diferenciação das cores de interfaces das aplicações para discentes e docentes	38
Figura 14 - Diagrama de fluxo de interface do protótipo do FarmaALG mobile	39
Figura 15 - Bloco de código do arquivo ClassesContext, parte integrante da lógica de gerenciamento de estado via Context API.	42
Figura 16 - Aplicativo de demonstração da biblioteca NativeBase	43
Figura 17 - Tela inicial do FarmaALG desenvolvida para aplicativos móveis	43
Figura 18 - Classe Java escrita dentro do software de edição de texto Atom	43
Figura 19 - Código C escrito na tela de edição de código na versão mobile do FarmaALG ..	43
Figura 20 - Tela do exercício selecionado, com enunciado, linguagens disponíveis para solução e casos de teste esperados	44
Figura 21 - Tela de execução do exercício, listando resultados dos casos de teste com entrada executada, saída esperada e obtida	44
Figura 22 - Modal de seleção de linguagem para resposta do exercício	45
Figura 23 - Diagrama de arquitetura	47
Figura 24 - Tela de início do App Aluno	49
Figura 25 - Tela de Entrar do App Aluno.....	49
Figura 26 - Tela de Cadastro do App Aluno	49
Figura 27 - Tela principal do App Docente	50
Figura 28 - Tela principal do App Aluno	50
Figura 29 - Tela de Gamificação do App Aluno.....	51
Figura 30 - Tela de Gamificação no App Docente.....	54
Figura 31 - Tela de Minhas Turmas no App Aluno	54
Figura 32 - Tela de Minhas Turmas no App Docente	55
Figura 33 - Modal de Criação de Nova Turma no App Docente	55
Figura 34 - Tela de Turma no App Aluno	56
Figura 35 - Modal de dados do professor no App Aluno	56
Figura 36 - Edição de turma no App Aluno	56
Figura 37 - Tela de Meus Exercícios no App Aluno.....	57
Figura 38 - Tela de Exercício Selecionado no App Aluno	57
Figura 39 - Tela de Criação de Exercício no App Docente.....	58
Figura 40 - Tela de Exercício no App Aluno.....	59
Figura 41 - Modal de Troca de Linguagem no App Aluno	59
Figura 42 - Tela de Edição de Exercício no App Docente	60

Figura 43 - Tela de visualização de alunos respondentes no App Docente	60
Figura 44 - Tela de Edição de Código no App Docente	61
Figura 45 - Modal de Troca de Linguagem no App Docente	61
Figura 46 - Tela de Edição de Código no App Aluno.....	62
Figura 47 - Tela de Execução de Código (Aba Casos de Teste) no App Aluno.....	62
Figura 48 - Tela de Execução de Código (Aba Execução) no App Aluno.....	62
Figura 49 - Tela de Execução de Código (Aba Casos de Teste) no App Docente	63
Figura 50 - Tela de Execução de Código (Aba Casos de Teste) no App Docente	63
Figura 51 - Tela de Perfil de Aluno no App Aluno	64
Figura 52 - Diagrama de Casos de Uso Web.....	71
Figura 53 - Diagrama de Classes.....	72
Figura 54 - Diagrama Entidade Relacionamento.....	73
Figura 55 - Diagrama de Sequência do Cadastro	74
Figura 56 - Diagrama de Sequência de Login	75
Figura 57 - Diagrama de Sequência de Atualizar Informações	75
Figura 58 - Diagrama de Sequência de Criar Turma.....	76
Figura 59 - Diagrama de Listar Turma	77
Figura 60 - Diagrama de Sequência de Editar Turma	78
Figura 61 - Diagrama de Sequência de Remover Turma	78
Figura 62 - Diagrama de Sequência de Criar Exercício	79
Figura 63 - Diagrama de Sequência de Listar Exercícios.....	80
Figura 64 - Diagrama de Sequência de Editar Exercício.....	81
Figura 65 - Diagrama de Sequência de Remover Exercício.....	82
Figura 66 - Diagrama de Sequência de Criar Caso de Teste	82
Figura 67 - Diagrama de Sequência de Listar Caso Teste	83
Figura 68 - Diagrama de Sequência de Editar Caso de Teste	84
Figura 69 - Diagrama de Sequência de Remover Caso de Teste	85
Figura 70 - Diagrama de Sequência de Executar Caso de Teste.....	85

LISTA DE TABELAS

Tabela 1 - NÍVEIS E PONTUAÇÃO CORRESPONDENTE	52
Tabela 2 - CONQUISTAS, PONTUAÇÃO E RARIDADE	53

SUMÁRIO

1. INTRODUÇÃO	11
1.1 PROBLEMA	12
1.2 OBJETIVOS	13
1.2.1 GERAIS	13
1.2.2 OBJETIVOS ESPECÍFICOS	14
1.3 JUSTIFICATIVA	14
2. FUNDAMENTAÇÃO TEÓRICA	15
2.1 GAMIFICAÇÃO	16
2.1.1 CONCEITOS DE GAMIFICAÇÃO	16
2.1.2 MECÂNICAS	17
2.1.3 GAMIFICAÇÃO NA EDUCAÇÃO	18
2.2 FarmaALG	19
2.2.1 DEFINIÇÃO E HISTÓRICO	19
2.4 IMPLEMENTAÇÃO ATUAL	23
2.5 SOFTWARES SIMILARES	24
2.5.1 KHAN ACADEMY - <i>Salman Khan</i>	24
2.5.2 GEEKIE GAMES - <i>Geekie Desenvolvimento de Software S/A</i>	26
2.5.3 DUOLINGO - <i>Luis Von Ahn</i>	27
3. MATERIAIS E MÉTODOS PROPOSTOS	29
3.1 METODOLOGIA DE DESENVOLVIMENTO	29
3.1.1 UML	29
3.1.2 SCRUM	30
3.1.3 KANBAN	31
3.2 FERRAMENTAS DE DESENVOLVIMENTO	31
3.2.1 REACT NATIVE	32
3.2.2 REACT	32
3.2.3 EXPO	33
3.2.4 FIGMA	33
3.2.5 POSTGRESQL	33
3.2.6 SPRING BOOT	34
3.2.7 NATIVE BASE	34
3.2.8 DRAWIO	35

3.2.9 HARDWARE UTILIZADO	35
3.3 PROTOTIPAÇÃO MOBILE.....	36
3.3.1 INTERFACES DESENVOLVIDAS E FLUXO	38
3.4.1 DESENVOLVIMENTO WEB	40
3.4.2 DESENVOLVIMENTO BACK-END.....	40
3.4.3 DESENVOLVIMENTO MOBILE.....	42
3.4 MODELAGEM DO PROJETO	45
3.4.1 DIAGRAMA DE CASOS DE USO.....	45
3.4.3 DIAGRAMA DE CLASSES	45
3.4.4 DIAGRAMA ENTIDADE E RELACIONAMENTO (DER)	46
3.4.5 DIAGRAMA DE SEQUÊNCIA.....	46
4. APRESENTAÇÃO DO SISTEMA.....	46
4.1 ARQUITETURA DO SISTEMA.....	47
4.2 FUNCIONALIDADES	49
5. CONSIDERAÇÕES FINAIS.....	65
6. REFERÊNCIAS BIBLIOGRÁFICAS.....	67
APÊNDICE A: DIAGRAMA DE CASOS DE USO WEB	71
APÊNDICE C: DIAGRAMA DE CLASSES	72
APÊNDICE D: DIAGRAMA ENTIDADE RELACIONAMENTO	73
APÊNDICE E: DIAGRAMAS DE SEQUÊNCIA	74
APÊNDICE F: ESPECIFICAÇÃO DE CASOS DE USO	86
APÊNDICE G - LISTA DE PROTÓTIPOS DE TELA DESENVOLVIDOS PARA A VERSÃO MOBILE.....	134
APÊNDICE H - SPRINTS DE DESENVOLVIMENTO DO PROJETO.....	139
APÊNDICE I - ROTAS DA API.....	143

1. INTRODUÇÃO

Nos dias atuais, com a quantidade de distrações que existem nas mais diversas telas, é cada vez mais difícil tornar os estudantes interessados pelo aprendizado. Muitas vezes as ferramentas de estudo não conseguem despertar a vontade e o interesse do educando em plataformas digitais. De acordo com HOED (2016), diversos fatores são os responsáveis por causar a desmotivação nos alunos, do mau uso de ferramentas educacionais ao conteúdo da disciplina que não desperta o interesse de aprender. Segundo ZAGATTI (2016), alunos que se sentem ansiosos, chateados ou deprimidos possuem dificuldades em assimilar as informações corretamente o que dificulta a aprendizagem.

O mercado de aplicativos móveis tornou-se um dos maiores ramos da indústria de software desde o seu início. De acordo com relatórios da União Internacional das Telecomunicações (UIT), existem cerca de 4,9 bilhões de usuários conectados na internet, desse montante, cerca de 3,85 bilhões de usuários acessam por meio de smartphones. Em 2019, segundo a Salesforce, 80% de todo o tráfego da web veio de smartphones. Podemos citar também dados específicos de uso de aparelhos celulares dentro dos sistemas operacionais das empresas Apple e Google, segundo BRICKMAN (2020), durante o segundo semestre de 2020, foram baixados mais de 28 bilhões de aplicativos na Play Store, principal loja do Android, e mais de 9 bilhões na App Store, loja de conteúdo do sistema iOS.

A necessidade de desenvolvimento de aplicativos móveis continua a crescer, mas, ao mesmo tempo, o caso de uso exclusivo desses aplicativos apresenta vários problemas para os desenvolvedores, pois cada sistema operacional de smartphone é sua própria plataforma, independente dos demais em termos de ecossistema de aplicativos, a experiência do usuário pretendida e as tecnologias de desenvolvimento usadas. Como a base de usuários potencial para a maioria dos aplicativos móveis abrange várias plataformas, os desenvolvedores precisam desenvolver aplicativos separados para cada plataforma, seguindo as diretrizes de design das plataformas e usando suas tecnologias de desenvolvimento.

O FarmaALG implementado por KUTZKE (2015) é uma ferramenta para aplicação da Educação a Distância (EAD), no ensino de programação e lógica, onde os alunos podem submeter códigos de algumas linguagens e receber o feedback

imediatamente do resultado do exercício. Ele provê um Ambiente de Desenvolvimento Integrado (IDE) *online*, onde são submetidos os códigos.

A gamificação é uma técnica muito utilizada para aumentar o engajamento dos seus usuários. Segundo Zichermann e Cunningham (2011), ambientes se tornam mais eficazes para o engajamento do indivíduo quando trabalham com as emoções e desejos dos usuários. De acordo com eles, a aprendizagem é motivadora quando a atividade se torna divertida. É nesse contexto em que podemos perceber o quanto a gamificação está ligada diretamente com a motivação e o engajamento dos alunos nas salas de aula, sejam físicas ou virtuais.

Logo, o presente Trabalho de Conclusão de Curso tem como objetivo aprimorar uma plataforma de entrega de exercícios voltados a programação, por meio da otimização da versão mais recente da plataforma, atualizando a versão web preexistente inserindo os conceitos de gamificação e desenvolvendo uma nova versão para dispositivos móveis. Além disso, será abordado com detalhes a importância do uso da gamificação no ambiente estudantil e como aplicá-la de uma forma eficiente, com o objetivo de aumentar o interesse dos estudantes nos temas abordados nas disciplinas e temas inseridos na plataforma FarmaALG.

1.1 PROBLEMA

Estudos sugerem que grande parte dos alunos têm se sentido desmotivados de como estão sendo conduzidas as aulas e atividades atualmente. Um deles, realizado pelo Datafolha (2020), mostrou um cenário muito preocupante, onde mais de 50 por cento dos alunos não se sentiam motivados para realizarem suas atividades acadêmicas. Outro levantamento realizado por MEDEIROS (2014), demonstra uma série de dados levantados junto a turmas de ensino superior de Direito e Administração de um Centro Universitário da cidade de Fortaleza/CE, apresentando um alto índice de desmotivação principalmente durante os semestres iniciais, gerando a chamada evasão universitária.

De acordo com BARDAGI et al (2014), o termo “evasão universitária” pode ser resumido por vários cenários na relação do aluno com a universidade: a evasão do curso, definida como a troca da graduação cursada; a evasão da instituição, quando o aluno troca de centro universitário por desacreditar nos processos daquela na qual estava inserido; e a temida evasão de sistema, quando o aluno desiste do

ensino superior e de qualquer curso oferecido por ele, levando o indivíduo a uma condição de conformismo com o nível estudantil alcançado. Ainda dentro da tese de BARDAGI et al (2014), uma série de pesquisas apontaram índices altos de abandono em instituições públicas e particulares, variando em taxas de 20% a 50%, principalmente nos três primeiros meses do semestre (50% das evasões do período). Os motivos desses índices incluem o mau desempenho no curso, a frustração causada por aspectos da vida escolar e um relacionamento deficitário entre professor e aluno.

Uma análise detalhada dos dados de *analytics* da implementação atual do FarmaALG colocam essa realidade em números diretos dentro do curso de Análise e Desenvolvimento de Sistemas da UFPR. A ferramenta consiste em extrair os dados através do armazenamento em um banco de dados, onde pode-se obter algumas informações com as métricas analisadas. Durante o segundo semestre de 2021, em um universo de 180 estudantes inscritos em turmas de uma disciplina introdutória do curso que fizeram uso da ferramenta, apenas 48% concluíram mais de 70% dos exercícios propostos durante toda a disciplina.

Esses dados deixam claro que os cursos universitários das instituições brasileiras precisam ir além das técnicas comuns de ensino, como forma a engajar o aluno em um universo cada vez mais conectado, junto a um acompanhamento mais direto de suas dificuldades dentro das atividades acadêmicas.

Apesar da necessidade em inovar nas metodologias de ensino existem uma série de condições que precisam ser analisados para que se obtenha os resultados esperados nesse avanço educacional, segundo KARPINSKI (2017) os fatores críticos para o sucesso são a qualidade, gestão, infraestrutura, recursos humanos, recursos financeiros, modelo pedagógico e a avaliação de aprendizagem. Cada um tem suas peculiaridades e características se eles forem aplicados de maneira eficiente podemos garantir a eficiência da educação a distância.

1.2 OBJETIVOS

1.2.1 GERAIS

Desenvolver um aplicativo móvel, e adaptar a aplicação web já existente da plataforma FarmaALG, introduzindo nas duas versões uma série de ferramentas

de gamificação capazes de aumentar a motivação, o interesse e o engajamento dos estudantes nas atividades acadêmicas dentro dos cursos de tecnologia da UFPR.

A proposta de gamificação usa como base um sistema de pontuação onde ao resolver exercícios os alunos recebem pontos que podem ser consultados em seu perfil ou no Ranking que será disponibilizado nas plataformas onde o FarmaALG estiver disponível.

1.2.2 OBJETIVOS ESPECÍFICOS

- a) Analisar as versões existentes do FarmaALG, implementadas por KUTZKE (2015) e MUELLER (2021), consultando a arquitetura, *endpoints* e interface, de forma a entender as especificidades de cada funcionalidade;
- b) Projetar o funcionamento do algoritmo de gamificação, planejando sistemas de pontos, técnicas de engajamento e sua implementação nas interfaces web e *mobile* de forma a não descaracterizar o objetivo principal da plataforma;
- c) Desenvolver interfaces com base na *User Experience* para aplicação na implementação da plataforma para dispositivos móveis;
- d) Implementar as funcionalidades desenhadas com foco na escalabilidade e manutenção no longo prazo;
- e) Documentar os padrões utilizados de forma a manter a documentação atualizada e tornar o FarmaALG de fácil manutenção no futuro;

1.3 JUSTIFICATIVA

Durante os primeiros meses da graduação, muitos alunos enfrentam limitações financeiras que impossibilitam a aquisição de um computador pessoal ou notebook. Diante dessa realidade, propomos a inclusão de uma versão para dispositivos móveis nos celulares dos estudantes como alternativa viável. Essa solução visa garantir a equidade de oportunidades, permitindo que todos os alunos possam realizar os exercícios independentemente de sua situação financeira. Além disso, ao disponibilizar o aplicativo em dispositivos móveis, promovemos maior flexibilidade no

processo de aprendizado, possibilitando aos alunos acessarem o conteúdo de forma conveniente e adaptável, independentemente do local onde se encontram.

Segundo dados citados anteriormente extraídos do próprio FarmaALG, mais da metade dos alunos pesquisados não concluiu a maior parte das atividades propostas durante a disciplina. A partir disso, mostra-se preocupante o cenário diante do uso da ferramenta, pois as informações apresentam que muitos dos estudantes não estão utilizando a ferramenta como o esperado.

Foi nesse contexto que surgiu a ideia da implementação da gamificação no sistema original e na versão *mobile*, buscando assim aumentar e melhorar o engajamento com a ferramenta, passando a ser melhor e mais utilizada por alunos e professores.

2. FUNDAMENTAÇÃO TEÓRICA

Com o aprimoramento e avanço das tecnologias digitais, a educação vem revendo suas metodologias de ensino-aprendizagem e muitas escolas e universidades já adotaram o sistema EAD (Educação a Distância) e o sistema híbrido de ensino, que consiste em mesclar a educação tradicional com o EAD. Segundo o Censo de Educação Superior - INEP (2020), o número de matrículas totais de cursos presenciais soma aproximadamente cinco milhões, contra três milhões dos cursos na categoria a distância. A modalidade de estudo virtual cresceu aproximadamente 230% quando comparada aos valores de 2010, enquanto os números da presencial aumentaram 2% quando comparados ao mesmo ano.

A crescente demanda pelo modelo de ensino a distância (EAD) tem evidenciado a importância de possuir uma plataforma de aprendizado eficiente, uma vez que é por meio dela que os estudantes realizam e submetem suas atividades. No entanto, segundo o Censo de Educação Superior - INEP (2020), os cursos de tecnologia na modalidade a distância continuam possuindo uma das mais baixas taxas de conclusão, com apenas 16% dos ingressantes concluindo o curso. Essa estatística pode ser atribuída a diversas razões, incluindo falhas no sistema de apresentação dos exercícios nas plataformas e a estruturação geral do ambiente de aprendizado, que muitas vezes não consegue despertar a curiosidade e o engajamento dos estudantes. No entanto, dentre todas as dificuldades enfrentadas, o principal desafio reside na

externalização de comportamentos inadequados por parte dos alunos, o que acaba comprometendo significativamente o processo de aprendizagem (ZAGATTI, 2016).

Essa problemática revela a necessidade de abordar de forma estratégica e abrangente os desafios enfrentados no ambiente virtual de aprendizagem, seja ele dentro de cursos à distância ou presenciais. É preciso criar estratégias para estimular a participação ativa dos alunos, despertando seu interesse e mantendo-os engajados ao longo do curso. Isso pode ser alcançado por meio da utilização de recursos interativos, como vídeos, exercícios práticos, discussões em fóruns e projetos colaborativos. Além disso, é essencial estabelecer uma estrutura clara e organizada para a apresentação dos conteúdos e exercícios, garantindo que sejam acessíveis, relevantes e contextualizados para os estudantes.

2.1 GAMIFICAÇÃO

2.1.1 CONCEITOS DE GAMIFICAÇÃO

Conforme apontado por Daniel Strmečk (2015), a gamificação é uma abordagem que se baseia na aplicação de elementos e princípios dos jogos em contextos que não são intrinsecamente jogos. Seu propósito primordial é potencializar a motivação, o engajamento e a experiência dos usuários.

Para Zichermann e Cunningham (2011) o engajamento é definido pela quantidade de tempo em que um indivíduo tem grande quantidade de conexões com outra pessoa ou ambiente. Vianna (2013) diz que o nível de engajamento do indivíduo é influenciado diretamente pelo grau de dedicação perante as tarefas estipuladas. Essa dedicação, no ambiente dos jogos, é transformada em recompensas e soluções que criam um contexto mais imersivo, divertido e lúdico. O sucesso da gamificação depende diretamente do nível de engajamento dos usuários.

Para Li, Grossman e Fitzmaurice (2012), manter a motivação do indivíduo em qualquer ambiente, exige a necessidade de fornecer estímulos de diferentes formatos e com alta qualidade. Para isso utilizamos o apoio das ferramentas e elementos dos jogos, os quais conseguem conquistar a atenção dos usuários por uma quantidade de tempo relativamente grande.

2.1.2 MECÂNICAS

Muntean (2011) nos diz que, toda e qualquer mecânica de um jogo é mostrada em sua funcionalidade, o que apresenta o aspecto fundamental de qualquer contexto gamificado. Zichermann e Cunningham (2011) entendem que a mecânica de um sistema de jogo é composta por diversas ferramentas que possuem a capacidade de produzir respostas visuais e significativas aos jogadores. Nesse contexto as ferramentas que mais se destacam são:

- Pontos: elemento que pode ser utilizado das mais diversas maneiras, possibilita com que o jogador possa realizar seu acompanhamento e sua progressão durante a interação com o sistema. Os pontos podem servir para estimular o jogador ou até como parâmetro para que outros usuários possam acompanhar o desempenho e resultados dos jogadores, um exemplo colocado no contexto da educação é o acompanhamento do professor perante os alunos.
- Níveis: possibilitam a divisão da gamificação em etapas, indicam também o progresso do jogador. Pode ser utilizado no controle e gerenciamento das habilidades de cada jogador. Conforme o nível aumenta pode-se entender que as habilidades e conhecimentos do indivíduo também aumentaram.
- Placar: tem como principal proposta e objetivo realizar comparações. Placares geralmente são apresentados na forma de uma lista ordenada contendo os nomes e a unidade utilizada para sumarizar esse placar, podendo-se utilizar pontos, níveis, tarefas concluídas e os mais diversos itens da gamificação.
- Divisas: são elementos simbólicos com o objetivo de demarcar objetivos e o progresso. Essas podem ser utilizadas para aumentar o engajamento e o incentivo da promoção social. Alguns exemplos de divisas podem ser medalhas, distintivos e conquistas.
- Integração: esse elemento torna possível a um jogador novato se ver inserido no sistema. Representa o desenvolvimento do engajamento do usuário quando possui seu primeiro contato e experiência com o sistema. Segundo os autores, nos primeiros minutos no ambiente a complexidade do sistema deve ser revelada de forma devagar conforme

seu progresso, buscando sempre reforçar o usuário de forma positiva, também é necessário criar um ambiente com baixa possibilidade de falhas, apresentando assim as tarefas mais fáceis a serem executadas. Aqui encontra-se o aspecto mais desafiador no processo de gamificação, pois seu objetivo é cativar os usuários a se manterem dentro do sistema. O acerto no processo de integração pode garantir o engajamento e fidelidade dos usuários a longo prazo.

- Missões: são designadas aos usuários indicando o caminho a ser seguido no universo gamificado daquele sistema. As missões devem ser interessantes de se realizar, e deve-se possuir um grande número delas dado aos diferentes perfis de jogadores.

2.1.3 GAMIFICAÇÃO NA EDUCAÇÃO

A gamificação na educação tem o potencial de revolucionar a forma como os alunos se envolvem com o processo de aprendizagem. Ao incorporar elementos lúdicos e interativos, é possível quebrar o paradigma de que a educação deve ser maçante e tediosa. Quando implementada de maneira adequada no sistema educacional, a gamificação desperta o interesse dos estudantes, estimula sua participação ativa e promove um ambiente de aprendizado envolvente e motivador.

A gamificação aplicada na educação busca colocar em prática a teoria da autonomia, proposta por Deci *et al.* (1991), que foca na importância de promover e instigar o interesse pelo aprendizado com base na motivação dos envolvidos.

A proposta de *gamificar* um sistema de ensino com o objetivo de motivar os alunos e desenvolver sua autonomia no estudo mostra-se uma estratégia promissora. Ao transformar a aprendizagem em uma experiência envolvente e divertida, a gamificação desperta o interesse dos estudantes, incentivando-os a explorar o conteúdo de forma autônoma.

Além disso, ao oferecer feedback imediato e recompensas, a abordagem gamificada promove a autorregulação do aprendizado e estimula os alunos a utilizarem a plataforma além das exigências curriculares. Nesse sentido, a gamificação emerge como uma ferramenta educacional eficaz, que pode transformar a forma como os alunos se envolvem com o processo de ensino-aprendizagem, resultando em um maior desenvolvimento de suas habilidades e competências.

2.2 FarmaALG

O FarmaALG tem como objetivo desenvolver a qualidade no ensino da programação. Criada como uma ferramenta para otimizar o ensino de lógica de algoritmos para as linguagens utilizadas em todos os cursos de tecnologia da UFPR, com um fluxo de ponta a ponta que se inicia na escrita do código do exercício pelo discente e se completa na correção e análise de relatórios de desempenho da turma pelo docente.

Esse fluxo foi inspirado em um estudo e pesquisa que provém de uma tese de doutorado, que desenvolveu a ferramenta “FARMA: Uma Ferramenta de Autoria para Objetos de Aprendizagem de Conceitos Matemáticos”, criada por MARCZAL (2015). A ferramenta era focada no aprendizado de matemática escolar e utilizava preceitos do erro como forma de desenvolver o conhecimento.

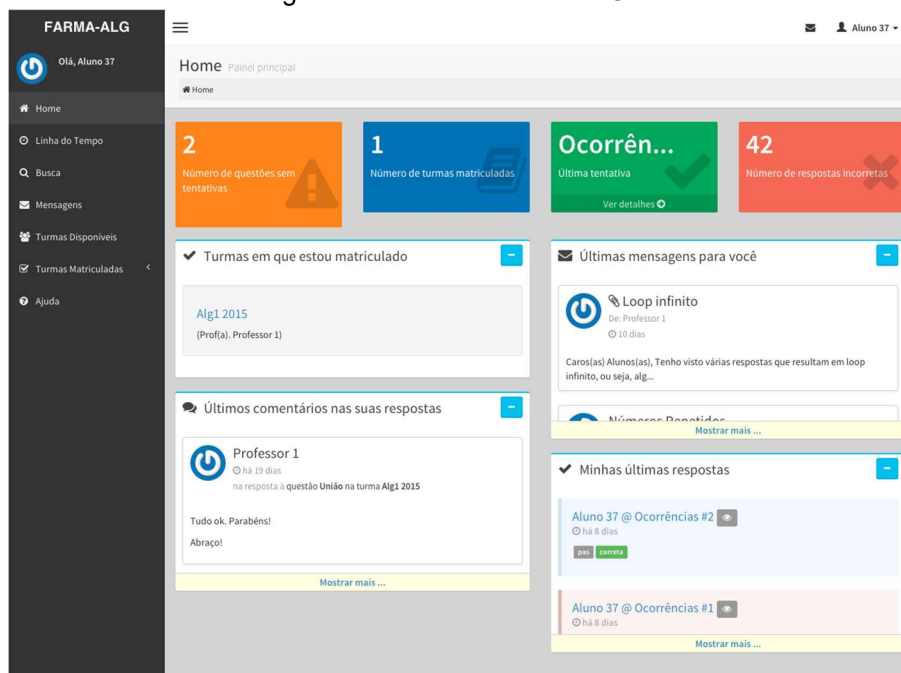
Assim KUTZKE (2015), reformulou essa tese voltando-a ao âmbito da programação, porém mantendo o preceito de aprendizado baseado no erro. A ferramenta executa o código enviado pelo aluno utilizando entradas pré-definidas tentando comparar com resultados também cadastrados previamente. O sistema fornece o feedback instantâneo da resposta do exercício, o que promove a agilidade quanto a correção para o aluno.

O FarmaALG está em sua terceira versão, mas até o momento só está disponível como plataforma *online* via navegador.

2.2.1 DEFINIÇÃO E HISTÓRICO

Criado por KUTZKE (2015), a primeira versão do FarmaALG foi pensada totalmente para a interface web conforme apresentado na Figura 01. As principais funcionalidades desta versão incluem o suporte das linguagens C e Pascal na gestão *online* dos códigos fontes de exercícios entregues pelo aluno. Essa abordagem é possível por meio de uma interface de entrega de atividades semelhante a um editor de texto ou IDE moderna, exibindo exemplos da entrega proposta, dicas que são exibidas ao longo do desenvolvimento do código dentro do editor.

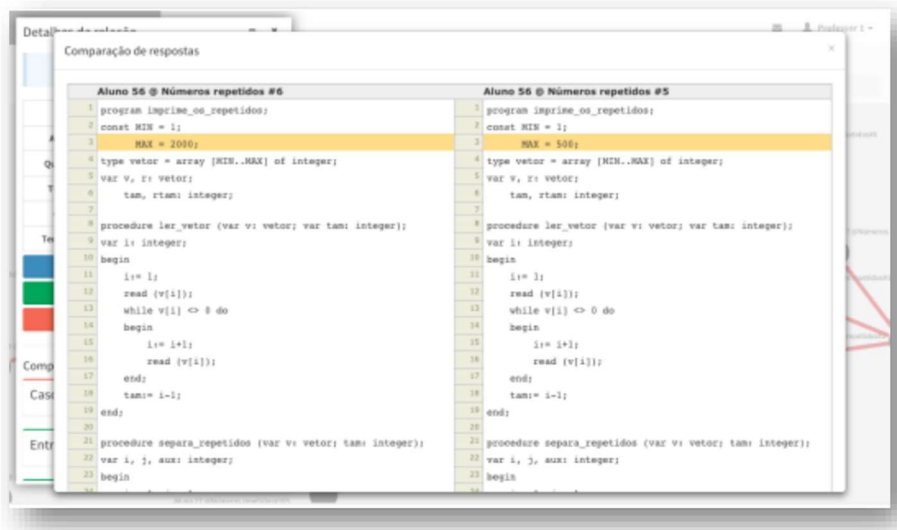
Figura 1 - Home do FarmaALG v1



FONTE: Kutzke (2015)

Segundo análises dessa versão realizadas por KUTZKE (2015) e MUELLER et al. (2021), um dos grandes diferenciais presentes era a capacidade de comparação de respostas de alunos distintos ou de entregas realizadas pelo mesmo discente, com semelhanças e diferenças exibidas lado a lado, de forma a ser possível realizar uma verificação humana de semelhanças na lógica ou plágio, isso pode ser observado na Figura 02.

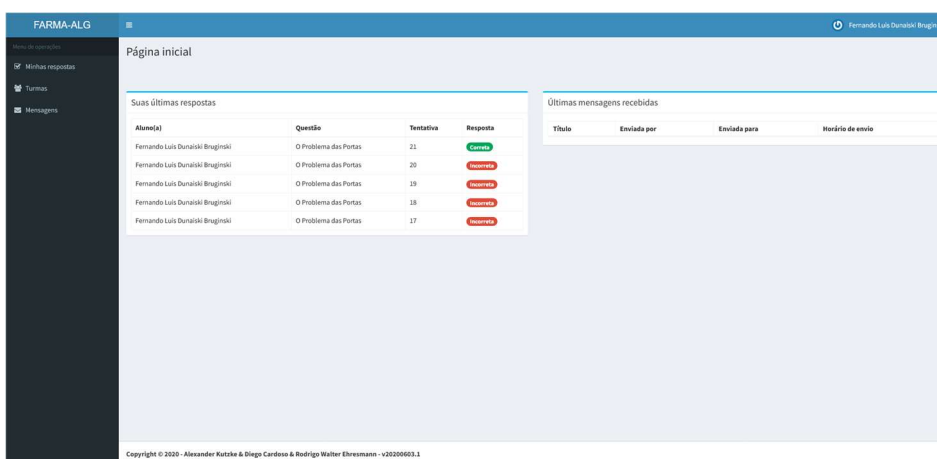
Figura 2 - Tela de comparação de exercícios do FarmaALG v1



FONTE: Kutzke (2015)

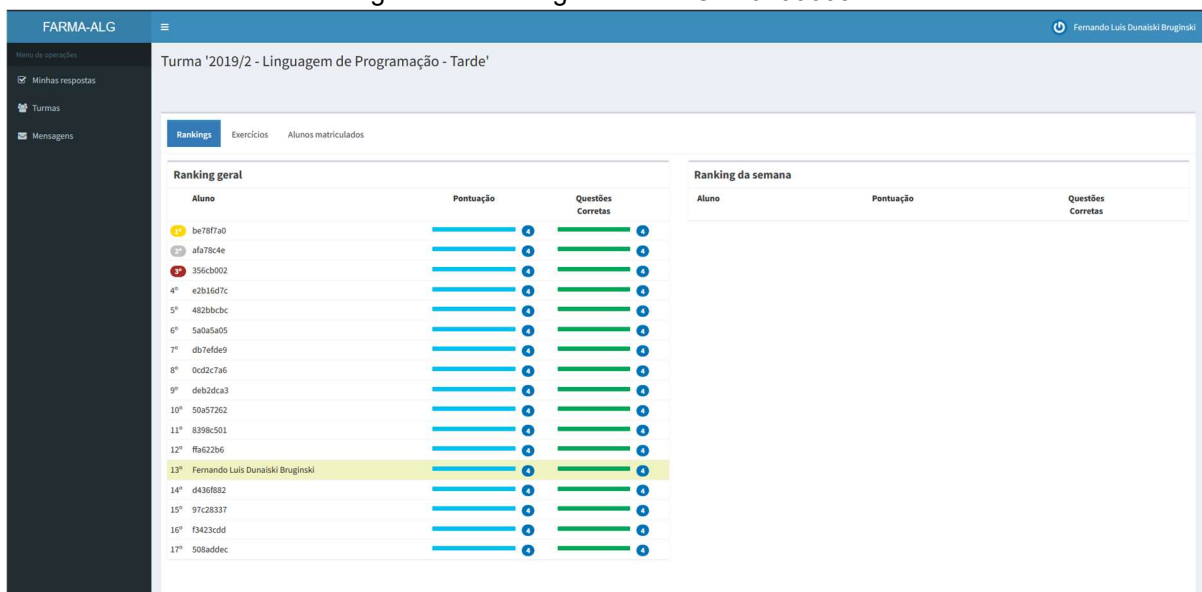
A segunda versão do sistema, desenvolvida em 2020, introduziu um elemento inicial de gamificação, por meio da implementação de um placar simples que registrava a quantidade de questões respondidas corretamente por alunos em uma turma específica. No entanto, essa proposta ainda estava em estágio embrionário, uma vez que não aplicava de forma prática os conceitos e elementos da gamificação. A Figura 03 ilustra a interface da segunda versão na perspectiva do aluno.

Figura 3 - FarmaALG v20200603.1



FONTE: Os autores (2023)

Figura 4 - Ranking. FarmaALG v20200603.1



FONTE: Os autores (2023)

A versão apresentada na Figura 04 está em vigência nos cursos da UFPR, sendo aplicada atualmente nos cursos de Análise e Desenvolvimento de Sistemas, Informática Biomédica e Ciência da Computação. Desenvolvida utilizando o *framework* Ruby on Rails, sob a linguagem altamente tipada Ruby. Princípios da arquitetura Model-View-Controller (MVC) foram utilizados na implementação, de forma a organizar eficientemente um sistema com grande quantidade de componentes, seja no *back-end* ou no *front-end*.

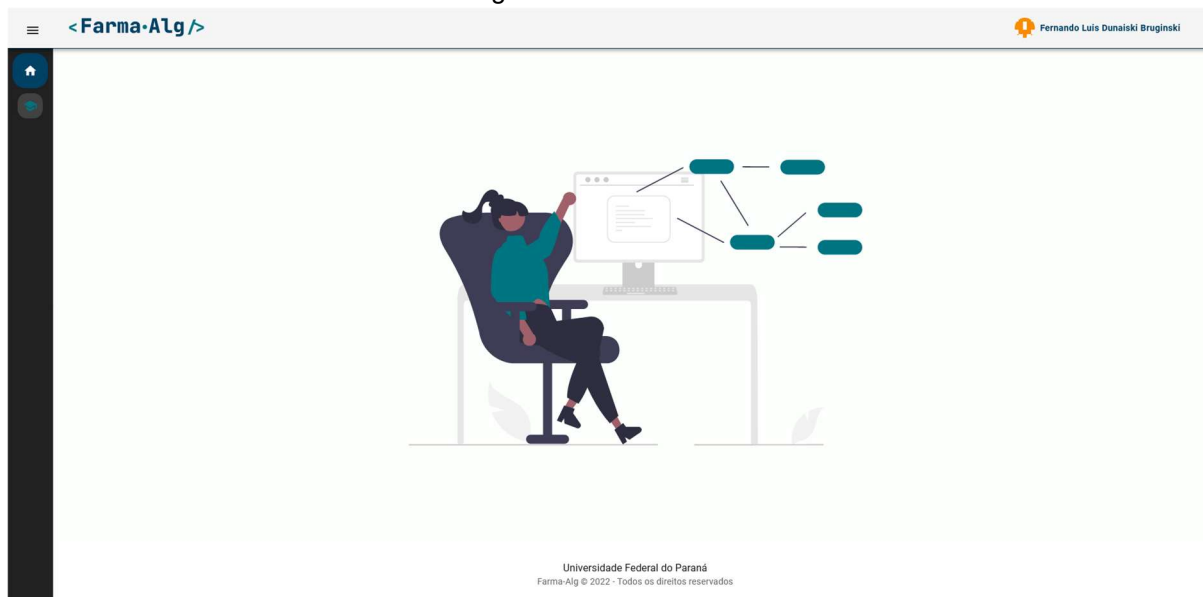
Para a persistência de dados na segunda versão, foi utilizada a tecnologia de NoSQL presente no banco de dados MongoDB de código aberto, sua característica principal é a não existência de esquemas, utilizando-se principalmente de estruturas de dados não relacionais estruturadas ou não. Essa escolha permitiu um relativo aumento na velocidade de escrita e leitura pelo *back-end*.

Como forma de ampliar o leque de opções de busca e análise de dados, KUTZKE (2015) implementou no projeto uma ferramenta conhecida como Elasticsearch, capaz de otimizar as possibilidades de exibição de *datasets* como relatórios de desempenho, análises de comparação de respostas individuais e em grupo, atuando como um elo que fortaleceu a análise do aprendizado dos discentes e se apresentando como grande diferencial dessa versão.

A terceira versão (2021) propôs uma melhoria na interface gráfica do sistema, utilizando novas tecnologias e uma implementação utilizando uma abordagem voltada

a micro serviços. A nova versão buscou garantir longevidade, manutenibilidade e escalabilidade ao sistema mantendo a essência original do sistema. A Figura 05 mostra como ficou o novo *layout* da plataforma em seu último versionamento.

Figura 5 - FarmaALG v3



FONTE: Mueller (2022)

2.4 IMPLEMENTAÇÃO ATUAL

Para a implementação da versão *mobile* do FarmaALG, foi necessário analisar a implementação mais recente da ferramenta para a plataforma Web, realizada por uma equipe de TCC do curso de Análise e Desenvolvimento de Sistemas da UFPR.

Dentro dessa análise foi possível encontrar várias diferenças entre a versão presente em produção e em uso pelos alunos, sendo elaborado um documento detalhando cada item que estava a diferir da versão original da plataforma e o quanto cada ponto afetava no funcionamento da plataforma, posteriormente encaminhado para a equipe de TCC, esse documento permitiu uma melhor definição dos requisitos que seriam adicionados dentro da plataforma *mobile*.

Além disso, foi liberado o acesso ao repositório de código da versão criada pela equipe de TCC, sendo assim, possível avaliar a qualidade das implementações, analisar a correspondência dos diagramas com o banco de dados implementado, verificar a funcionalidade de *endpoints*, validação das interfaces em comparação com a versão anterior e atualmente em produção, além do esboço de novas

funcionalidades que serão necessárias para o funcionamento pleno da plataforma *mobile*.

2.5 SOFTWARES SIMILARES

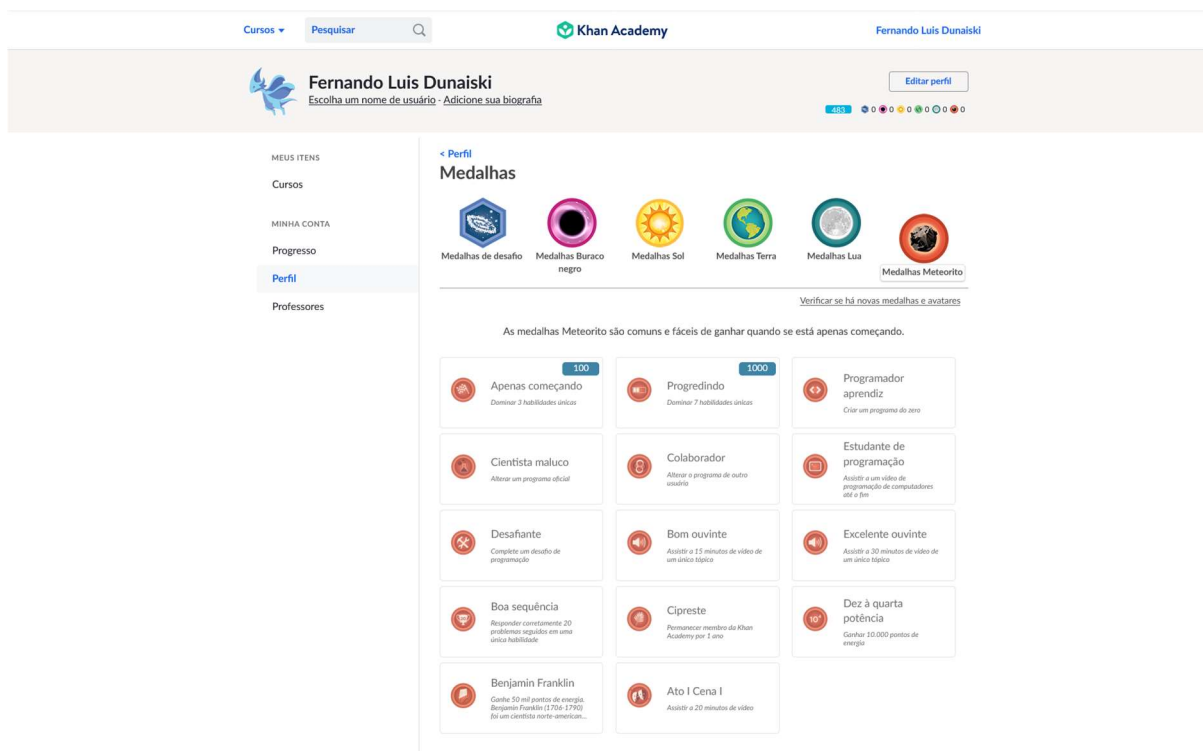
Para desenvolver o presente trabalho, foram analisados alguns *softwares* que apresentam uma proposta semelhante à apresentada no projeto, tais como a educação por meio da gamificação e sua disponibilidade em dispositivos móveis. Dessa forma, podemos rastrear os pontos negativos e positivos de cada aplicação, visando extrair os melhores conceitos de cada uma e comparando com a que está sendo desenvolvida.

2.5.1 KHAN ACADEMY - *Salman Khan*

A plataforma de aprendizado, fundada por Salman Khan, é uma abrangente ferramenta educacional que oferece uma ampla gama de disciplinas, incluindo programação para iniciantes. Conta com um sistema de gamificação que utiliza pontuação. Os pontos são gerados com base no tempo de vídeo que o aluno assiste, nas questões que ele responde durante as aulas e por medalhas que exigem requisitos específicos para serem desbloqueadas.

O sistema de medalhas é construído com base em raridades, as mais comuns e, portanto, mais fáceis de se adquirir são as “Medalhas Meteorito”, já as mais difíceis são apelidadas de “Medalhas Buraco Negro”. As mais raras da plataforma não possuem uma descrição, com o intuito de incentivar os alunos a buscarem como conquistá-las, essas são as que geram a maior quantidade de pontos. Como uma forma de incentivar os estudantes a conquistarem os pontos é possível alterar seu avatar, dependendo da quantidade de pontos que cada um exige.

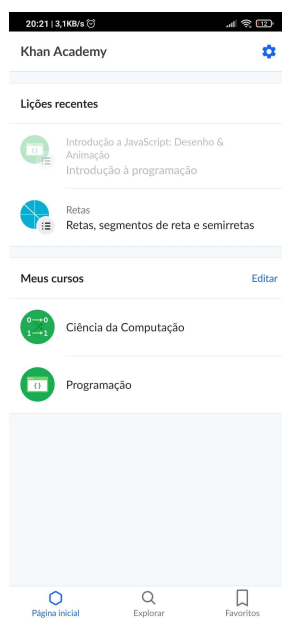
Figura 6 - Khan Academy



FONTE: Os autores (2023)

A aplicação possui uma interface simples e prática, onde podemos seleccionar quais os conteúdos de quais matérias queremos estudar. Também conta com uma aplicação disponível para dispositivos Android e iOS.

Figura 7 - Khan Academy Mobile (2022)



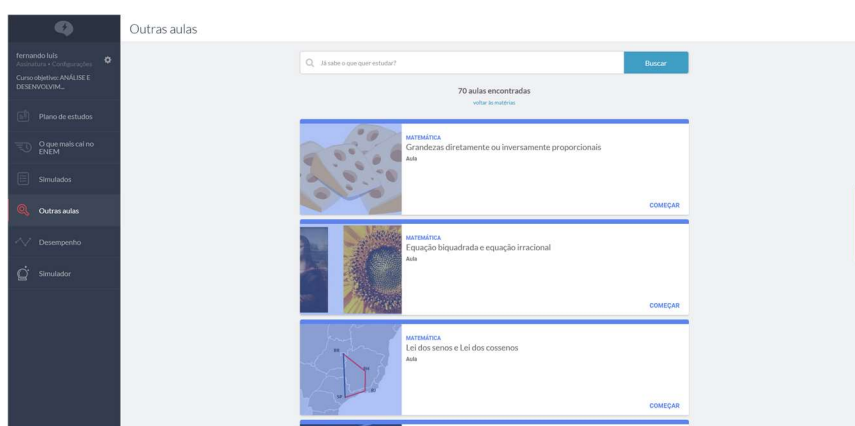
FONTE: Os autores (2023)

2.5.2 GEEKIE GAMES - *Geekie Desenvolvimento de Software S/A*

O Geekie Games é uma plataforma gamificada de estudos com foco para vestibulares e Enem, possui aulas sobre diversos temas de todas as matérias que são exigidas nos vestibulares.

A aplicação apresenta uma plataforma simples e está disponível para dispositivos Android e iOS. Um ponto negativo é a falta de ferramentas sem a assinatura do plano “Geekie Games Pro”, apenas com ela os elementos da gamificação começam a estar presentes no sistema.

Figura 8 - Geekie Games (2022)



FONTE: Os autores (2023)

O sistema de gamificação é construído com a utilização de fases, onde cada fase possui um conjunto de videoaulas e exercícios, conforme o aluno evolui nos conhecimentos ele pode desbloquear novos níveis.

Figura 9 - Geekie Games Mobile

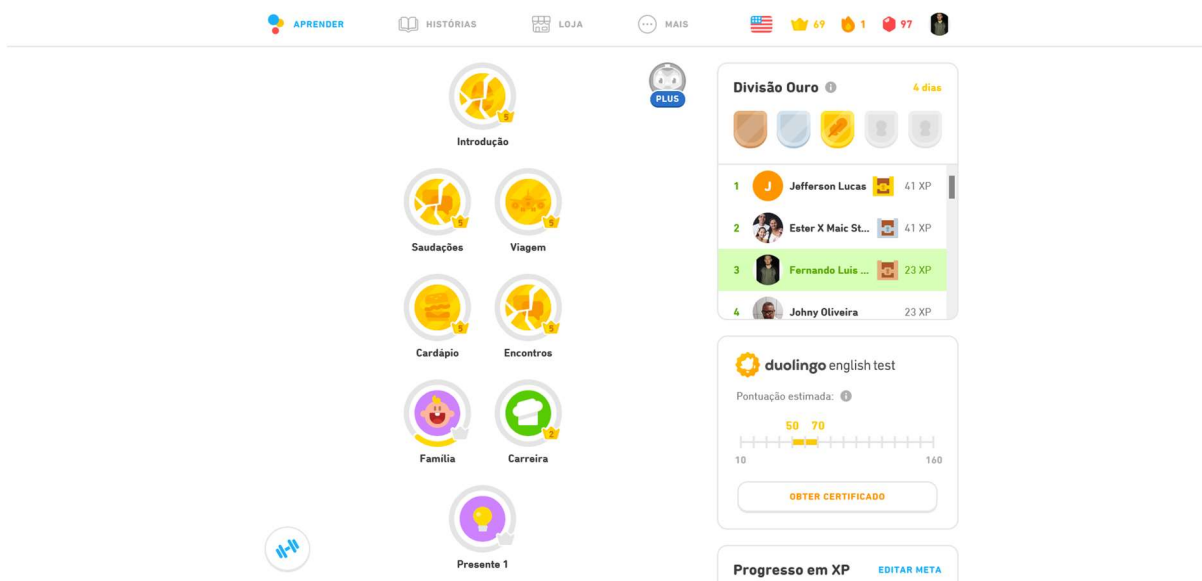


FONTE: Os autores (2023)

2.5.3 DUOLINGO - *Luis Von Ahn*

Essa é uma plataforma para aprendizado de idiomas, criada por Luis Von Ahn, que utiliza muito bem o sistema de gamificação, possuindo diversas mecânicas, como a aquisição de pontos de experiência, sistema de dez divisões indo de bronze até diamante, níveis separando a complexidade dos exercícios, moeda virtual que pode ser usada trocada por alguns benefícios dentro do jogo e sistema de aprendizado em formato de histórias.

Figura 10 - Tela inicial do Duolingo



FONTE: Os autores (2023)

O sistema de divisão apresenta uma mecânica, onde para o jogador se manter em sua divisão precisa adquirir uma quantidade de experiência semanal maior que a dos outros jogadores da mesma divisão, para subir para a próxima divisão precisa ficar entre os melhores da semana. O incentivo dado para subir de divisão é que a premiação semanal aumenta, permitindo com que você compre novos personagens, algumas dicas a mais e vidas para se utilizar nos exercícios.

Para evoluir no aprendizado é utilizado um sistema de árvore de fases que são divididos em temas e começam a se misturar conforme o jogador avança em sua jornada. Conforme passa de nível e adquire experiência, o jogador também desbloqueia conquistas que possuem níveis e raridades distintas. Estes também podem ser utilizados para comparação com amigos que se pode adicionar dentro da plataforma.

A plataforma está disponível para dispositivos web e mobile, exclusivamente para Android e iOS. Foi desenvolvido utilizando os conceitos de “*mobile first*”, que consiste em realizar o desenvolvimento da interface da aplicação começando pelos dispositivos móveis. Toda sua interface é bem colorida e animada, buscando trazer o máximo de imersão e interesse para seus usuários.

Figura 11 - Duolingo Mobile



FONTE: Os autores (2023)

3. MATERIAIS E MÉTODOS PROPOSTOS

A presente seção tem como objetivo apresentar as metodologias e métodos escolhidos para o desenvolvimento do projeto e as respectivas adaptações que se viram necessárias. Além disso, também serão demonstradas quais as ferramentas utilizadas, o planejamento e a execução do projeto.

3.1 METODOLOGIA DE DESENVOLVIMENTO

3.1.1 UML

A modelagem desempenha um papel crucial na obtenção de sucesso em projetos de software. A UML (Unified Modeling Language), de acordo com Booch, Rumbaugh e Jacobson (2005), é considerada a linguagem padrão para a estruturação de projetos de software. A escolha da UML para o desenvolvimento do aplicativo FarmaALG foi motivada pela sua importância no processo de análise e modelagem de software.

O cronograma de desenvolvimento baseou-se no modelo Scrum fazendo as adaptações necessárias para o formato mais viável para os membros da equipe.

3.1.2 SCRUM

A metodologia Scrum é uma abordagem ágil para o gerenciamento e desenvolvimento de projetos, que tem se tornado cada vez mais popular nos últimos anos. Criada por Ken Schwaber e Jeff Sutherland na década de 1990, o Scrum foi concebido como uma forma de lidar com a natureza complexa e volátil de projetos de software, mas desde então tem sido adotado em diversos outros contextos (SCHWABER; SUTHERLAND, 2020).

O Scrum se baseia em um conjunto de princípios e práticas que visam promover a colaboração, a flexibilidade e a entrega contínua de valor. A equipe trabalha em ciclos curtos e iterativos, chamados de Sprints, durante uma Sprint, o time define um conjunto de itens de trabalho a serem realizados, chamados de Backlog da Sprint. Esses itens são selecionados a partir do Product Backlog, que é uma lista de todas as funcionalidades e requisitos do projeto.

No quadro de sprints do desenvolvimento do projeto (APÊNDICE H) estão descritas as sprints executadas na segunda fase do projeto, em que os casos de uso descritos na especificação completa (APÊNDICE D) foram desenvolvidos.

No Apêndice H, é apresentado um quadro que registra as sprints executadas durante a segunda fase do projeto. Esse quadro tem como objetivo documentar as etapas específicas de desenvolvimento que ocorreram ao longo dessa fase. Nessa segunda fase, as sprints foram direcionadas para o desenvolvimento dos casos de uso previamente definidos e detalhados no Apêndice D. Cada sprint descrita no quadro de sprints corresponde a um período aproximado de quinze dias, contando com dois intervalos de sprint durante os recessos da UFPR.

Os eventos de Daily Scrum e Sprint Retrospective – que são descritos no Guia do Scrum (Schwaber; Sutherland. 2020), como parte da Sprint -, foram julgados desnecessários para o desenvolvimento deste projeto, o que foi realizado foram reuniões semanais de planejamento, com o intuito de promover o diálogo e discutir eventuais problemas, dúvidas e dificuldades encontradas pelos membros da equipe.

O Scrum enfatiza a transparência, a inspeção e a adaptação contínua, permitindo que a equipe se adapte às mudanças e maximize o valor entregue ao longo do projeto.

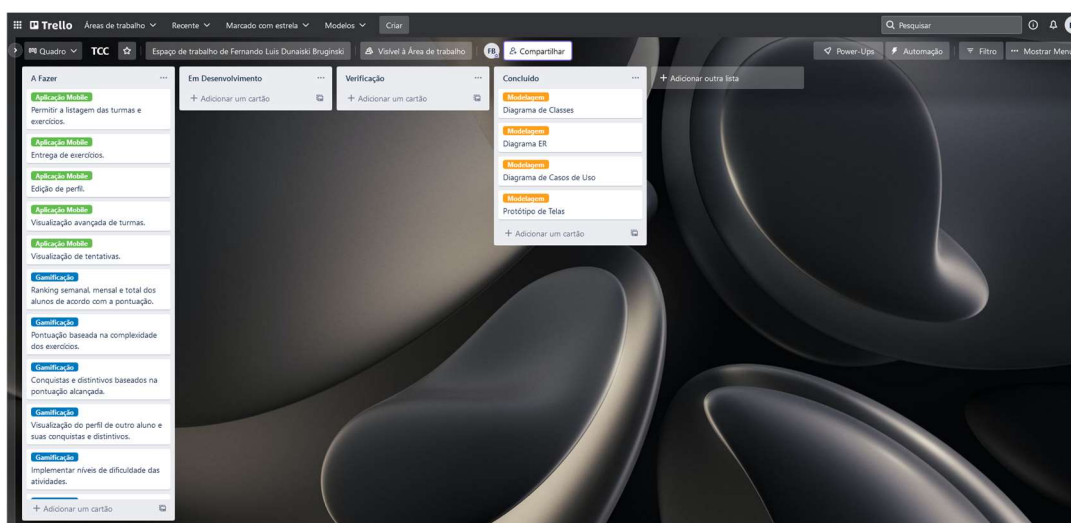
3.1.3 KANBAN

O Kanban é um método que indica para a equipe quanto trabalho pode ser executado num período e a qualidade do trabalho entregue para os clientes internos e externos, mantendo o foco na estratégia da empresa (ANDERSON e CARMICHAEL, 2016). Ahmad, Markkula e Ovio (2013), complementam afirmando que essa visão é possível a qualquer momento do estágio de realização das atividades no Kanban devido a organização da sua estrutura visual.

A utilização do quadro Kanban, permite visualizar e otimizar os processos existentes, buscando transformar a cultura organizacional, ao invés de apenas substituir processos (ANDERSON, 2010).

Neste projeto, a ferramenta utilizada como quadro Kanban foi o Trello – desenvolvido em 2011 pela Frog Creek Software, e a utilização dela se deu pela experiência prévia dos integrantes com o sistema prático e visual que facilita muito a visibilidade do processo de desenvolvimento como um todo.

Figura 12 - Kanban de Desenvolvimento



FONTE: Os autores (2023)

3.2 FERRAMENTAS DE DESENVOLVIMENTO

Para a implementação da ferramenta FarmaALG prevista nesse trabalho, a equipe mapeou a utilização da arquitetura de *front-end* e *back-end* prevista no trabalho de MUELLER (2021), onde tecnologias de ponta como microsserviços e virtualização são utilizadas para otimizar o ecossistema da plataforma.

3.2.1 REACT NATIVE

O React Native é uma biblioteca para desenvolvimento móvel multiplataforma, desenvolvido pelo Facebook e lançado em 2015, usando as bases e conceitos da React. Ao longo das versões, obteve suporte completo para o desenvolvimento Android e iOS por meio da comunidade open source. Da mesma forma que o React, React Native utiliza o Javascript como linguagem padrão.

O funcionamento do React Native é baseado no conceito de uma Máquina Virtual Javascript, responsável por fazer a ponte entre o código escrito no arquivo e os módulos nativos do Android ou iOS. As chamadas de módulos nativos como botões, inputs, drawers e demais componentes são feitas na API nativa junto com o código de bibliotecas de terceiros. Essa estrutura permite uma ampla modularidade entre sistemas operacionais, além de consistência entre os estilos utilizados no sistema.

3.2.2 REACT

O React.js é uma biblioteca JavaScript desenvolvida pelo Facebook, lançada em 2013, que revolucionou a forma como os aplicativos web são construídos. Com o React.js, os desenvolvedores podem criar interfaces de usuário interativas e reativas de forma eficiente e escalável.

Uma das principais características do React.js é o seu modelo de programação baseado em componentes. Com isso, é possível dividir a interface do usuário em pequenos pedaços reutilizáveis, chamados de componentes, que podem ser compostos para criar interfaces complexas. Essa abordagem modular traz benefícios significativos, como uma melhor organização do código, maior reutilização e manutenção simplificada.

Graças à sua popularidade e ao suporte da comunidade, o React.js se tornou uma escolha comum para o desenvolvimento de aplicativos web modernos. Empresas como Facebook, Instagram, Airbnb e Netflix utilizam o React.js em suas aplicações, aproveitando seus benefícios de desempenho, modularidade e produtividade. Com sua abordagem inovadora e foco na reatividade, o React.js continua a evoluir e moldar o cenário do desenvolvimento web.

3.2.3 EXPO

O Expo é um conjunto de ferramentas criado para tornar o React Native mais performático e ampliar as suas possibilidades de desenvolvimento. A maioria de suas funções é ofertada por bibliotecas de terceiros de forma individual, porém, o expo ao longo do tempo uniu essas ferramentas em um único pacote.

Em um projeto com Expo, a criação, compilação, navegação e distribuição de aplicativos é feita sob o ecossistema do Expo CLI, sem usar as ferramentas de linha de comando padrões do React Native. Isso traz uma série de benefícios, pois para executar um aplicativo, basta possuir um ambiente Node.js configurado e conexão com a internet para a instalação das dependências existentes no projeto.

Atualmente o Expo possui ampla adoção no mercado corporativo por conta de sua facilidade de *deploy* e grande compatibilidade de plataformas e bibliotecas. Entretanto, bibliotecas não suportadas podem ser adicionadas a um projeto expo por meio do Bare Workflow, que se adapta o projeto e compatibilidade Expo ao comportamento do ambiente React Native tradicional.

3.2.4 FIGMA

Lançado em 2016, o Figma ganhou popularidade rapidamente devido à sua facilidade de uso, recursos robustos e capacidade de trabalhar em diferentes plataformas. O Figma é uma poderosa ferramenta de design colaborativo baseada na nuvem. Ele permite que equipes trabalhem juntas em tempo real, criando interfaces de usuário, protótipos interativos e designs gráficos. Com recursos avançados de edição e compartilhamento, o Figma facilita a colaboração eficiente em projetos de design, independentemente da localização dos membros da equipe.

3.2.5 POSTGRESQL

O PostgreSQL é um poderoso sistema de gerenciamento de banco de dados relacional, que oferece uma ampla gama de recursos e funcionalidades avançadas. Lançado em 1989, o PostgreSQL tem uma comunidade ativa e uma reputação sólida de confiabilidade, desempenho e conformidade com os padrões de SQL.

Com suporte para transações ACID (Atomicidade, Consistência, Isolamento e Durabilidade), o PostgreSQL garante a integridade e a consistência dos dados armazenados. Além disso, ele oferece suporte a consultas complexas, indexação avançada e recursos de otimização de desempenho, permitindo o processamento eficiente de grandes volumes de dados (POSTGRESQL, 2022).

3.2.6 SPRING BOOT

O Spring Boot é um framework de desenvolvimento de aplicativos Java que simplifica a criação de aplicativos robustos e escaláveis. Lançado em 2014 pela Pivotal Software, o Spring Boot baseia-se no popular framework Spring e tem como objetivo facilitar a configuração e o desenvolvimento de aplicativos Java.

Além disso, é altamente modular e extensível. Ele fornece um ecossistema de módulos e bibliotecas que podem ser facilmente integrados ao projeto, abrangendo desde a camada de acesso a dados até a criação de APIs RESTful. Sua abordagem simplificada e suas funcionalidades avançadas tornam o Spring Boot uma opção valiosa para desenvolvedores em busca de produtividade e eficiência no desenvolvimento de aplicativos (SPRING BOOT DOCS, 2023).

3.2.7 NATIVE BASE

O Native Base é uma biblioteca de componentes de interface de usuário para o desenvolvimento de aplicativos móveis multiplataforma usando o React Native. Lançado em 2016, o Native Base fornece um conjunto abrangente de componentes pré-projetados e estilizados, que podem ser facilmente integrados aos projetos React Native (NATIVE BASE IO, 2023).

Com o Native Base, os desenvolvedores podem criar interfaces de usuário consistentes e atraentes em diferentes plataformas, incluindo Android e iOS. A biblioteca oferece uma ampla variedade de componentes, como botões, formulários, cabeçalhos, navegação e muito mais, que são estilizados de acordo com as diretrizes de design nativas de cada plataforma.

3.2.8 DRAWIO

O draw.io é uma ferramenta online de diagramação e criação de fluxogramas. Lançado em 2012, permite que os usuários criem diagramas profissionais de forma intuitiva e colaborativa, sem a necessidade de instalar qualquer software adicional.

Com o 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, para compartilhar e colaborar com outras pessoas.

Sua natureza online e integração com serviços de armazenamento em nuvem tornam o draw.io uma opção conveniente para a criação e compartilhamento de diagramas em diferentes cenários e dispositivos.

3.2.9 HARDWARE UTILIZADO

O hardware escolhido para o desenvolvimento da aplicação pelos discentes envolveu o uso de duas máquinas pessoais, com especificações recentes e alinhados com as necessidades para a execução dos servidores da aplicação Expo e a emulação de ambiente do sistema operacional Android dentro da IDE Android Studio. O desenvolvimento foi realizado em:

- Computador Desktop de Placa Mãe ASUS, com processador do tipo CPU AMD Ryzen 5 2600, 32GB de Memória RAM, placa de processamento gráfico discreto NVIDIA GTX 1650 Super e armazenamento SSD SATA, executando Sistema Operacional MX Linux 21.
- Notebook Acer, com processador do tipo APU AMD Ryzen 7 5700U, 20GB de memória RAM, placa de processamento gráfico integrado AMD Radeon e armazenamento SSD NVMe, executando Sistema Operacional Ubuntu Linux 22.04.1 LTS.

3.3 PROTOTIPAÇÃO MOBILE

Dentro das análises realizadas das três versões existentes da plataforma, foi possível listar os requisitos esperados para implementação no aplicativo *mobile* da plataforma FarmaALG, de forma que ele se torne uma ferramenta de trabalho útil para os usuários, sejam eles alunos ou professores da universidade.

Através da documentação dos *endpoints* utilizados pelo *front-end* na versão desenvolvida por MUELLER et al. (2021), se fez possível desenvolver uma descrição detalhada de cada funcionalidade existente atualmente, sendo primordial para uma escrita de requisitos clara e concisa para a versão *mobile*. Alguns pontos que foram levantados como pilares da versão *mobile* incluem:

- a) Necessidade da criação de dois aplicativos, um para docentes e um para discentes, com experiências distintas e personalizadas para cada perfil de usuário;
- b) Presença completa das áreas de respostas de exercícios, sendo possível respondê-los ou consultá-los por meio do App, com experiência idêntica a apresentada na plataforma web;
- c) Desenvolvimento de áreas para abrigar as funções relacionadas a gamificação, incluindo *rankings* de desempenho do aluno por linguagem, histórico de pontos e dados sobre o nível atual;

Para consolidação dos conhecimentos obtidos pela equipe na análise dos *endpoints* pelo Swagger, foi realizado o contato com a equipe do projeto FarmaALG de 2021, de forma a sanar dúvidas de como determinados fluxos de dados presentes nas versões anteriores foram remodelados para uma melhor escalabilidade.

As principais mudanças notadas incluem a existência de uma área de perfil dedicada para edição de dados do usuário atual, otimizações na exibição de informações para o usuário, como a existência de descrições para turmas, a possibilidade que um único exercício possa ser respondido em linguagens de programação diferentes e a remoção da auto inscrição na turma realizada pelo aluno, nessa versão a inscrição é feita por meio da lista de matrículas de determinada disciplina.

Após esses processos, conseguimos definir quais funcionalidades e possíveis telas poderiam ser implementadas na versão *mobile*, dentro das operações e telas já existentes no FarmaALG, foram definidas:

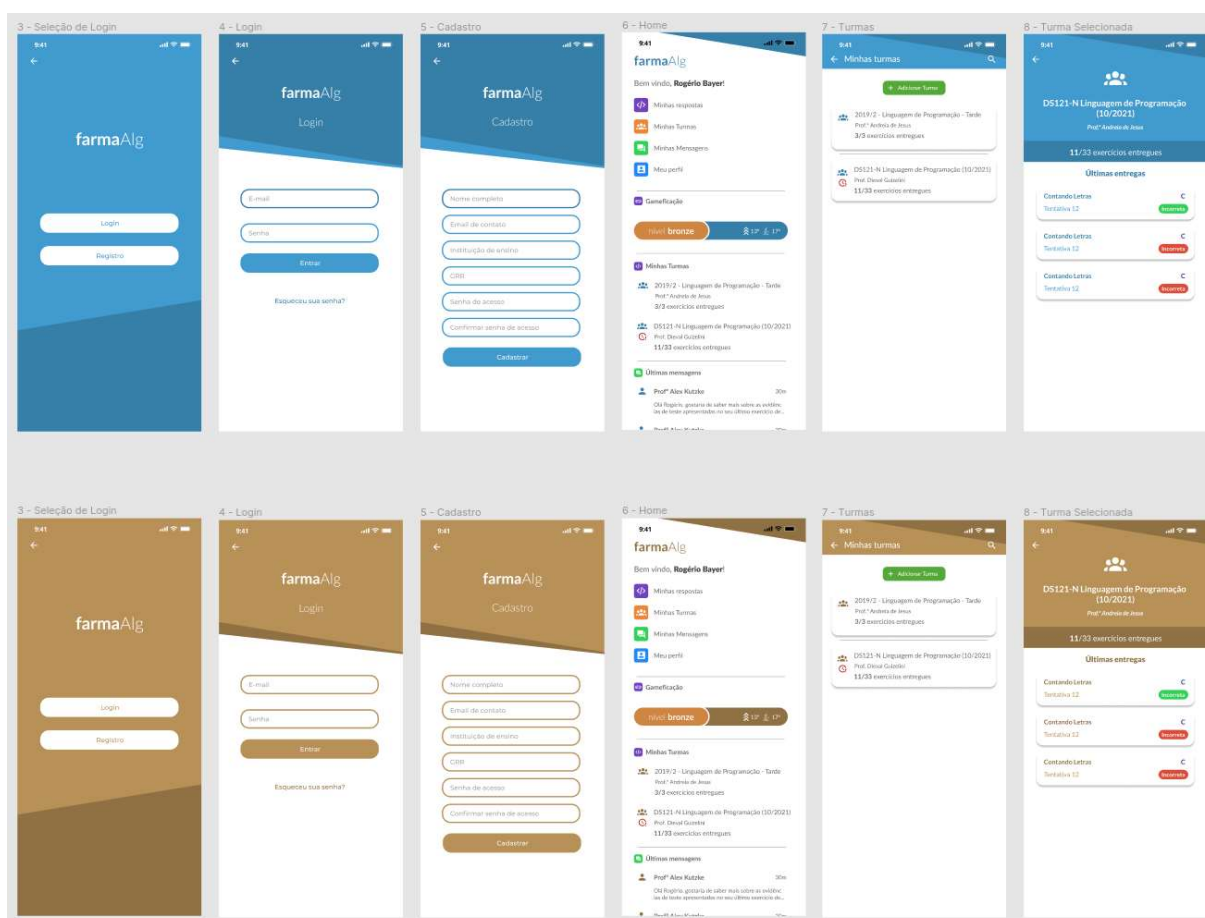
1. Permitir a listagem das turmas e exercícios;
2. Entrega de exercícios;
3. Edição de perfil;
4. Visualização avançada de turmas;
5. Visualização de tentativas.

Dentro dos objetivos específicos da gamificação e penderes de implementação, são primordiais para o escopo do aplicativo:

1. Ranking Geral da classificação dos alunos por pontos;
2. Diferentes níveis de usuário dependendo da quantidade de pontos adquiridos;
3. Conquistas de medalhas ao concluir um limite pré-estabelecido de ações e atividades;
4. Pontuação individual para cada exercício resolvido com sucesso pelo aluno;
5. Visualização do perfil de gamificação exibindo suas medalhas, nível e pontuação.

Foram desenvolvidas 13 telas individuais, divididas em duas versões com cores distintas, a cor azul foi a escolhida para o aplicativo focado em alunos, já o dourado foi designado na versão para professores, diferenciando as experiências dentro de dois aplicativos e criando identificação da comunidade no uso diário. A escolha de duas aplicações deu-se pela praticidade e agilidade de desenvolver aplicativos distintos, pois o trabalho com permissionamento dentro de um único aplicativo demandaria muito mais tempo e pela exclusividade e experiência personalizada que dois aplicativos podem oferecer.

Figura 13 - Diferenciação das cores de interfaces das aplicações para discentes e docentes



FONTE: Os autores (2023)

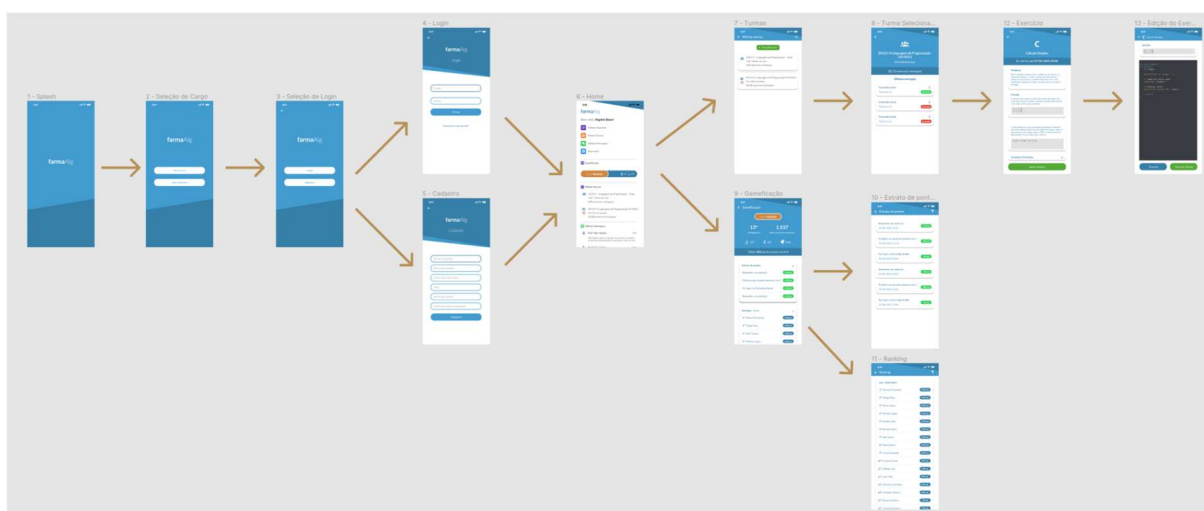
3.3.1 INTERFACES DESENVOLVIDAS E FLUXO

Dentro da tese desenvolvida por Mackinlay (1992), qualquer software que trabalhe com interfaces de usuário pode se tornar excessivamente grande, complexo e difícil de implementar, depurar e modificar. Esse estudo apontou que cerca de 48% do código de aplicativos é dedicado à interface do usuário, tomando precisamente metade do tempo (50%) dedicado para o desenvolvimento total da aplicação. Isso traz a necessidade de uma grande precisão na etapa de prototipação, de forma a diminuir o tempo gasto com a implementação no momento da codificação. Dentro do apêndice D, estão presentes todas as telas desenvolvidas junto a uma descrição com informações sobre as funcionalidades existentes dentro de cada uma delas.

No apêndice D, são apresentadas as especificações dos casos de uso das telas desenvolvidas, juntamente com a descrição detalhada das funcionalidades incorporadas em cada uma delas.

Para uma visualização mais clara da experiência que o usuário teria dentro do App junto ao desenvolvimento do protótipo, optou-se por desenvolver um diagrama de fluxo de interface. A partir desse diagrama foi possível criar sub-diagramas e fluxos focados em cada caso de uso como ferramenta para otimização da usabilidade. Além disso, essa visão geral é efetivamente a combinação de todas as visões baseadas em comportamento dentro dos casos de uso, sendo chamado por muitos estudiosos de “visão arquitetônica da interface” (MACKINLAY, 1994).

Figura 14 - Diagrama de fluxo de interface do protótipo do FarmaALG mobile



FONTE: Os autores (2023)

Com os diagramas de fluxo de interface do usuário realizado (Figura 14), a equipe teve acesso a uma visão de alto nível da interface de um sistema, sendo possível obter rapidamente uma compreensão de como o usuário utilizará a solução, qual é o fluxo principal, onde determinadas funcionalidades podem ser alocadas e como isso pode ter uma melhoria contínua durante o desenvolvimento do código, inclusive na validação de campos de formulário e interfaces com grande quantidade de itens.

3.4 DESENVOLVIMENTO

Nesta seção do documento, iremos descrever de forma sucinta os principais desafios que a equipe enfrentou durante a adaptação do front-end, back-end e criação dos dois aplicativos mobile, bem como os aprendizados adquiridos ao longo desse processo.

3.4.1 DESENVOLVIMENTO WEB

O desenvolvimento web neste projeto acadêmico utilizou a versão de MUELLER(2021) como base, sendo adaptada para incorporar os novos conceitos da gamificação. A implementação da interface web foi realizada empregando a linguagem de programação Javascript, em conjunto com a biblioteca React.

Uma das principais contribuições dessa adaptação foi a criação da página de gamificação, a qual proporciona aos usuários uma visão detalhada do seu progresso no sistema. Dentro das demais áreas do sistema, foi incluída uma barra de progressão fixa ao lado do nome de usuário, a qual representa visualmente o nível atual do aluno e o progresso necessário para alcançar o próximo nível. Essa barra de progressão funciona como uma ferramenta motivacional, fornecendo uma representação clara e tangível do avanço do estudante dentro do sistema gamificado.

Nesse sentido, a combinação da linguagem de programação Javascript com a biblioteca React permitiu uma implementação eficiente e dinâmica da interface web, já comprovada pela implementação de MUELLER (2021), incorporando os elementos do projeto em uma estrutura escalável e responsiva.

3.4.2 DESENVOLVIMENTO BACK-END

Antes de começar o desenvolvimento back-end foi necessário realizar um estudo sobre a implementação pré-existente da aplicação, para entender o funcionamento e quais seriam as mudanças necessárias para atender às novas necessidades do negócio.

A análise da implementação pré-existente teve como objetivo avaliar a arquitetura, a lógica de negócio e as funcionalidades já implementadas. Essa etapa permitiu compreender o fluxo de dados da aplicação, as interações com o front-end e com o container de sandbox.

Após os estudos serem concluídos, decidiu-se que a melhor opção era continuar com as tecnologias que já estavam sendo implementadas, sendo elas, linguagem Java com o framework Spring Boot que facilita e agiliza o desenvolvimento de aplicações de microsserviços, utilização do banco de dados PostgreSQL e a IDE escolhida foi o Visual Studio Code.

Buscando atender as regras de negócio do escopo deste trabalho, foram necessárias diversas alterações nas funções e controladores existentes, afetando seus métodos, comportamento e outras características. Essas modificações foram implementadas visando adaptar a lógica do sistema às novas necessidades e funcionalidades relacionadas à gamificação.

Uma das principais mudanças realizadas foi o ajuste de classes existentes, junto a novas classes e tabelas, as quais desempenham um papel fundamental na implementação dos conceitos de gamificação. Essas classes foram projetadas para registrar as atividades do usuário e atribuir medalhas com base na quantidade de vezes que cada ação é realizada. Cada medalha possui uma pontuação associada e uma raridade específica, pré-determinada pelo grupo, proporcionando um sistema de recompensas completo.

Além disso, para a criação do ranking, são considerados os pontos acumulados por cada usuário. No entanto, foram tomadas precauções para proteger a privacidade dos alunos e a competitividade saudável. Para isso, dentro do ranking da aplicação dos discentes, o ranking exibe somente o primeiro e último nome inteiramente coberto por asteriscos, mantendo visíveis somente as iniciais. Dessa forma, por se tratar de um ranking geral, com grande volume de inscrições, é possível garantir um bom nível de anonimização, ao mesmo tempo em que se mantém a integridade e a transparência do ranking.

Durante essa etapa do projeto, o maior desafio encontrado foi definir a estratégia e lógica necessárias para atribuir as medalhas aos usuários de acordo com suas atividades. A complexidade estava em garantir que cada usuário recebesse as medalhas corretas de acordo com as ações específicas que ele executava no sistema.

Para superar esse desafio, a solução encontrada foi a criação de uma tabela de log centralizado, que registra todas as atividades executadas pelos usuários na aplicação. Essa tabela funciona como uma fonte de informação em tempo real sobre as cada ação realizada, permitindo que o sistema verifique continuamente se o usuário atende aos critérios para receber uma determinada medalha. Dessa forma,

sempre que o usuário registra uma ação, o sistema realiza uma verificação para avaliar se ele cumpriu os requisitos necessários para obter uma nova medalha.

3.4.3 DESENVOLVIMENTO MOBILE

A partir dos estudos de ferramentas e da criação de um protótipo de alta fidelidade das telas e interfaces esperadas pelo aplicativo, foi possível iniciar o desenvolvimento do código do aplicativo *mobile*.

Para uma rápida escalada nas funcionalidades dentro do cronograma estipulado, optou-se pelo uso do Javascript junto ao React Native em seu estado mais puro, com uso mínimo de bibliotecas e frameworks adicionais que fossem capazes de modificar as lógicas mais críticas de funcionamento da aplicação, garantindo assim a escalabilidade e fácil manutenibilidade no longo prazo.

Para o tratamento e gerenciamento de estado de uma sessão no aplicativo foram avaliadas diversas possibilidades, como o uso da biblioteca Redux, utilizada globalmente por grandes empresas em aplicativos criados nas primeiras versões do React Native, a biblioteca Recoil, criada recentemente para extrair o máximo da organização de funções por meio de Hooks, incorporado na versão 0.58, e o uso das ferramentas próprias do framework, conhecida como Context API. A escolha final, amparada pelo Professor Orientador, se baseou na comparação da curva de aprendizado das bibliotecas de terceiros e de ferramentas próprias da linguagem, que tem melhor adoção no mercado, optando-se assim, pelo uso da Context API.

Figura 15 - Bloco de código do arquivo ClassesContext, parte integrante da lógica de gerenciamento de estado via Context API.

```
const setClassesList = async (list) => {
  dispatch({ type: "setClasses", payload: list });
};

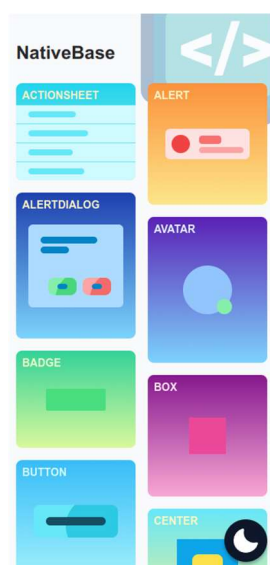
return (
  <ClassesContext.Provider
    value={{
      classesState,
      setClassesList,
    }}
  >
    {children}
  </ClassesContext.Provider>
);

export { ClassesContext, ClassesProvider };
```

FONTE: Os autores (2022)

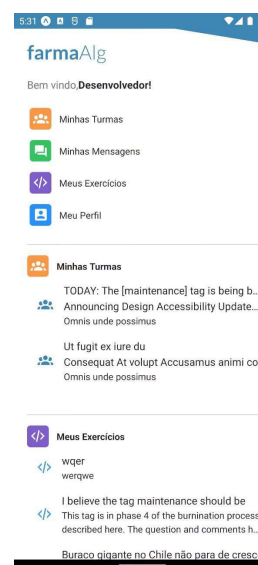
Por sua vez, a biblioteca NativeBase foi a escolhida para a criação de interfaces gráficas, formulários, botões e posicionamento, motivada principalmente por sua popularidade dentro da comunidade global de desenvolvedores de React Native. Essa escolha permitiu um desenvolvimento de todas as telas em tempo satisfatório, incluindo aquelas com designs mais complexos e maior quantidade de funcionalidades inseridas.

Figura 16 - Aplicativo de demonstração da biblioteca NativeBase



FONTE: Os autores (2023)

Figura 17 - Tela inicial do FarmaALG desenvolvida para aplicativos móveis



FONTE: Os autores (2023)

O maior desafio das etapas de coleta de requisitos e prototipação foi a funcionalidade principal do aplicativo *mobile*, envolvendo a escrita e edição de código com destaque de texto conforme a linguagem selecionada (*highlight*) com experiência de usuário boa e com interface de fácil entendimento mesmo com discentes mais novos dentro da Universidade.

A solução encontrada foi o uso da biblioteca open-source react-native-code-editor, desenvolvida por RIVAS (2022), permitindo a personalização do editor de texto conforme a seleção do usuário e o pacote de temas definido pelo aplicativo. O resultado, após a customização realizada, foi de uma experiência semelhante a uma IDE simples e moderna como o software Atom, mesmo em ambiente *mobile*.

Figura 18 - Classe Java escrita dentro do software de edição de texto Atom

```

1 <?php
2 class Books{
3
4     public $price;
5     public $title;
6
7     public function setPrice($par){
8         $this->price = $par;
9     }
10
11    public function getPrice(){
12        echo $this->price;
13    }
14 }
15
16 $book = new Books();

```

Figura 19 - Código C escrito na tela de edição de código na versão mobile do FarmaALG

```

5:32
←
1 #include <stdlib.h>
2 #include <stdio.h>
3
4 int main()
5 {
6     char nome[255];
7     scanf("%s", nome);
8     printf("hello, %s!\n", nome);
9
10    return 0;
11 }

```

FONTE: Os autores (2023)

A exibição de Casos de Teste na aplicação, a entrada a ser executada no código, a saída esperada e seus resultados são parte primordial da ferramenta FarmaALG e devem levar a rápida compreensão de suas informações, conforme apontado por KUTZKE (2015):

A visualização dos dados pertinentes à mediação do erro é um ponto crucial no arcabouço. [...] É preciso que tenham instrumentos para que a visualização desses dados possa levar à produção de conhecimento sobre estes.

Por se tratar de uma preocupação permanente em todas as versões do FarmaALG até o presente momento, durante o desenvolvimento da aplicação *mobile* foi aplicado uma série de cuidados para garantir a exibição adequada desses dados nas respectivas telas de manipulação de exercícios, utilizando cores conhecidas para destacar resultados da execução e a exibição em várias áreas dos casos de teste e linguagens configuradas para o exercício.

Figura 20 - Tela do exercício selecionado, com enunciado, linguagens disponíveis para solução e casos de teste esperados

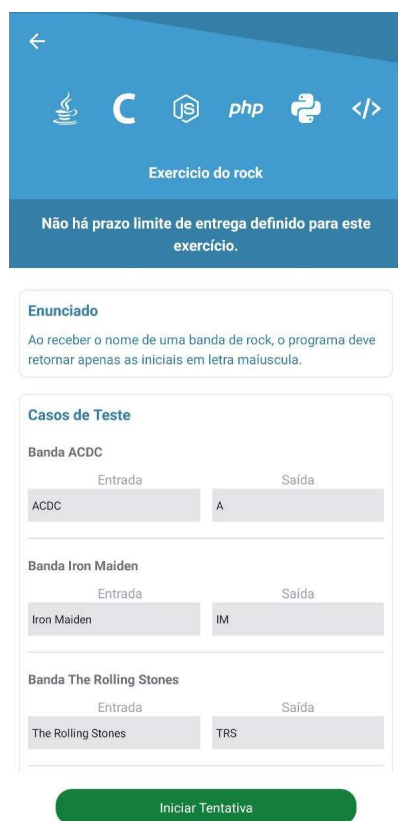


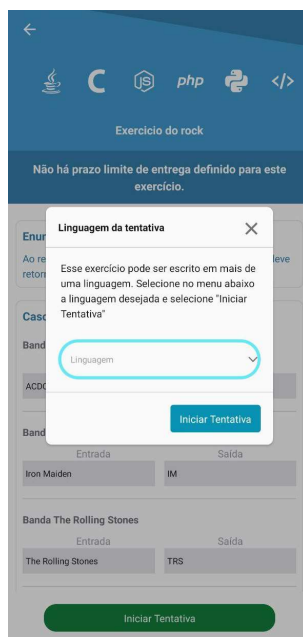
Figura 21 - Tela de execução do exercício, listando resultados dos casos de teste com entrada executada, saída esperada e obtida



FONTE: Os autores (2023)

Para a escrita do exercício dentro do aplicativo, caso o professor tenha ativado essa opção, o discente possui a possibilidade de escolher a linguagem na qual deseja desenvolver o código, essa seleção é realizada em um modal anterior à tela de escrita de código, que permite a exibição de um código exemplo da linguagem escolhida ao aluno e a personalização do *highlight* do editor por meio da lógica do aplicativo.

Figura 22 - Modal de seleção de linguagem para resposta do exercício



FONTE: Os autores (2023)

3.4 MODELAGEM DO PROJETO

3.4.1 DIAGRAMA DE CASOS DE USO

Os diagramas de caso de uso são úteis para a visualização e comunicação dos requisitos funcionais de um sistema, permitindo que os desenvolvedores, analistas e stakeholders entendam e discutam as principais funcionalidades do sistema de forma clara e concisa.

Para os autores Lima (2011) e Booch, Rumbaugh e Jacobson (2005) o diagrama de casos de uso tem um papel central na modelagem do comportamento de um sistema, representando toda a funcionalidade de um sistema, evidenciando interações externas com outras entidades. O Diagrama de Casos de Uso do sistema FarmaALG versão web pode ser visualizado no APÊNDICE A e da versão mobile no APÊNDICE B, essa segregação ocorre por conta de algumas funcionalidades existirem na versão web, mas não estarem presentes na versão mobile.

3.4.3 DIAGRAMA DE CLASSES

O Diagrama de Classes tem como finalidade apresentar um conjunto de classes, interfaces e colaborações mostrando o relacionamento entre eles.

Utilizam-se os diagramas de classe para exibir as classes do sistema, as relações entre elas e os principais atributos e métodos delas. Esse tipo de diagrama auxilia na implementação de softwares com orientação à objetos (AMBLER, 2005). O Diagrama de Classes do sistema FarmaALG pode ser visualizado no APÊNDICE C, vale ressaltar que algumas classes permaneceram com a mesma estrutura, mas tiveram suas implementações alteradas para atender às novas regras de negócio.

3.4.4 DIAGRAMA ENTIDADE E RELACIONAMENTO (DER)

Um diagrama de entidade e relacionamento é uma técnica de modelagem utilizada para representar a estrutura de um banco de dados de forma visual. Ele descreve as entidades (objetos) envolvidas no sistema, os relacionamentos entre essas entidades e os atributos que as caracterizam. O DER do sistema FarmaALG versão mobile está no APÊNDICE D.

3.4.5 DIAGRAMA DE SEQUÊNCIA

O diagrama de sequência fornece uma visão clara do fluxo de interações entre os objetos em um sistema, permitindo identificar possíveis falhas de comunicação, verificar a sequência de eventos e entender o comportamento do sistema em diferentes cenários.

No diagrama de sequência, representa-se uma série de interações entre objetos, descrevendo-se a mensagem passada entre eles e a ordem (no tempo) em que elas são enviadas. Para isso, utiliza-se a dimensão vertical do diagrama para o tempo e a horizontal, para a participação dos objetos na interação (RUMBAUGH, JACOBSON e BOOCH, 2005).

Para representação dos fluxos dos Casos de Uso (APÊNDICE B) que foram detalhados na especificação de casos de uso (APÊNDICE F), foram utilizados diagramas de sequência, que estão apresentados no APÊNDICE E.

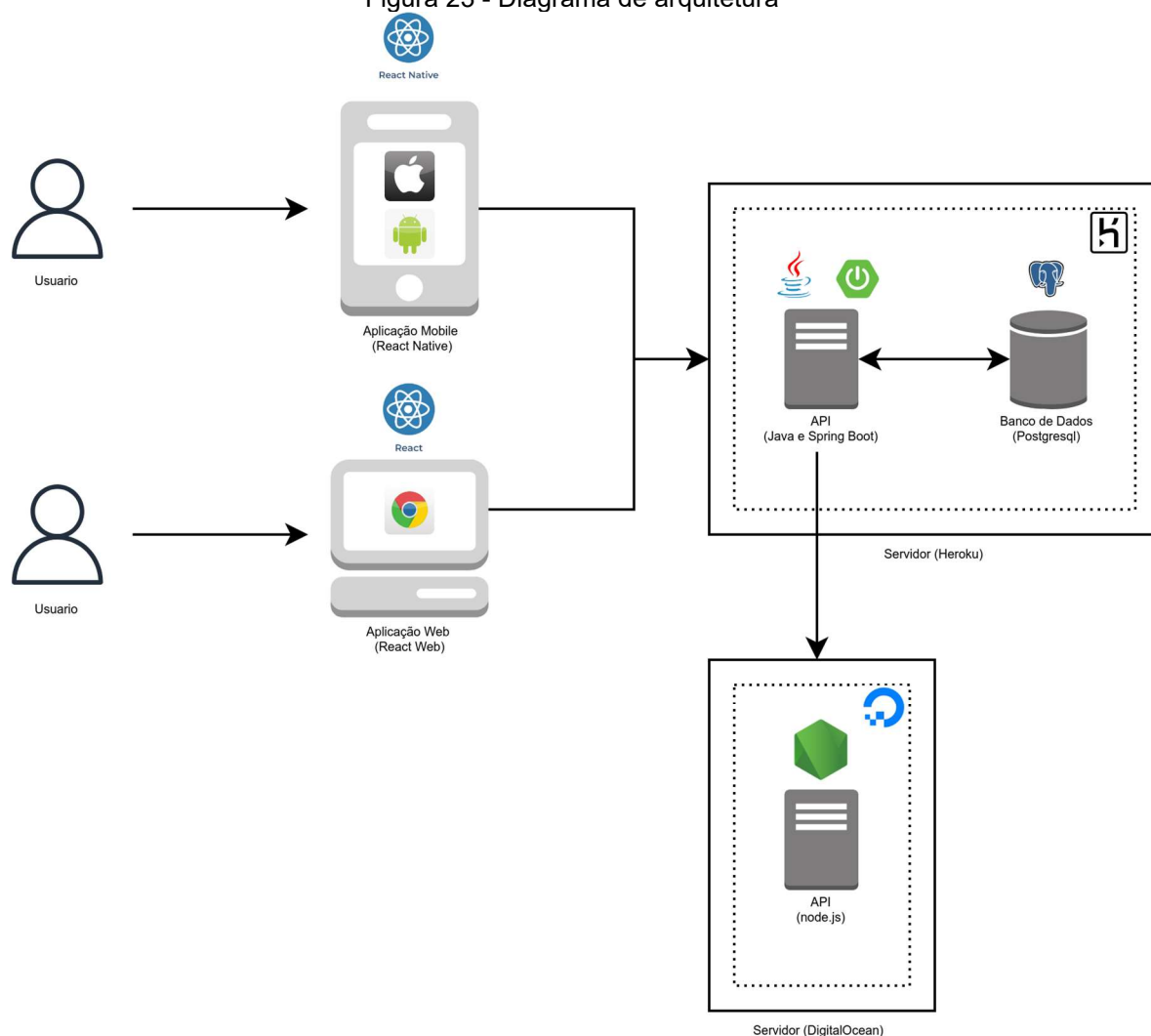
4. APRESENTAÇÃO DO SISTEMA

4.1 ARQUITETURA DO SISTEMA

A equipe manteve as tecnologias e métodos de desenvolvimento realizados por MUELLER (2021) na refatoração do projeto visando uma implementação focada em microsserviços. Diante disso, diversas metodologias utilizadas pela equipe anterior foram mantidas e ampliadas, como o uso do padrão de microsserviços, onde as funcionalidades principais do projeto são divididas para um desenvolvimento escalável, uniforme e modular.

Na abordagem escolhida para esse trabalho de conclusão, o sistema foi segmentado em cinco partes, sendo elas a aplicação mobile para alunos, a aplicação mobile para docentes, a aplicação web para alunos e professores, o servidor back-end e o servidor sandbox para a execução de códigos, as últimas três partes possuindo base de código anterior já desenvolvida pela equipe de 2021.

Figura 23 - Diagrama de arquitetura



FONTE: Os autores (2023)

O projeto web e mobile são desenvolvidos utilizando as bibliotecas React e React Native, respectivamente. A escolha do React Native no mobile foi baseada por sua capacidade de desenvolvimento de aplicativos móveis para os dois principais sistemas operacionais Android e iOS utilizando uma única base de código e sua semelhança semântica perante o React, o que permitiu uma maior curva de aprendizado no desenvolvimento simultâneo dos projetos web e mobile.

No lado do servidor, o back-end é implementado utilizando a linguagem Java e o framework Spring Boot, que traz uma série de pré-configurações de servidores, containerização e manipulação de dados por REST. A API gerencia as solicitações recebidas do front-end, processa a lógica e se conecta ao banco de dados PostgreSQL para executar operações como login, busca de informações de usuário, salvar respostas de exercícios e demais tarefas relacionadas ao armazenamento de dados.

Para a execução dos exercícios, a API principal se comunica com uma segunda API construída em Node.js. A API Node.js é responsável por executar os exercícios, recebendo as solicitações da API Java/Spring Boot e executando dentro de uma sandbox de forma segura, de forma que caso um código complexo e mal otimizado seja enviado para execução como resposta de um exercício, ele não afetará o desempenho da API principal, permitindo assim, o tratamento personalizado das exceções de falha.

Em termos de hospedagem, o front-end, desenvolvido em React e a API principal, desenvolvida em Java/Spring Boot, é hospedada na plataforma *Heroku* em dois contêineres distintos do tipo *Heroku Eco Dyno*, com 512 MB de memória RAM e um compartilhamento de CPU flexível de 1x a 4x, sendo suficientes para que cada aplicação tenha o desempenho esperado.

Por sua vez, a API de sandbox é hospedada em um servidor do tipo VPS (Virtual Private Server) da DigitalOcean. O container do Docker é executado em uma máquina virtual com 1 GB de memória, 25 GB de espaço em disco e configurado com a imagem do Ubuntu 22.04 (LTS). Essas especificações fornecem recursos suficientes para executar e dimensionar a API de forma eficiente, mesmo em caso de múltiplas requisições simultâneas ou a execução de código com lógica problemática.

4.2 FUNCIONALIDADES

A presente seção tem como objetivo fornecer uma abordagem clara, objetiva e minuciosa acerca das funcionalidades das implementações realizadas pela equipe dentro da versão web do FarmaALG e dos dois aplicativos, Docente e Aluno. Por meio de um formato de tutorial descritivo, será realizada uma exposição minuciosa das principais características e operações proporcionadas para o usuário.

Na interface de abertura tanto do aplicativo FarmaALG Aluno quanto do FarmaALG Docente, são apresentados dois botões principais com suas respectivas funcionalidades. O primeiro botão, intitulado "Entrar", tem como objetivo redirecionar o usuário para a tela de login. Já o segundo botão, com o texto "Registrar", tem a finalidade de direcionar o usuário para a tela de cadastro. Vale ressaltar que, no contexto do aplicativo FarmaALG Aluno, somente os alunos possuem permissão para efetuar o registro por meio dessa funcionalidade. Por outro lado, no aplicativo FarmaALG Docente, apenas os professores têm acesso à possibilidade de se cadastrar através deste botão.

Figura 24 - Tela de início do App Aluno



Figura 25 - Tela de Entrar do App Aluno

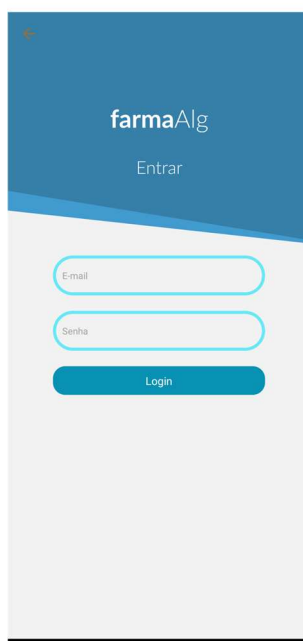
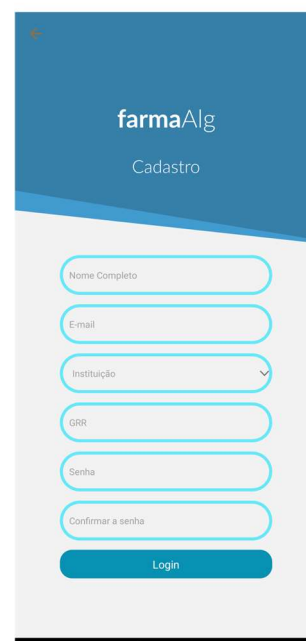


Figura 26 - Tela de Cadastro do App Aluno



FONTE: Os autores (2023)

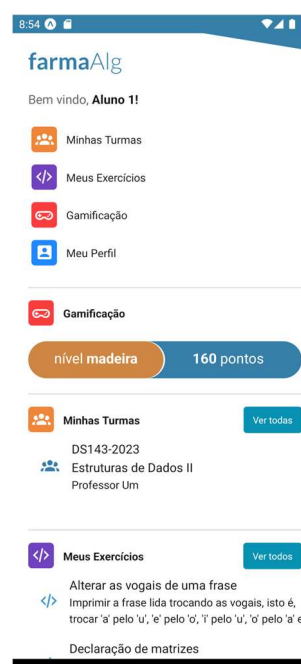
Na tela de login, tanto no aplicativo FarmaALG Aluno quanto no FarmaALG Docente, são apresentados dois campos obrigatórios: "e-mail" e "senha". Além disso, há um botão de "login" que permite o acesso ao sistema. Essa tela é compartilhada pelos dois aplicativos e conta com validações para garantir que os campos sejam preenchidos corretamente. Caso haja campos inválidos ou em branco, a devida validação é realizada, alertando o usuário sobre a necessidade de correção.

A tela de cadastro nos aplicativos FarmaALG Aluno e FarmaALG Docente é diferenciada para atender às particularidades de cada perfil de usuário. Além dos campos obrigatórios comuns, como "nome", "e-mail", "instituição", "senha" e "confirmar senha", são incluídos campos adicionais específicos. No aplicativo FarmaALG Aluno, é solicitado o preenchimento do campo "GRR" (Número de Registro de Graduação), permitindo a identificação individualizada dos alunos. Já no aplicativo FarmaALG Docente, é disponibilizado o campo "matrícula institucional" para capturar o número de matrícula dos professores na instituição.

Figura 27 - Tela principal do App Docente



Figura 28 - Tela principal do App Aluno



FONTE: Os autores (2023)

Na tela principal do aplicativo, é apresentado um conjunto de atalhos no topo, compreendendo quatro principais funcionalidades: "Minhas Turmas", "Meus Exercícios", "Gamificação" e "Meu Perfil". Abaixo, são exibidas as três turmas mais

recentes na seção "Minhas Turmas" e os três exercícios mais recentes na seção "Meus Exercícios". Cada seção é acompanhada por um botão "Ver Todos", permitindo que o usuário seja redirecionado para a tela completa correspondente, exibindo todas as turmas ou exercícios disponíveis.

É importante destacar que, na exibição para alunos, apenas as turmas e exercícios relacionados às turmas nas quais o aluno está cadastrado serão exibidos. Por outro lado, na exibição para docentes, caso o professor não tenha exercícios próprios, os exercícios públicos criados por outros professores são exibidos.

Figura 29 - Tela de Gamificação do App Aluno



FONTE: Os autores (2023)

No aplicativo FarmaALG Aluno, uma funcionalidade adicional é a exibição de um resumo da gamificação, que inclui o nível alcançado pelo aluno e sua pontuação atual, trazendo uma perspectiva visual do seu progresso. Ao clicar nesse resumo ou no atalho correspondente na área principal, o usuário é redirecionado para a tela principal da gamificação. Nessa tela, o header destaca o nível atual do aluno, sua posição no ranking geral, o saldo de pontos obtidos e os pontos necessários para alcançar o próximo nível. O nome dos níveis foi determinado seguindo o padrão de aplicações gamificadas utilizando o nome de materiais da natureza, por ordem de valiosidade. Segue abaixo uma tabela com os níveis possíveis.

Tabela 1 - NÍVEIS E PONTUAÇÃO CORRESPONDENTE

Nível	Intervalo de Pontuação
Algodão	0 - 130
Madeira	130 - 200
Ferro	200 - 400
Aço	400 - 800
Cobre	800 - 1300
Bronze	1300 - 1800
Prata	1800 - 2300
Ouro	2300 - 2800
Platina	2800 - 3300
Diamante	+ 3300

FONTE: Os autores (2023)

Logo abaixo, são exibidos cards com as medalhas conquistadas pelo aluno. Também são apresentados os primeiros 10 alunos do ranking geral, com seus nomes anonimizados, exibindo somente as iniciais. Além disso, é disponibilizada uma lista com todas as medalhas possíveis, permitindo ao aluno conferir todas as conquistas disponíveis.

Ao clicar em uma medalha, um modal é exibido com informações detalhadas, como o nome da medalha, a atividade necessária para sua conquista, sua raridade e a pontuação associada a ela. As medalhas são separadas por categorias, existem algumas categorias diferentes para as medalhas, sendo a mais fácil de conseguir mais comum e menos valiosa. As quantidades de ações necessárias para conclusão de cada medalha seguem um padrão para todas elas, indo de uma ação até cinquenta. A aquisição de todas as medalhas concede ao usuário o nível mais alto que é o “Diamante”. Segue abaixo a tabela de medalhas com suas respectivas pontuações.

Tabela 2 - CONQUISTAS, PONTUAÇÃO E RARIDADE

Nome da conquista	Categoria	Pontuação	Raridade
1 exercício respondido	Exercícios	10	COMUM
3 exercícios respondidos	Exercícios	30	INCOMUM
5 exercícios respondidos	Exercícios	50	RARO
10 exercícios respondidos	Exercícios	100	SUPER RARO
25 exercícios respondidos	Exercícios	250	ÉPICO
50 exercícios respondidos	Exercícios	1000	LENDÁRIO
1 acesso ao App	Acesso ao App	10	COMUM
3 acessos ao App	Acesso ao App	30	INCOMUM
5 acessos ao App	Acesso ao App	50	RARO
10 acessos ao App	Acesso ao App	100	SUPER RARO
25 acessos ao App	Acesso ao App	250	ÉPICO
50 acessos ao App	Acesso ao App	1000	LENDÁRIO
1 Tentativa de resposta a exercícios	Tentativas	10	COMUM
3 Tentativa de resposta a exercícios	Tentativas	30	INCOMUM
5 Tentativa de resposta a exercícios	Tentativas	50	RARO
10 Tentativa de resposta a exercícios	Tentativas	100	SUPER RARO
25 Tentativa de resposta a exercícios	Tentativas	250	ÉPICO
50 Tentativa de resposta a exercícios	Tentativas	1000	LENDÁRIO
1 disciplina matriculada	Disciplinas	100	SUPER RARO

3 disciplinas matriculadas	Disciplinas	250	ÉPICO
5 disciplinas matriculadas	Disciplinas	1000	LENDÁRIO
Exercício resolvido corretamente	Exercícios	10	COMUM

FONTE: Os autores (2023)

Figura 30 - Tela de Gamificação no App

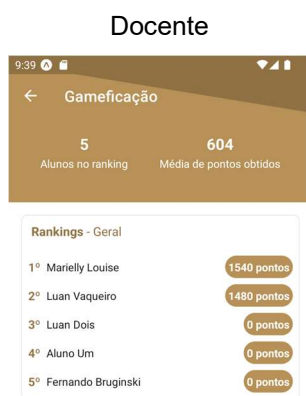


Figura 31 - Tela de Minhas Turmas no App



FONTE: Os autores (2023)

No aplicativo do professor, no topo da tela de gamificação é apresentada a quantidade de alunos presentes no ranking, a média de pontos obtidos por esses alunos e uma listagem do ranking com as pontuações de cada aluno.

Na tela "Minhas Turmas", no App para discentes, são exibidas as turmas em que o aluno se registrou. As turmas são apresentadas em uma listagem com o nome da disciplina, turma e nome do professor responsável. No topo da tela, é disponibilizado um botão "Registrar-se em Nova Turma" que aciona um modal para inserção do código de acesso da turma, definido pelo professor na criação da turma.

Figura 32 - Tela de Minhas Turmas no App

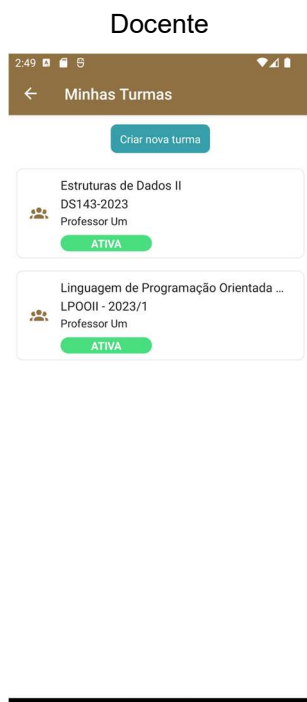
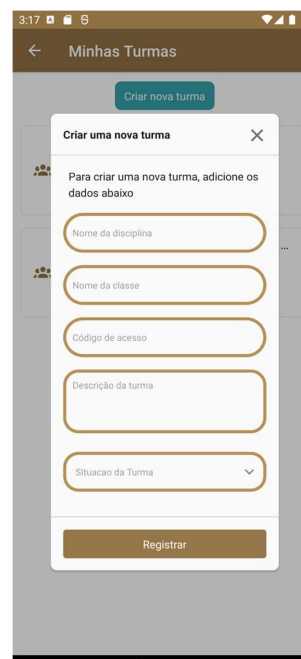


Figura 33 - Modal de Criação de Nova Turma no App Docente



FONTE: Os autores (2023)

Dentro do aplicativo para docentes, na tela "Minhas Turmas", são exibidas todas as turmas criadas pelo professor. Cada turma é apresentada com o status atual, que pode ser "Ativa", "Encerrada", "Arquivada" ou "Pausada". No modal de criação de uma nova turma, disparado pelo clique no botão "criar nova turma", são apresentados os campos "Nome da disciplina", "Nome da classe", "Código de acesso", "Descrição da turma" e "Situação", que pode ser selecionado entre "Ativa", "Encerrada", "Arquivada" ou "Pausada", todos os campos são obrigatórios para a criação de turma.

Figura 34 - Tela de Turma no App Aluno

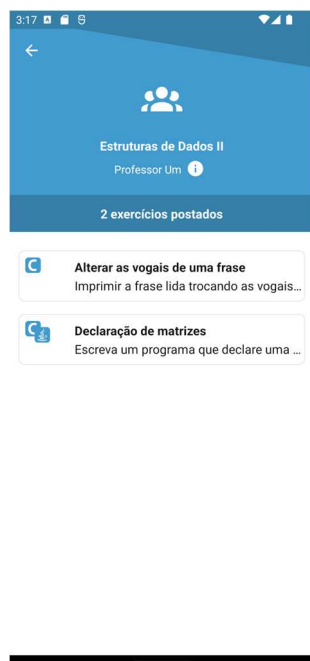


Figura 35 - Modal de dados do professor no App Aluno

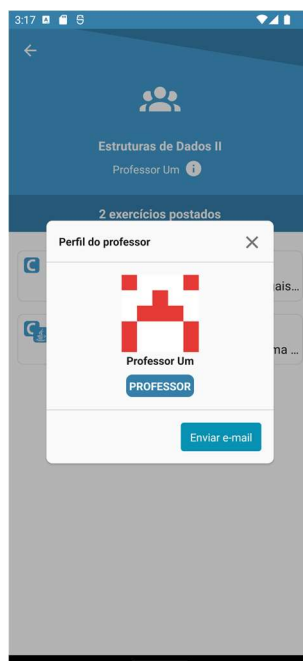
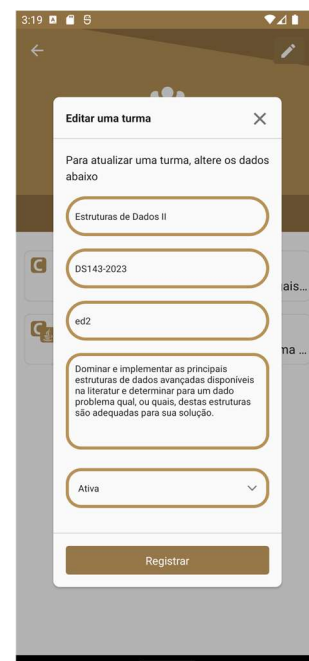


Figura 36 - Edição de turma no App Aluno



FONTE: Os autores (2023)

A tela da turma selecionada, nos dois aplicativos, exibe o nome da turma, nome do professor responsável e a quantidade de exercícios postados. Além disso, é apresentada uma listagem dos exercícios disponíveis, indicando as linguagens permitidas para as respostas e a descrição de cada exercício.

Ao clicar ao lado do nome do professor na aplicação para alunos, um modal é exibido, contendo um botão "Enviar E-mail". Ao acionar esse botão, o cliente de e-mail instalado no dispositivo móvel é aberto, permitindo ao aluno enviar uma mensagem diretamente para o professor.

Exclusivamente na aplicação para professores, é disponibilizado um botão com o ícone de lápis, que permite a edição dos dados de uma turma. Por meio desse botão, um modal é aberto, na qual é possível modificar informações como o nome, código de acesso ou status da turma, conforme necessário.

Figura 37 - Tela de Meus Exercícios no App

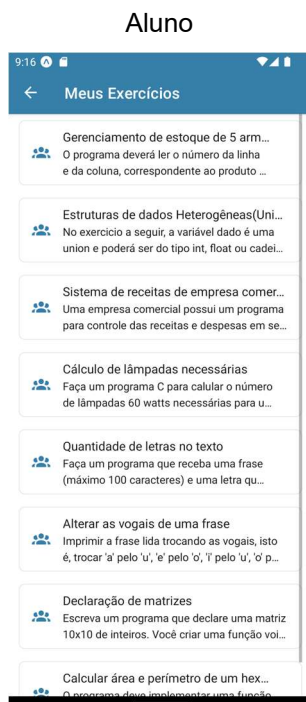
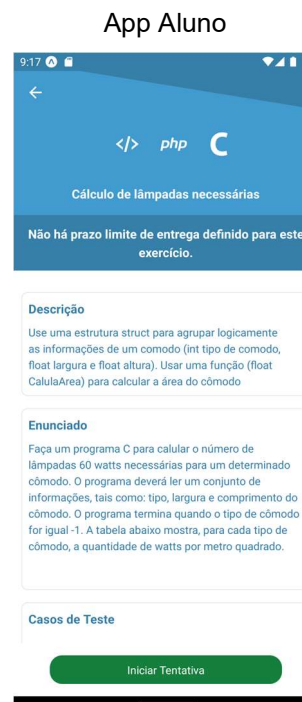


Figura 38 - Tela de Exercício Selecionado no



FONTE: Os autores (2023)

Na tela de "Meus Exercícios", disponível tanto para alunos quanto para professores, é possível acessar os exercícios relacionados ao perfil do usuário. No caso dos alunos, são exibidos todos os exercícios das disciplinas em que estão matriculados, apresentando o nome e enunciado de cada exercício. Já para os professores, são exibidos os exercícios que eles criaram, bem como aqueles que foram configurados com visibilidade pública. O professor tem a opção de filtrar os exercícios exibidos, permitindo visualizar apenas os exercícios criados por ele mesmo.

Figura 39 - Tela de Criação de Exercício no App Docente



9:41

← Criar Exercício

Para criar um novo exercício, preencha com atenção os dados abaixo:

Escolha a turma

Título do exercício

Descrição do exercício

Enunciado

Configurar linguagens

Configurar tags

Configurar dicas

Exercício visível para outros professores (Público)

Registrar

FONTE: Os autores (2023)

Dentro do aplicativo para professor, ao acessar a tela de "Meus Exercícios", é possível criar um exercício por meio do botão "Criar Novo Exercício". Ao clicar nesse botão, o usuário é redirecionado para a tela de criação de exercícios, onde são apresentados campos obrigatórios como a escolha da turma, título do exercício, descrição, enunciado e configurações de linguagens através de um modal e botões para acessar a tela de criação, edição e exclusão de dicas e *tags*.

Figura 40 - Tela de Exercício no App Aluno

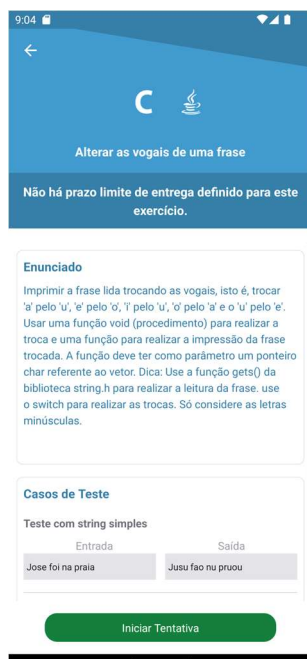
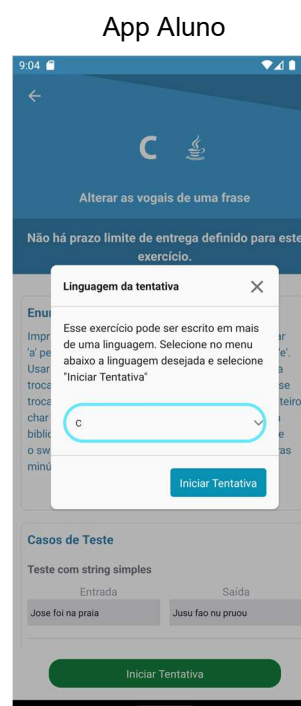


Figura 41 - Modal de Troca de Linguagem no App Aluno



FONTE: Os autores (2023)

Na tela de exercício, dentro do header são apresentadas as linguagens permitidas para a resposta do exercício e prazo limite de entrega do exercício, caso exista. Em seguida, são exibidos cards contendo a descrição do exercício, o enunciado do problema a ser resolvido e os casos de teste associados a esse exercício.

Na aplicação do aluno, é exibido um botão fixo denominado "Iniciar Tentativa" na tela de exercício. Esse botão tem como finalidade permitir que o aluno comece a resolver o exercício selecionado. Caso o exercício possua mais de uma linguagem disponível para a resposta, é exibido um modal de seleção de linguagem. Esse modal apresenta ao aluno as opções de linguagens disponíveis para aquele exercício, definidas pelo professor na criação, permitindo que o discente escolha aquela na qual deseja escrever sua solução.

Figura 42 - Tela de Edição de Exercício no App Docente

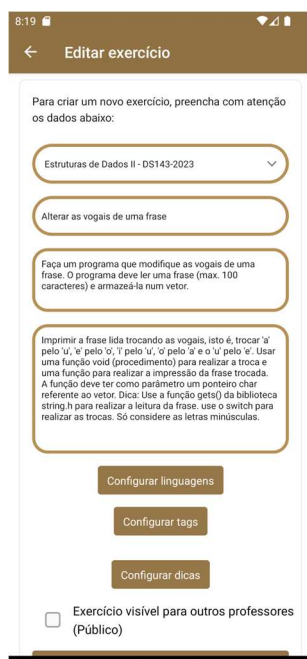


Figura 43 - Tela de visualização de alunos respondentes no App Docente



FONTE: Os autores (2023)

Dentro da aplicação docente, na tela do editor de código, juntamente com o cabeçalho, são exibidos dois botões adicionais. O primeiro botão, representado pelo ícone de lápis, permite ao professor acessar a tela de edição do exercício. Nessa tela, o professor tem a capacidade de modificar diversos aspectos do exercício, como a turma à qual o exercício está atribuído, o nome, o enunciado, a descrição, as linguagens permitidas, as *tags* associadas, as dicas para os alunos e a visibilidade pública do exercício. Na parte inferior da tela, são exibidos dois botões fixos, o primeiro direciona o usuário para a jornada de edição e execução de casos de teste. O segundo botão, por sua vez, redireciona o usuário para a visualização da listagem de alunos respondentes e não respondentes.

Após o clique em "Ver Respostas", o professor é redirecionado para a tela de alunos, onde é possível visualizar a lista completa de alunos e o status relacionado àquele exercício, indicando se já responderam, não responderam ou estão em processo de tentativa. Ao selecionar um aluno com resposta ou em tentativa, é possível verificar os detalhes da resposta, incluindo as tentativas realizadas, entradas e saídas do programa e, ao clicar em "ver código" é possível acessar tela de edição

de código é aberta em modo somente leitura para checagem do código completo submetido naquela tentativa pelo aluno.

Figura 44 - Tela de Edição de Código no App



Figura 45 - Modal de Troca de Linguagem no



FONTE: Os autores (2023)

Após o clique em "Editar e Executar Casos de Teste" no aplicativo do professor ou "Iniciar tentativa" no aplicativo do aluno, o usuário é redirecionado para a tela de edição de código. Nessa tela, um código de exemplo da primeira linguagem selecionada para o exercício é exibido. No topo da tela, são destacados dois botões, o ícone de "troca" permite a troca da linguagem de programação, caso o usuário deseje selecionar uma linguagem diferente para o editor. O segundo botão, com o ícone de "play", direciona o usuário para a tela de casos de teste.

Figura 46 - Tela de Edição de Código no App Aluno

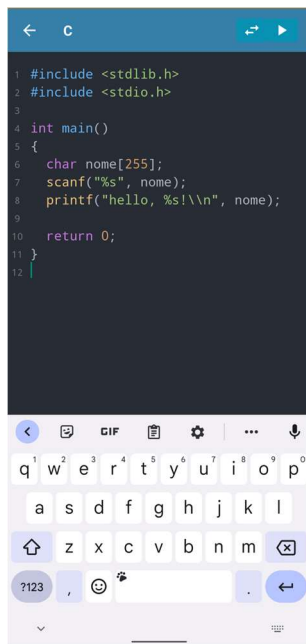


Figura 47 - Tela de Execução de Código (Aba Casos de Teste) no App Aluno

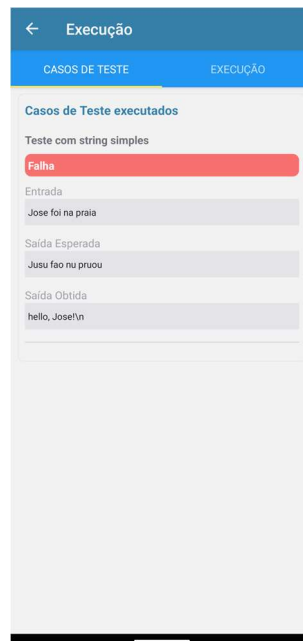
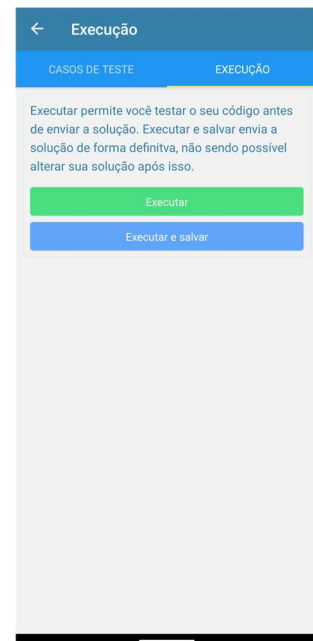


Figura 48 - Tela de Execução de Código (Aba Execução) no App Aluno



FONTE: Os autores (2023)

Após escrever o código solução, o aluno pode clicar no botão de "play", que o direciona para a tela de casos de teste. Nesta tela, serão exibidas duas abas importantes. Na aba "Casos de Teste", serão apresentados os casos de teste específicos do exercício em questão. Na aba "Execução", existem dois botões distintos. O primeiro botão permite executar os casos de teste com base no código desenvolvido pelo aluno na tela do editor de código e redireciona o usuário para a aba "casos de teste" com os resultados da tentativa em seus respectivos casos de teste. O segundo botão, executa o código e encerra a tentativa, enviando-a permanentemente, sem a possibilidade de edição posterior.

Figura 49 - Tela de Execução de Código (Aba Casos de Teste) no App Docente

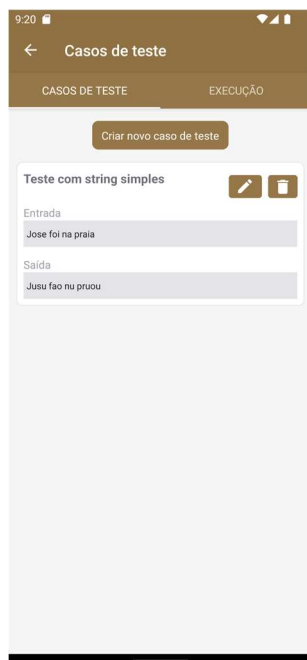
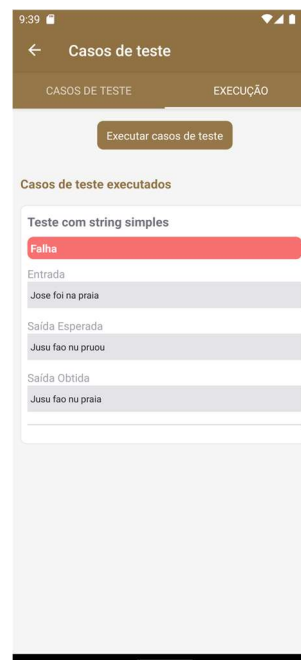


Figura 50 - Tela de Execução de Código (Aba Casos de Teste) no App Docente

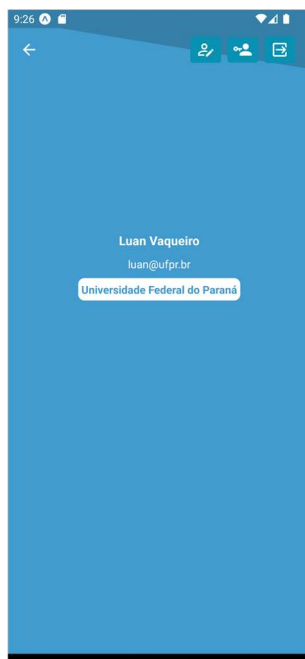


FONTE: Os autores (2023)

Dentro do aplicativo de professor, após a entrada na tela de casos de teste, são apresentadas duas abas distintas. Na primeira aba, chamada de "Casos de Teste", o professor tem a possibilidade de criar, editar e deletar casos de teste por meio de seletores e modais. Para cada caso de teste, é solicitado o nome, a entrada do código e a saída esperada. Na segunda aba, intitulada "Execução", são listados os casos de teste executados e seus respectivos resultados da última execução. Essa seção fornece ao professor informações detalhadas sobre os resultados obtidos pelos casos de teste.

Além disso, é disponibilizado um botão para "Executar Casos de Teste", com base no código que o professor definiu na tela anterior. Caso nenhum caso de teste tenha sido executado anteriormente, os casos de teste esperados são exibidos, permitindo ao professor verificar se são compatíveis com o código que será desenvolvido pelos alunos.

Figura 51 - Tela de Perfil de Aluno no App Aluno



FONTE: Os autores (2023)

Ao clicar no atalho "Meu Perfil" nos dois aplicativos, o usuário é direcionado para uma tela onde são exibidos os dados do usuário logado, como nome, e-mail e instituição. No topo desta tela, destacam-se três ícones, ao clicar no ícone de edição um modal de edição é aberto, permitindo ao usuário atualizar suas informações, como nome, e-mail e instituição. O ícone de senha leva o usuário ao modal de edição de senha. Por fim, o ícone de seta realiza o logout do aplicativo, encerrando a sessão do usuário e retornando à tela de login.

5. CONSIDERAÇÕES FINAIS

Ao longo deste trabalho, foi abordada a problemática da evasão de disciplinas de linguagens de programação nos cursos da UFPR, especialmente considerando o cenário de retorno ao presencial, no qual observamos uma comunidade acadêmica receosa e desmotivada. Nossa solução proposta foi aprimorar a plataforma FarmaALG, utilizada em disciplinas de Análise e Desenvolvimento de Sistemas, Ciência da Computação e Informática Biomédica, por meio de um processo de *redesign* e refatoração.

Analisamos as implementações anteriores realizadas pela equipe de MUELLER et al. (2021) e demos continuidade ao trabalho, aperfeiçoando a versão web já existente e desenvolvendo a versão mobile da plataforma FarmaALG. Utilizamos integralmente a área de execução de códigos (*sandbox*) e o back-end, garantindo uma experiência fluida e eficiente aos usuários. Com essas melhorias, esperamos mitigar a evasão de disciplinas de linguagens de programação, proporcionando um ambiente virtual atraente, interativo, acessível e seguro para os estudantes.

Durante o aprimoramento da plataforma, reconhecemos a importância de introduzir elementos de gamificação para engajar os alunos. Um dos principais pontos implementados foi um sistema de pontuação, no qual os alunos podem acumular pontos por meio de várias atividades, como se inscrever em disciplinas, realizar tentativas, entregar exercícios e acessar a plataforma regularmente. Esses pontos são utilizados para gerar medalhas personalizadas para cada aluno e contribuem para um ranking geral, no qual o aluno pode verificar sua posição em relação aos outros usuários. Essa abordagem visa recompensar o progresso e o engajamento dos alunos, tornando a aprendizagem mais divertida e envolvente.

Este trabalho também evidencia a importância do FarmaALG para a comunidade dos cursos de computação da UFPR. Reconhecemos o trabalho fundamental de MARCZAL (2014), KUTZKE (2015), EHRESMANN (2018) e MUELLER et al. (2021), que contribuíram para expandir o potencial da plataforma por meio do aprimoramento contínuo de suas funcionalidades dentro da própria universidade. O projeto FarmaALG é um exemplo concreto do poder transformador da UFPR no avanço da pesquisa e desenvolvimento na área de computação.

Como equipe, sentimo-nos satisfeitos com os resultados alcançados neste TCC. No entanto, reconhecemos que há espaço para aprimoramentos futuros. Sugerimos a realização de estudos adicionais para avaliar a efetividade da plataforma FarmaALG na redução da evasão de disciplinas de linguagens de programação, bem como a possibilidade de expandir seu uso para outros cursos e instituições de ensino.

Destacamos a identificação da ausência da funcionalidade de visualização de tentativas e do código submetido por alunos em um exercício dentro da implementação web realizada anteriormente, tendo sido identificada tardiamente pela equipe. Essa funcionalidade acabou sendo implementada apenas nos aplicativos mobile desenvolvidos no escopo desse projeto. Por sua vez, reconhecemos a importância da sugestão proposta no trabalho de MUELLER (2021), que sugere a implantação de um sistema de grafos aliado ao módulo de exibição de respostas ao docente, para possibilitar a visualização e análise das tentativas e códigos submetidos pelos alunos.

No campo da gamificação, podemos destacar a possibilidade de adição de pontuação personalizada por exercício. Essa abordagem permitiria recompensar os alunos de acordo com a dificuldade dos exercícios, incentivando-os a enfrentar desafios mais difíceis e conquistando pontuações mais altas.

Por fim, esperamos que este trabalho possa servir como ponto de partida para novas pesquisas e iniciativas que visem enfrentar os desafios relacionados à evasão de disciplinas de linguagens de programação. Acreditamos que, com o engajamento contínuo da comunidade acadêmica e o apoio institucional, podemos criar soluções cada vez mais eficazes e impactantes, impulsionando o progresso educacional e tecnológico em nossa sociedade.

6. REFERÊNCIAS BIBLIOGRÁFICAS

- AHMAD, Muhammad Ovais; MARKKULA, Jouni; OIVO, Markku. **Kanban in software development: A systematic literature review**. 2013 39th Euromicro Conference On Software Engineering And Advanced Applications, [s.l.], v. 1, n. 1, p.9-16, set. 2013. IEEE.
- AMBLER, S. W. **The Elements of UML 2.0 Style**. Cambridge: Cambridge University Press, 2005.
- ANDERSON, David J. **Kanban: successful evolutionary change for your technology business**. Seattle: Blue Hole, 2010.
- ANDERSON, D. J; CARMICHAEL, A. **Essential Kanban Condensed**. 2016. Disponível em: <https://ssusa.s3.amazonaws.com/c/307492866/media/5c37d23d07b67/EssentialKanban-Condensed-English.pdf>. Acesso em 20 jun. 2022.
- BARDAGI, M.; HUTZ, C. S. (2014). **Evasão universitária e serviços de apoio ao estudante: uma breve revisão da literatura brasileira**. *Psicologia Revista*, 14(2), 279–301.
- BOOCH, GRADY; RUMBAUGH J.; JACOBSON I.. **UML: guia do usuário**. 6ª Reimpressão. Rio de Janeiro: Elsevier, 2005.
- BRICKMAN, J. **Sensor Tower's Q2 2020 Data Digest: Global App Ecosystem Sets New Record with 37.8 Billion Downloads**. Londres, 01 jul. 2020. Disponível em: <https://sensortower.com/blog/q2-2020-data-digest>. Acesso em: 3 mai. 2022.
- CARD, S. K.; PIROLI, P.; MACKINLAY, J. D. **The cost-of-knowledge characteristic function: Display evaluation of direct-walk dynamic information visualizations**. CHI'94, ACM, p. 238–244, 1994.
- CONVERGÊNCIA DIGITAL. **Celulares representam 80% do tráfego digital**. São Paulo, 27 jan. 2020. Disponível em: <https://www.convergenciadigital.com.br/Internet-Movel/Celulares-representam-80%25-do-trafego-digital-52676.html>. Acesso em: 3 mai. 2022.
- DUOLINGO. **Informações sobre o Duolingo**. Disponível em: <https://pt.duolingo.com/info>. Acesso em: 28 mai. 2023.
- DECI, E. L.; VALLERAND R. J.; PALLETIER, L. G.; RYAN, R. M. (1991) **Motivation and Education: The Self-Determination Perspective**, *Educational Psychologist*, 26:3-4, 325-346, DOI: 10.1080/00461520.1991.9653137
- EHRESMANN, R. W. **Gamificação significativa na educação: uma proposta de ferramenta com enfoque no aspecto motivacional do aluno**. 101 f. Dissertação (Mestrado). Curitiba: Universidade Federal do Paraná, 2018. Disponível em: <https://acervodigital.ufpr.br/handle/1884/54903>. Acesso em 01 mai 2022.

FERREIRA, H. **Draw.io**: Desenhar diagramas nunca foi tão fácil. Pplware, 09 mar. 2015. Disponível em: <https://pplware.sapo.pt/internet/draw-io-desenhar-diagramas-nunca-foi-tao-facil/>. Acesso em: 28 de maio de 2023.

GIFE. **Pesquisa Datafolha aponta falta de motivação dos estudantes e dificuldade para estabelecer rotina de aprendizagem em casa**. São Paulo: 7 dez. 2020. Disponível em: <https://gife.org.br/pesquisa-datafolha-aponta-falta-de-motivacao-dos-estudantes-e-dificuldade-para-estabelecer-rotina-de-aprendizagem-em-casa/>. Acesso em: 2 maio 2022.

HOED, R. M. **Análise da evasão em cursos superiores: o caso da evasão em cursos superiores da área de Computação**. Dissertação (Mestrado em Ciência da Computação) – Universidade de Brasília, Brasília, 2016.

INSTITUTO NACIONAL DE ESTUDOS E PESQUISAS EDUCACIONAIS ANÍSIO TEIXEIRA. **Resumo técnico do Censo da Educação Superior 2020** [recurso eletrônico]. Brasília: Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira, 2022.

KARPINSKI, J; DEL MORO, N; CASTRO, M; LARA, L. **Fatores críticos para o sucesso de um curso em EAD: a percepção dos acadêmicos**. São Paulo: jul. 2017. Disponível em: <https://doi.org/10.1590/S1414-40772017000200010>. Acesso em: 13 abr. 2022.

KHAN ACADEMY. **Sobre a Khan Academy**. Disponível em: <https://pt.khanacademy.org/about>. Acesso em: 28 mai. 2023.

KUTZKE, R. A. **Informática educacional e a mediação do erro na educação: um estudo teórico-crítico e uma proposta de instrumento computacional**, Curitiba. 2015. 197 f. Tese (Doutorado em Informática). Curitiba: Setor de Ciências Exatas, Universidade Federal do Paraná, 2015. Disponível em: <https://acervodigital.ufpr.br/handle/1884/40907>. Acesso em: 01 mai. 2022.

LIMA, A. S. **UML 2.3 do requisito à solução**. 1º edição. São Paulo: Érica, 2011.

LOPES, M. **O que é Figma e como usar?** EBAC Online. Disponível em: <https://ebaonline.com.br/blog/o-que-e-figma-e-como-usar#title1>. Acesso em: 30 abr. 2023.

MACKINLAY, J. **UIST '92**: Proceedings of the 5th annual ACM symposium on User interface software and technology. Monterey: ACM Digital Library, 1992. Disponível em: <https://dl.acm.org/doi/proceedings/10.1145/142621>. Acesso em: 06 mai. 2022.

MARCZAL, D. **FARMA**: Uma ferramenta de autoria para objetos de aprendizagem de conceitos matemáticos. Tese (Doutorado em Informática). Curitiba: Universidade Federal do Paraná - Departamento de Informática, 2014.

MEDEIROS, C. P. de; MASCARENHAS, I. P.; SILVA, T. L. F. da; SOUSA, E. C. **A motivação no ensino superior: um estudo com alunos dos cursos de administração e direito**. Revista Brasileira de Educação Superior, São Paulo, v. 2, n. 1, p. 45-58, jan./jun. 2014.

MUELLER, J. L.; ALFLEN, L.; SALES, R. Z. L; **FarmaALG**: Uma implementação utilizando uma abordagem de microsserviços. 127 f. Dissertação (Conclusão de Curso). Curitiba: Setor de Educação Profissional e Tecnológica, Universidade Federal do Paraná, 2021. Disponível em: <https://www.acervodigital.ufpr.br/handle/1884/73093>. Acesso em 01 mai 2022.

MUNTEAN, C. I. **Raising engagement in e-learning through gamification**. The 6th International Conference on Virtual Learning ICVL, 2011. Disponível em: https://www.academia.edu/36387149/Raising_engagement_in_e_learning_through_gamification. Acesso em 01 mai 2022.

MUSSLINER, B. O. **O problema da evasão universitária no sistema público de ensino superior**: uma proposta de ação com base na atuação de uma equipe multidisciplinar. BJD, São José dos Pinhais, v. 7, n. 4, p. 42674 - 42692, 28 abr. 2021. Disponível em: <https://www.brazilianjournals.com/index.php/BRJD/article/download/28957/22870>. Acesso em: 2 maio 2022.

NATIVE BASE. **A Brief History of NativeBase**. 2023. Disponível em: <https://docs.nativebase.io/getting-started>. Acesso em: 07 ago. 2022.

ONU. **2,9 bilhões de pessoas nunca acessaram a internet**. Disponível em: <https://brasil.un.org/pt-br/161450-29-bilh%C3%B5es-de-pessoas-nunca-acessaram-internet>. Acesso em: 5 jun. 2023.

POSTGRESQL. **About: What is PostgreSQL?** 2022. Disponível em: <https://www.postgresql.org/about>. Acesso em: 07 ago. 2022.

RIVAS, C. **React-native-code-editor**. Disponível em: <https://github.com/RivasCVA/react-native-code-editor>. Acesso em: 02 mai. 2022.

ROCHA, L. **Geekie Games**: saiba como tudo começou. Disponível em: <https://www.geekie.com.br/blog/geekie-games-saiba-como-tudo-comecou>. Acesso em: 20 mai. 2023.

SCHWABER, K; SUTHERLAND, J. **O Guia do Scrum**: o guia definitivo para o scrum: as regras do jogo. Edição Online: Scrumguides.Org, 2020. 16 p. Disponível em: <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-GuidePortugueseBR-3.0.pdf>. Acesso em: 21 jul. 2022

SILVA, Thalita Lima Ferreira da et al. **A MOTIVAÇÃO NO ENSINO SUPERIOR: UM ESTUDO COM ALUNOS DOS CURSOS DE ADMINISTRAÇÃO E DIREITO**. Revista Gestão em Análise, v. 3, n. 1/2, p. 104-113, abr. 2016. Disponível em: <https://periodicos.unichristus.edu.br/gestao/article/view/151>. Acesso em: 02 maio 2022.

SJARANCIO, S. **React.js**: A Brief History. Medium, jan. 2019. Disponível em: <https://medium.com/@sjarancio/reactjs-a-brief-history-3c1e969a477f>. Acesso em: 02 out. 2022.

SPRING BOOT DOCS. **Spring Boot**. 2023. Disponível em: <https://spring.io/projects/spring-boot>. Acesso em: 02 out. 2022.

STRMEČKI, D.; BERNIK, A.; RADOŠEVIĆ, D. **Gamification in E-Learning: Introducing Gamified Design Elements into E-Learning Systems.** Journal of Computer Sciences 2015, p. 1108–1117, 12 dez. 2015.

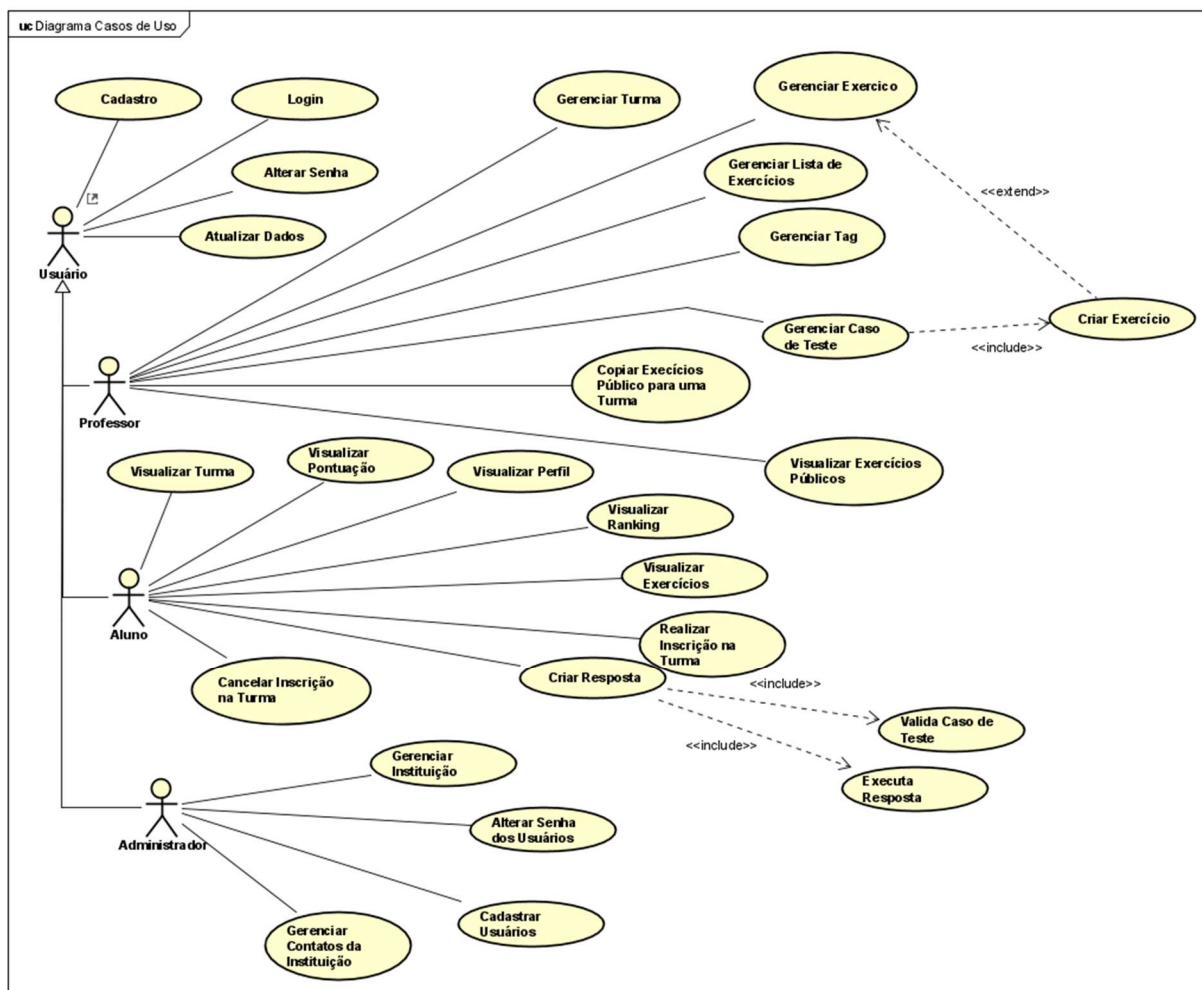
VIANNA, Y. **Gamification, Inc. Como reinventar empresas a partir de jogos.** MJV Press, 2013.

ZAGATTI, P. **Uso de gamificação em ambientes virtuais de aprendizagem para reduzir o problema da externalização de comportamentos indesejáveis.** São Paulo, 2016.

ZICHERMANN, G. **Gamification by Design.** O'Reilly, 2011.

APÊNDICE A: DIAGRAMA DE CASOS DE USO WEB

Figura 52 - Diagrama de Casos de Uso Web



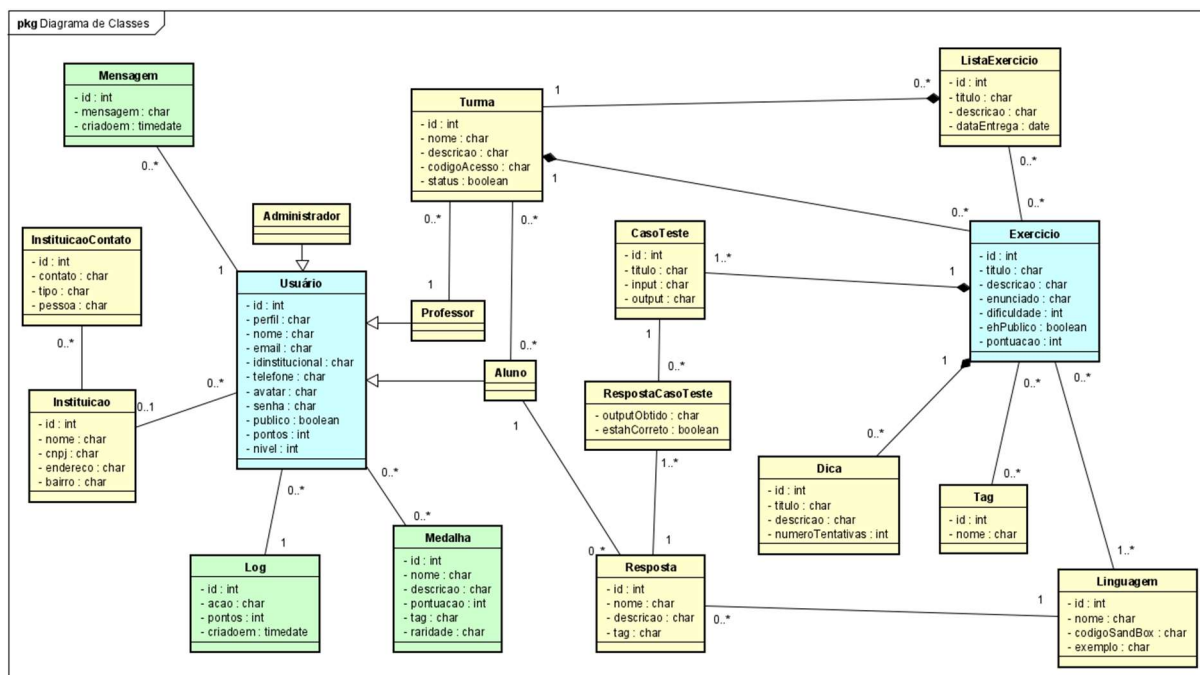
FONTE: Os autores (2023)

APÊNDICE C: DIAGRAMA DE CLASSES

Legenda:

- Classes novas
- Classes que sofreram alterações
- Classes sem modificação

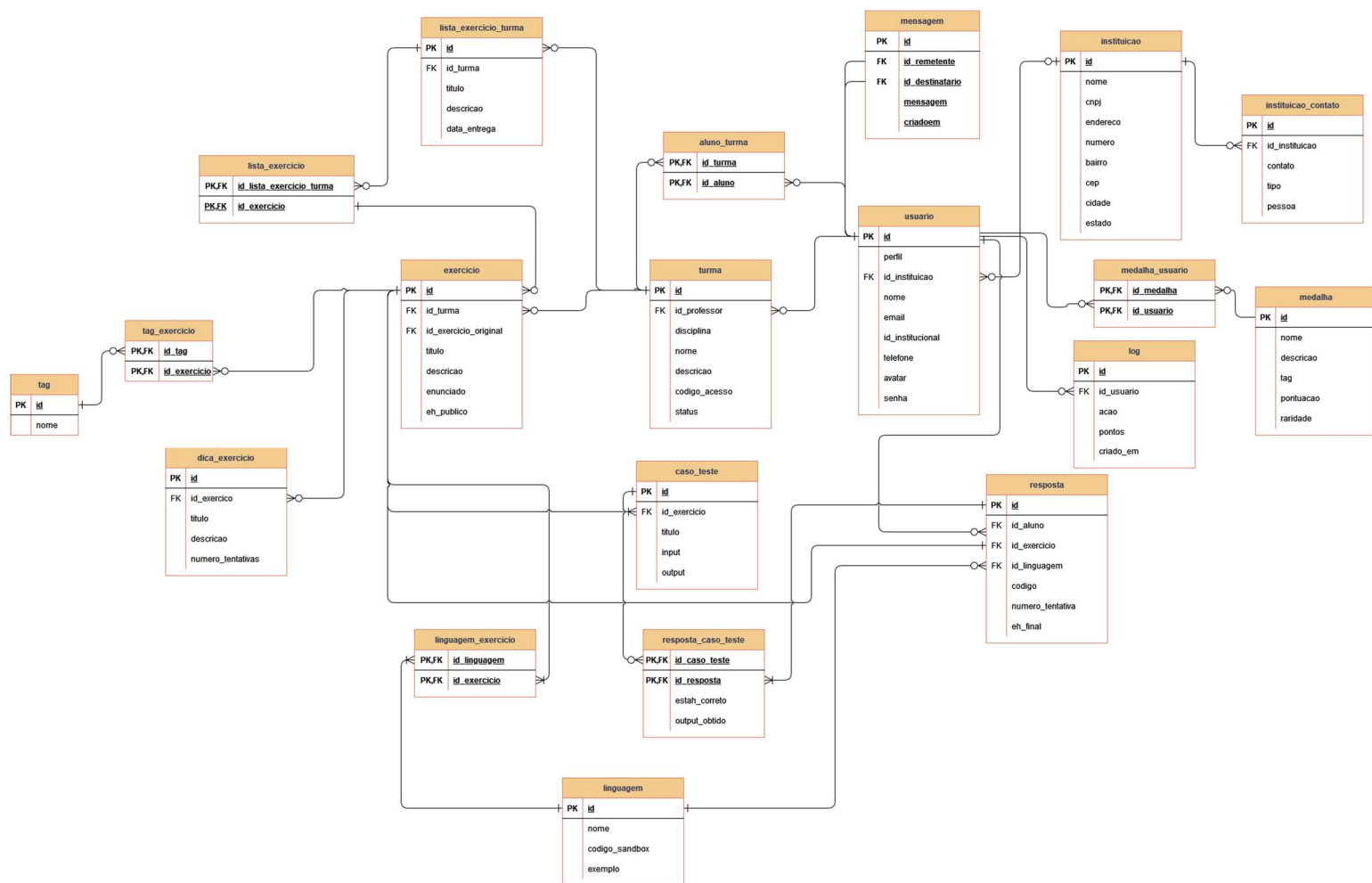
Figura 53 - Diagrama de Classes



FONTE: Os autores (2023)

APÊNDICE D: DIAGRAMA ENTIDADE RELACIONAMENTO

Figura 54 - Diagrama Entidade Relacionamento

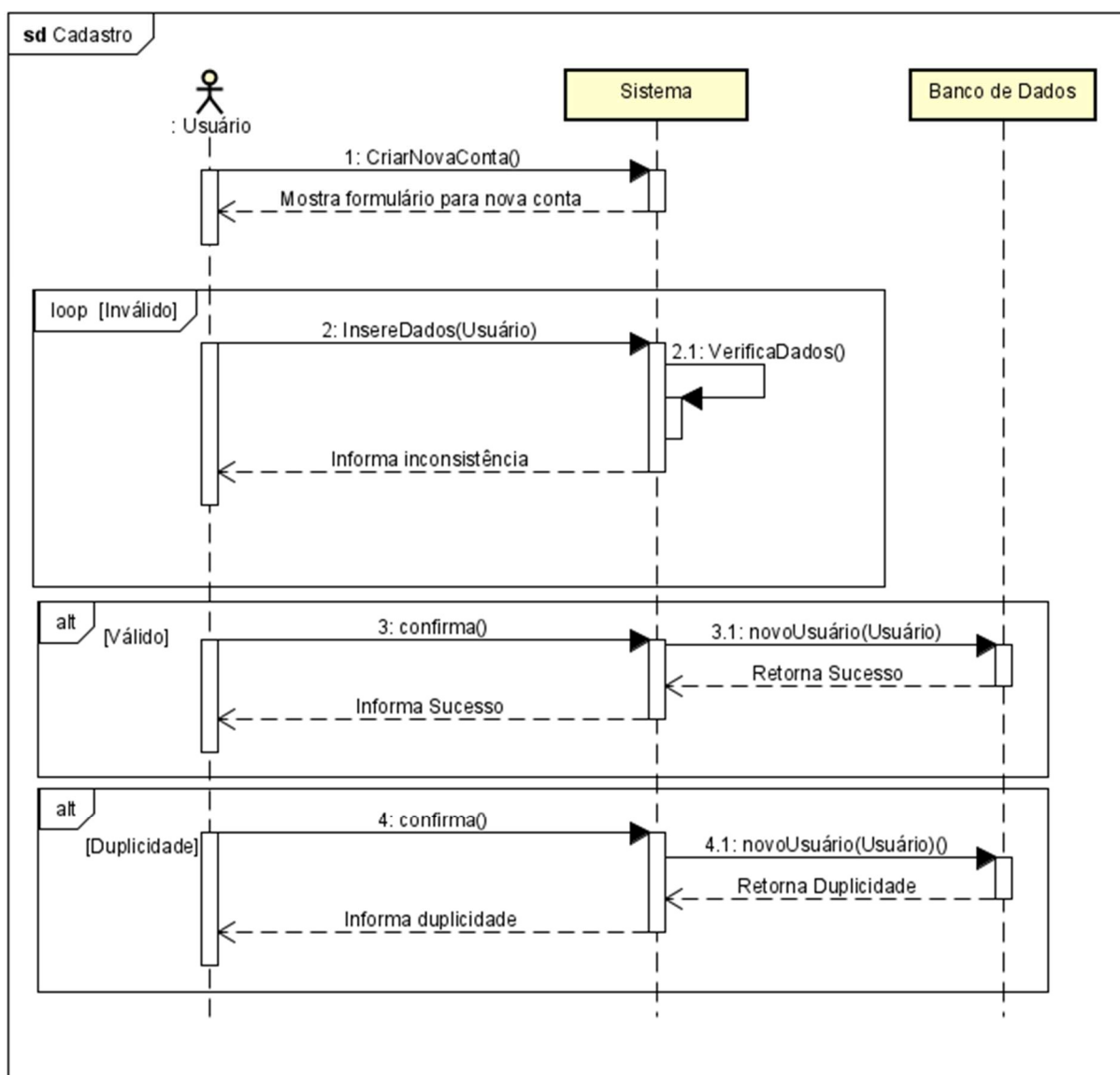


FONTE: Os autores (2023)

Este diagrama possui estética distinta dos demais devido às limitações de versão do software de modelagem Astah.

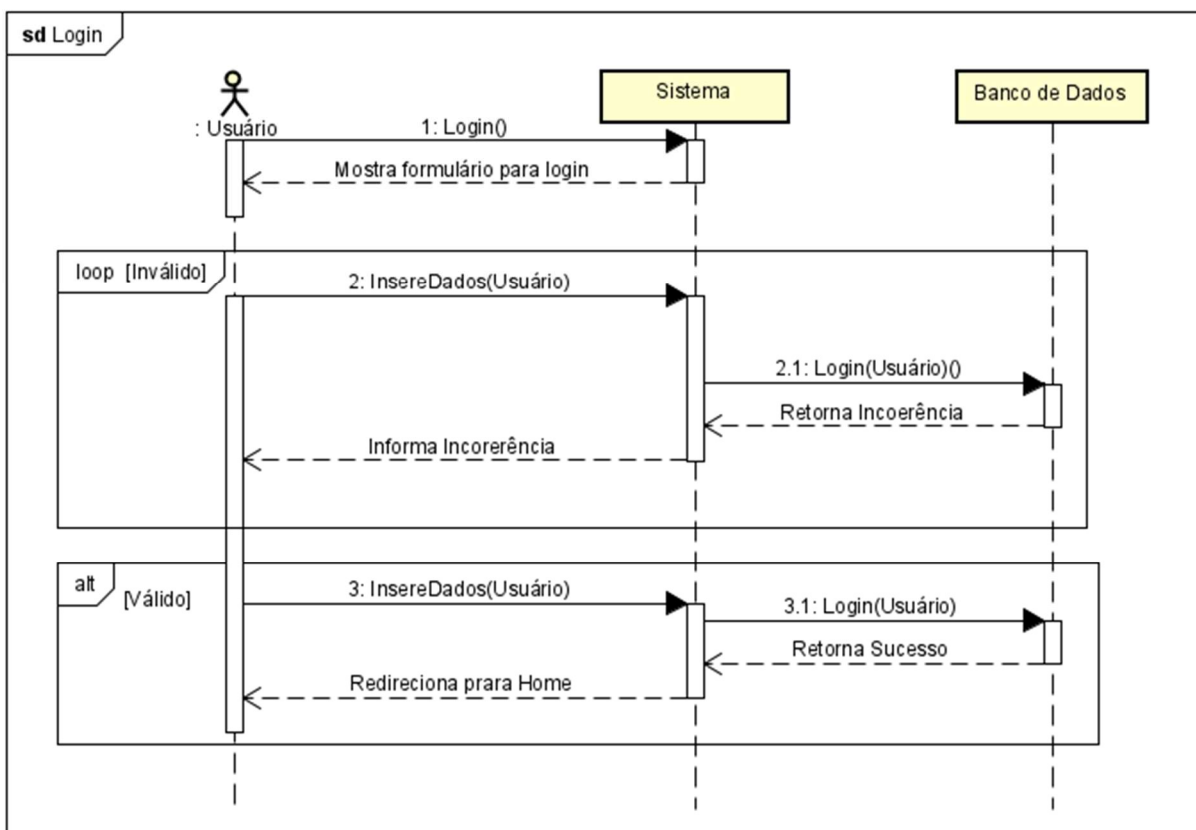
APÊNDICE E: DIAGRAMAS DE SEQUÊNCIA

Figura 55 - Diagrama de Sequência do Cadastro



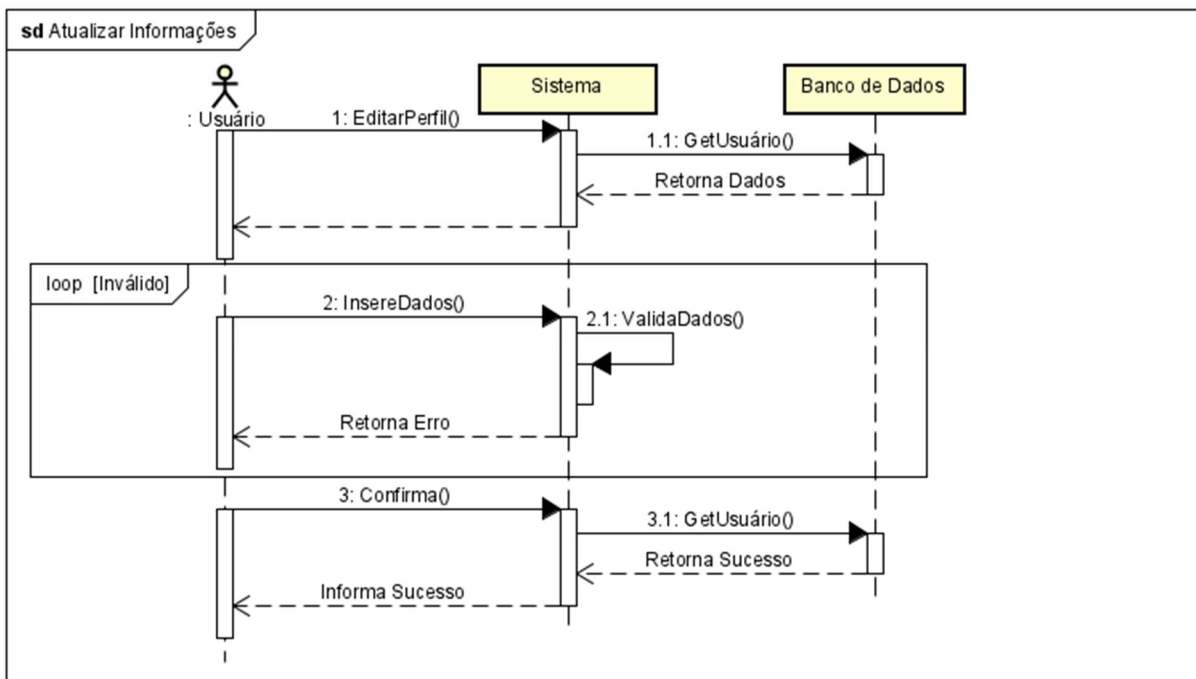
FONTE: Os autores (2023)

Figura 56 - Diagrama de Sequência de Login



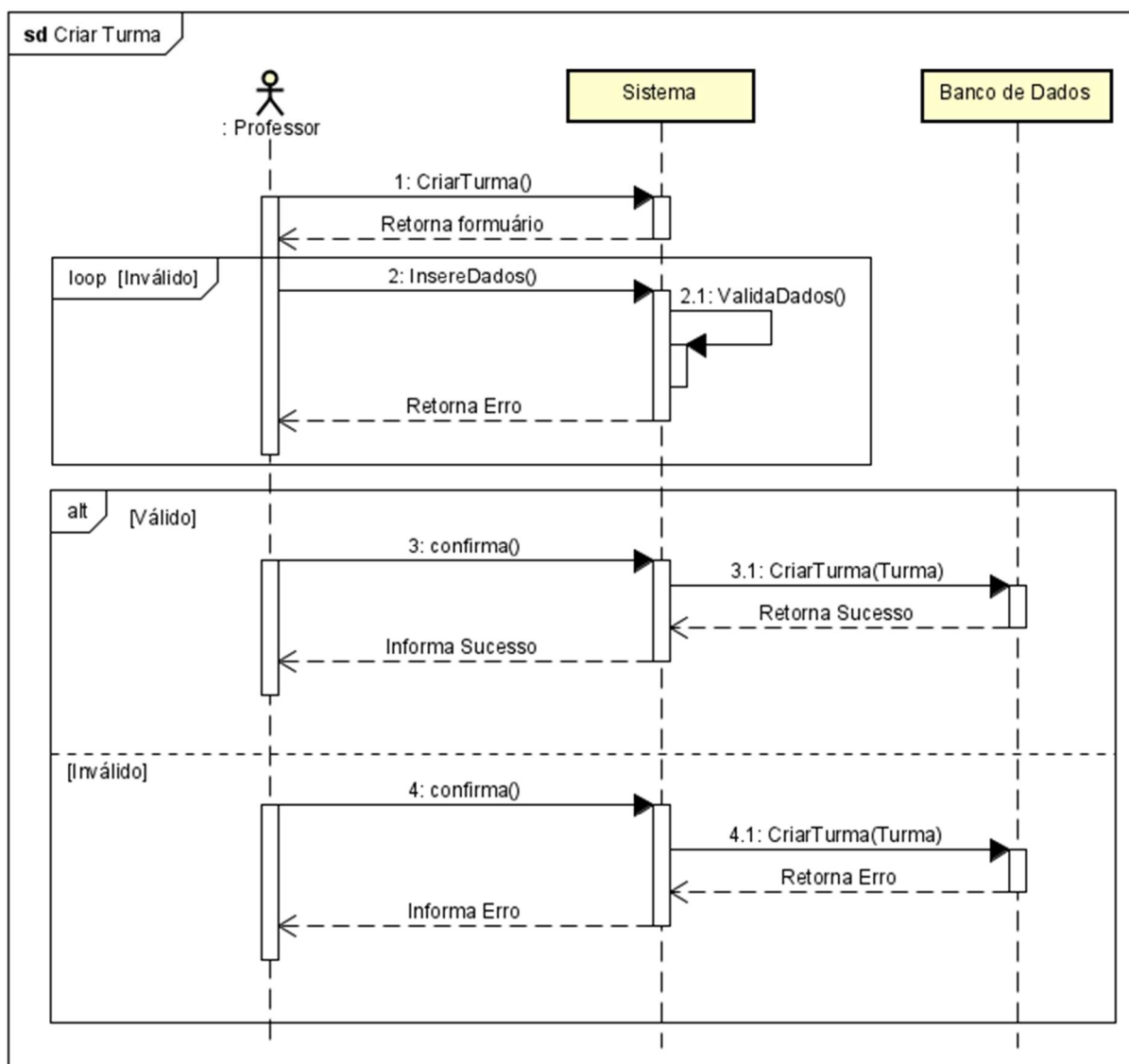
FONTE: Os autores (2023)

Figura 57 - Diagrama de Sequência de Atualizar Informações



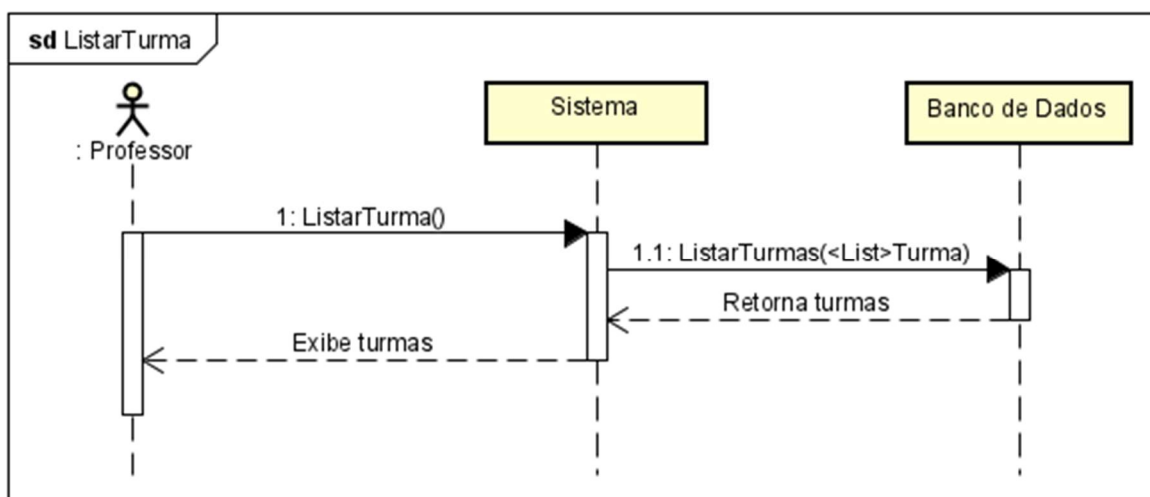
FONTE: Os autores (2023)

Figura 58 - Diagrama de Sequência de Criar Turma



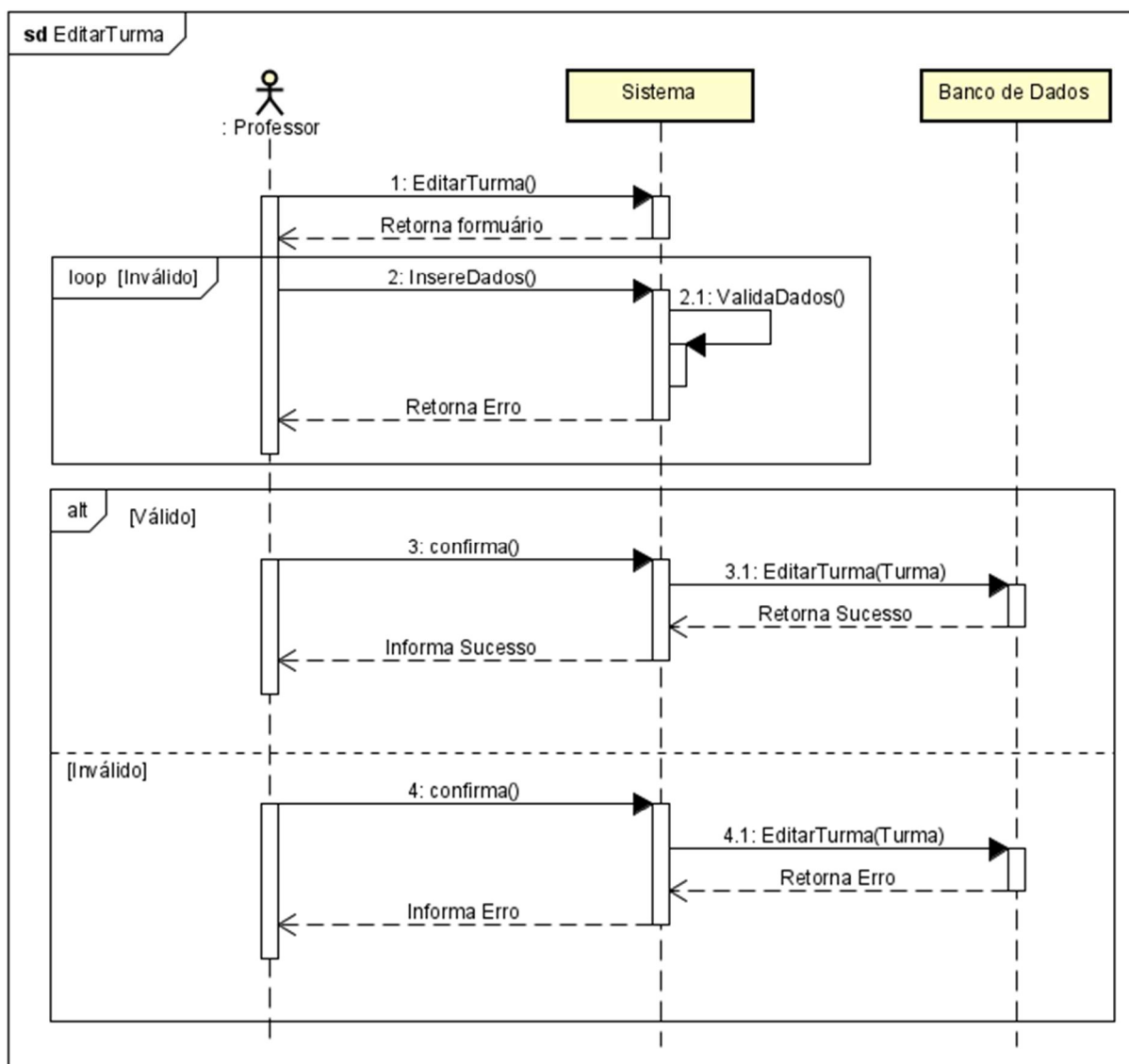
FONTE: Os autores (2023)

Figura 59 - Diagrama de Listar Turma



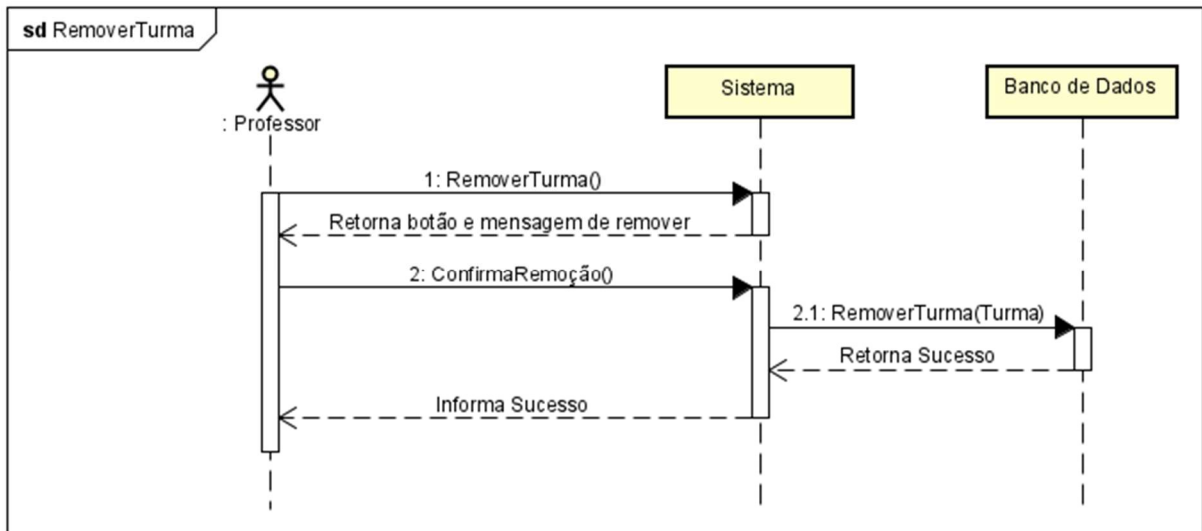
FONTE: Os autores (2023)

Figura 60 - Diagrama de Sequência de Editar Turma



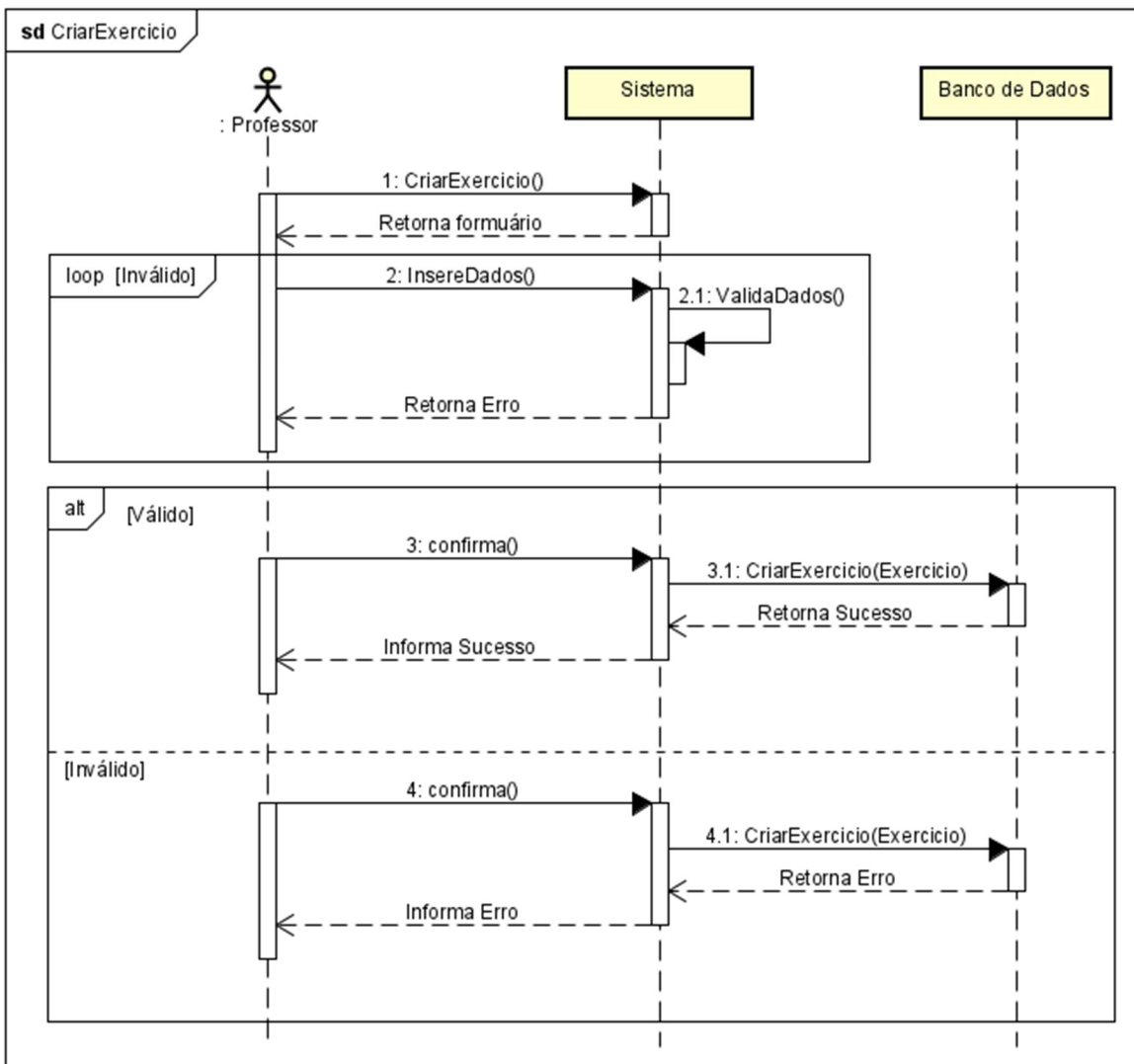
FONTE: Os autores (2023)

Figura 61 - Diagrama de Sequência de Remover Turma



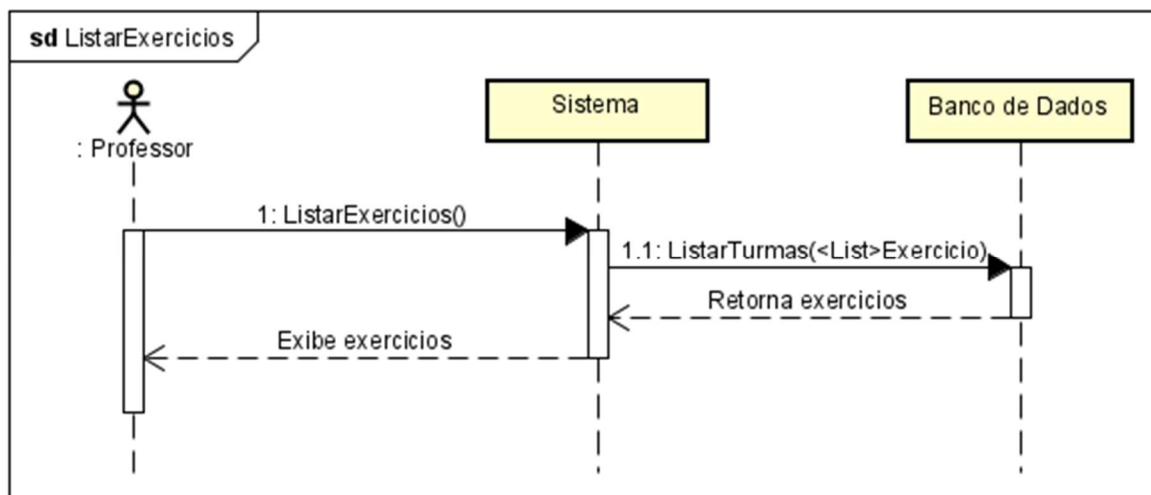
FONTE: Os autores (2023)

Figura 62 - Diagrama de Sequência de Criar Exercício



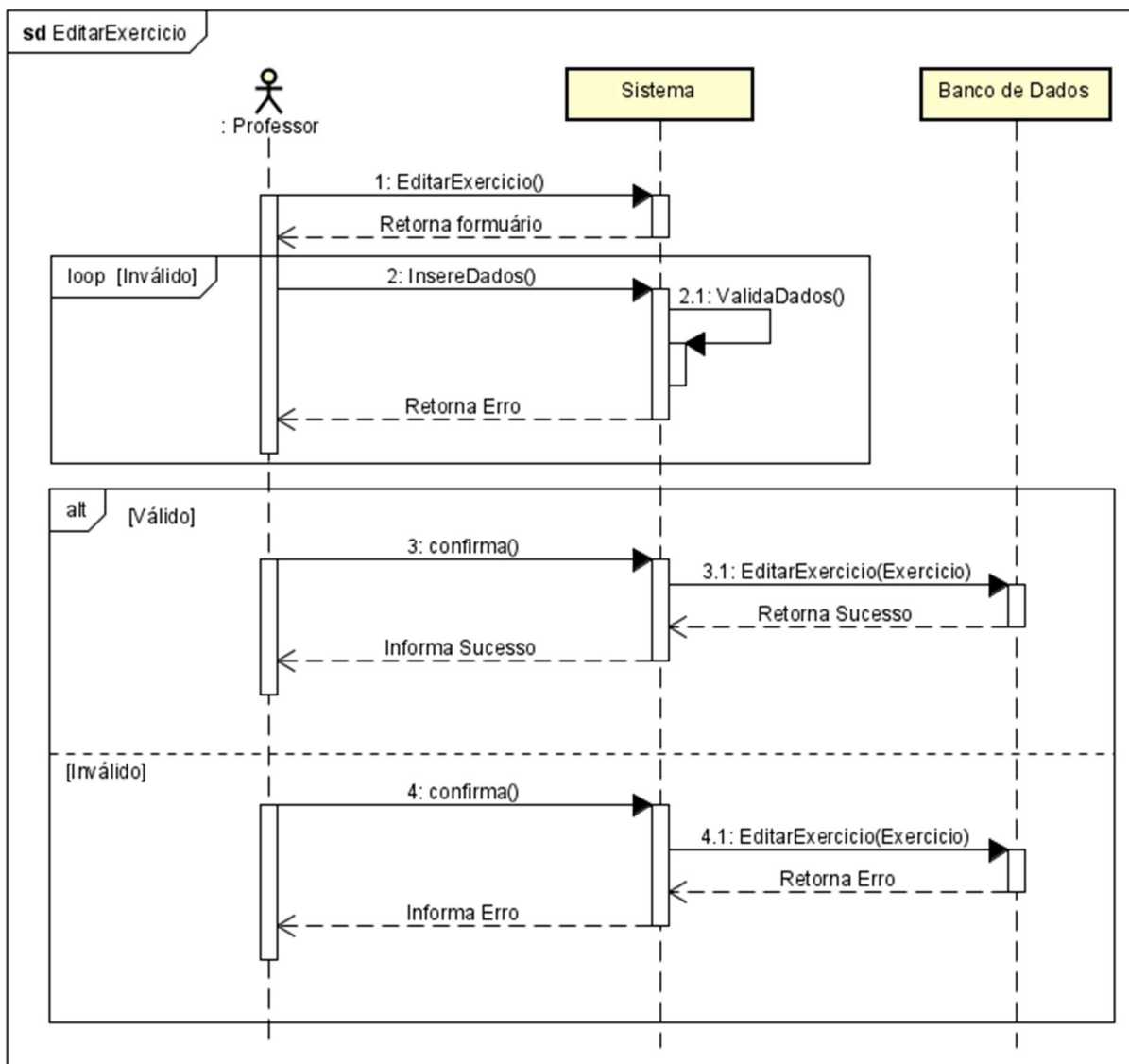
FONTE: Os autores (2023)

Figura 63 - Diagrama de Sequência de Listar Exercícios



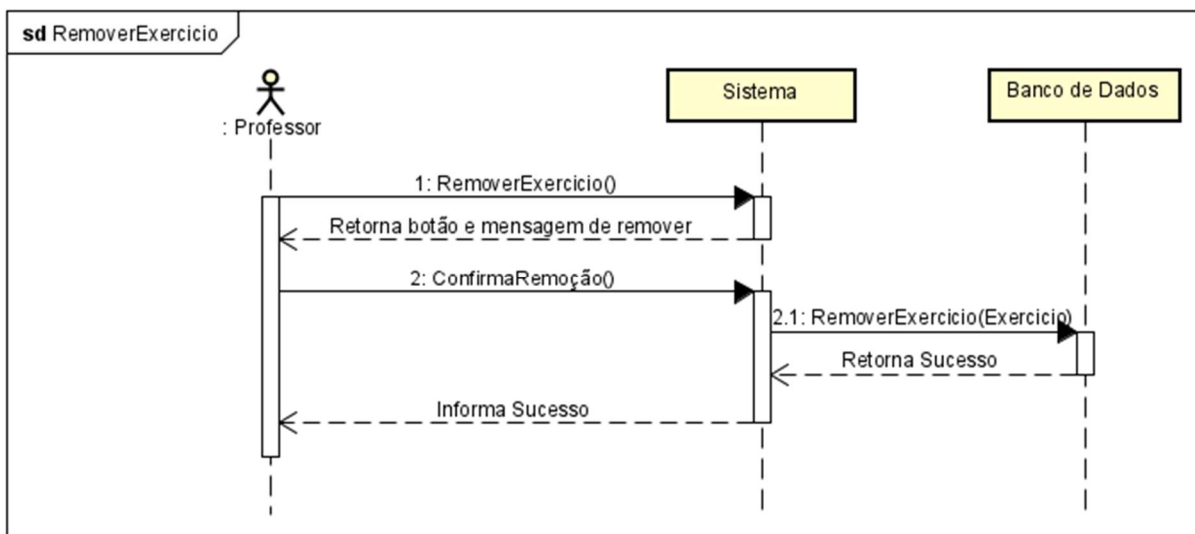
FONTE: Os autores (2023)

Figura 64 - Diagrama de Sequência de Editar Exercício



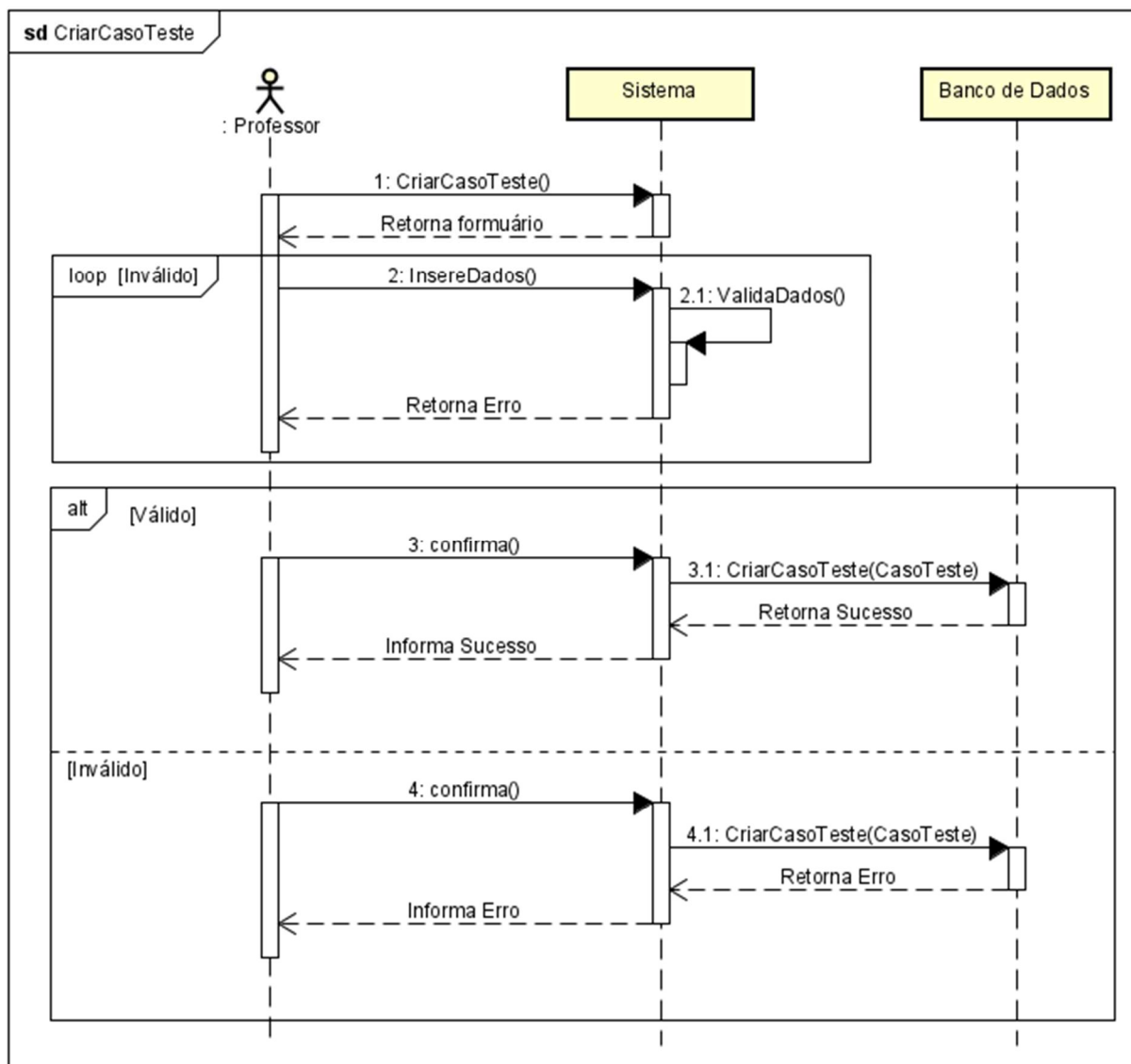
FONTE: Os autores (2023)

Figura 65 - Diagrama de Sequência de Remover Exercício



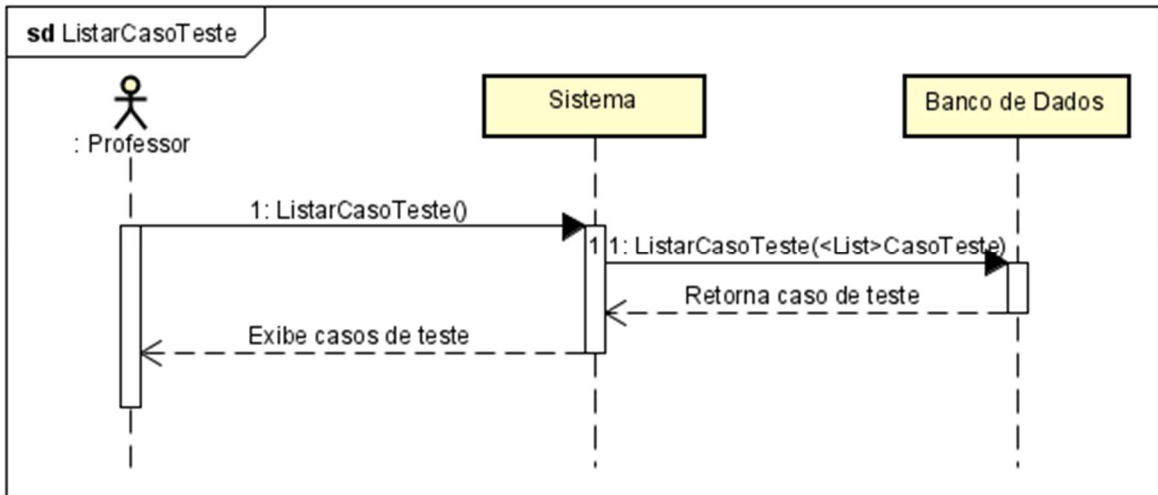
FONTE: Os autores (2023)

Figura 66 - Diagrama de Sequência de Criar Caso de Teste



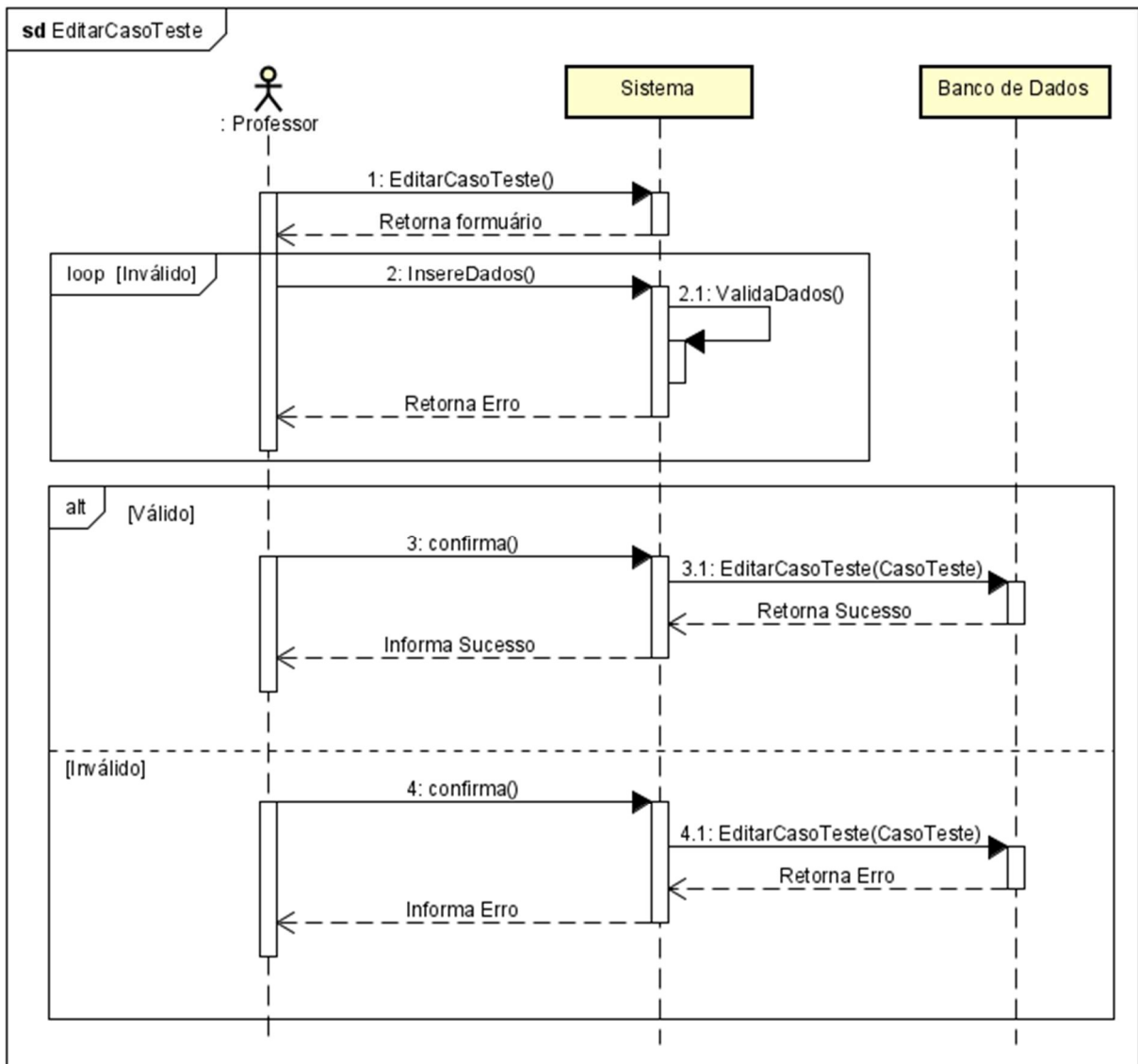
FONTE: Os autores (2023)

Figura 67 - Diagrama de Sequência de Listar Caso Teste



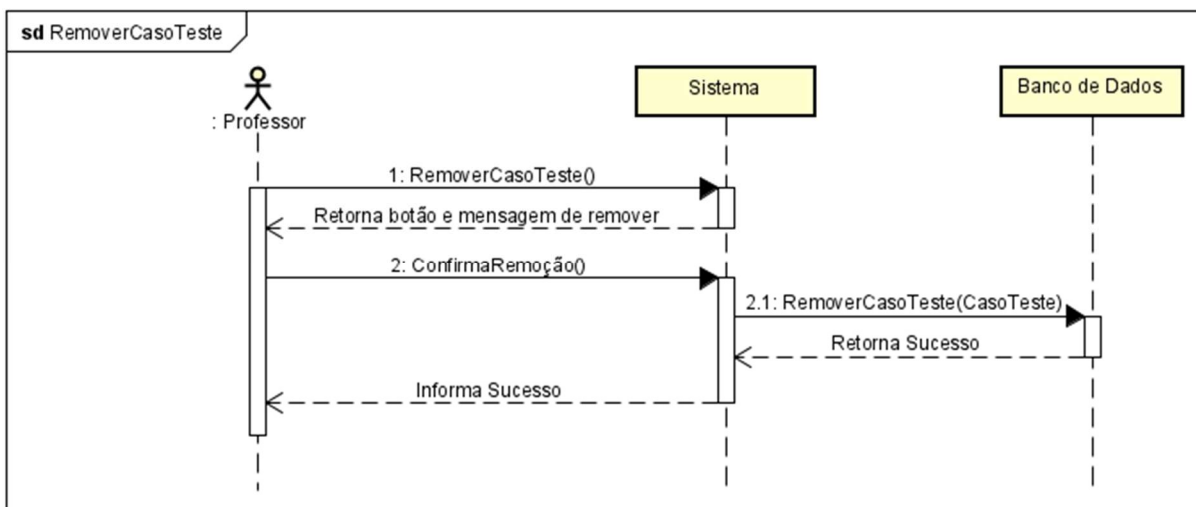
FONTE: Os autores (2023)

Figura 68 - Diagrama de Sequência de Editar Caso de Teste



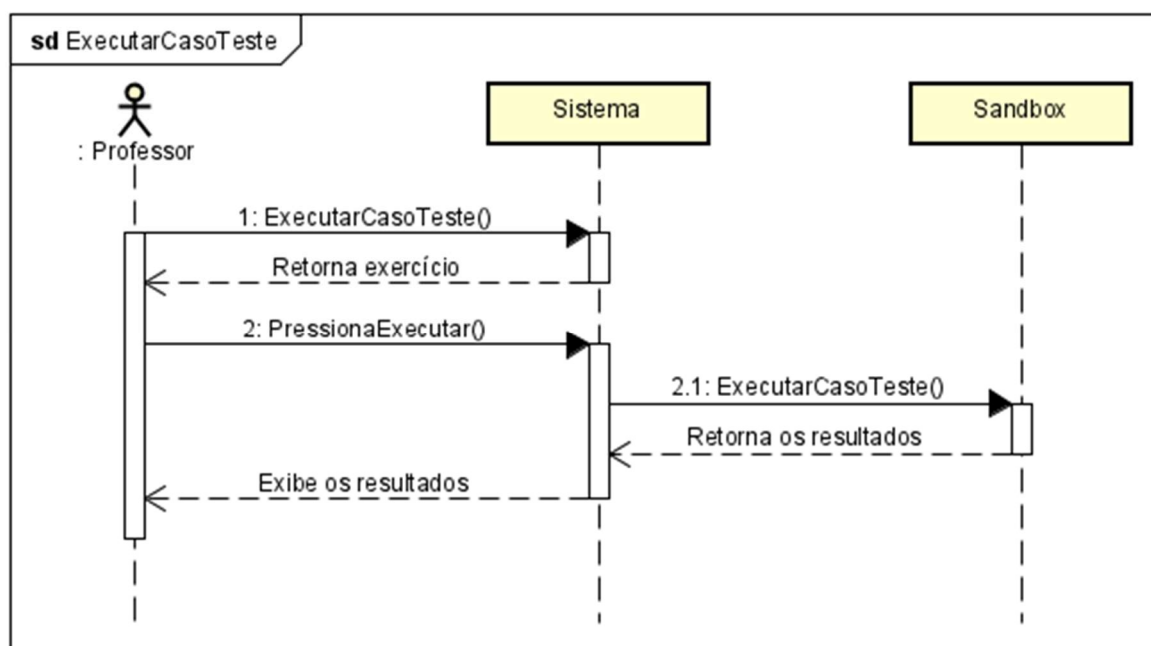
FONTE: Os autores (2023)

Figura 69 - Diagrama de Sequência de Remover Caso de Teste



FONTE: Os autores (2023)

Figura 70 - Diagrama de Sequência de Executar Caso de Teste



FONTE: Os autores (2023)

APÊNDICE F: ESPECIFICAÇÃO DE CASOS DE USO

UC01 - CADASTRAR PROFESSOR

Caso de Uso		Cadastrar Professor
Ator Principal		Usuário
Ator Secundário		-
Resumo		Permite que um usuário se cadastre no sistema como professor
Pré-condições		-
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa a página de cadastro.	
2		Carrega a tela de cadastro
3		Exibe o formulário do perfil desejado. Docente: <ul style="list-style-type: none"> ● Nome Completo; ● Email; ● Instituição; ● Matrícula; ● Telefone; ● Senha de acesso; ● Confirmar senha de acesso. Botões: <ul style="list-style-type: none"> ● Registrar.
4	Preenche o formulário e seleciona a opção "Registrar".	
5		Verifica as informações inseridas e registra as informações no banco de dados.

	Caso de Uso encerra.	
Fluxo de Exceção		
E (5.1) Email já cadastrado		
Passo	Ação do Ator	Ação do Sistema
1		Sistema informa que o e-mail já foi cadastrado
2	Caso de uso retorna para o passo 4 do fluxo principal.	
E (5.2) Senha não confere com confirmação		
Passo	Ação do Ator	Ação do Sistema
1		Sistema informa que as senhas não conferem
2	Caso de uso retorna para o passo 4 do fluxo principal.	
E (3.3) Campos obrigatórios não preenchidos		
Passo	Ação do Ator	Ação do Sistema
1		Sistema informa que os dados obrigatórios não foram preenchidos. Mensagens de erro: <ul style="list-style-type: none"> - Campo nome: "O nome precisa possuir pelo menos 5 caracteres" - Campo e-mail: "E-mail é obrigatório" - Campo matrícula: "Matrícula é obrigatória" - Campo telefone: "O campo é obrigatório" - Campo senha: "Senha é obrigatória" - Confirmação de senha: "Confirmação de senha é obrigatória"
2	Caso de uso retorna para o passo 4 do fluxo principal.	
E (3.4) Nome incompleto		
Passo	Ação do Ator	Ação do Sistema
1		Sistema informa que o nome está incompleto

2	Caso de uso retorna para o passo 4 do fluxo principal.	
E (3.5) Campo de telefone com menos de 11 dígitos		
Passo	Ação do Ator	Ação do Sistema
1		Sistema informa que o telefone precisa de, no mínimo 11 dígitos
2	Caso de uso retorna para o passo 4 do fluxo principal.	

UC02 - LOGIN

Caso de Uso		Login
Ator Principal		Usuário
Ator Secundário		-
Resumo		Permite que um usuário realize autenticação no sistema.
Pré-condições		O usuário precisa estar cadastrado.
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa a página de login.	
2		Exibe a tela com os campos do formulário. Campos: <ul style="list-style-type: none"> • E-mail; • Senha; Botões: <ul style="list-style-type: none"> • Login;
3	Usuário preenche os campos exigidos e seleciona a opção "Login"	

4		Faz a correspondência das informações preenchidas.
5		Salva as informações na sessão e redireciona o usuário a sua respectiva página inicial
6	Caso de Uso encerra.	
Fluxo de Exceção		
E (3.2) Email ou Senha incorretos		
Passo	Ação do Ator	Ação do Sistema
1		Informe que os dados inseridos estão incorretos
2	Caso de uso retorna para o passo 3 do fluxo principal.	
E (3.3) Campos obrigatórios não preenchidos		
Passo	Ação do Ator	Ação do Sistema
1		Informa que os dados obrigatórios não foram preenchidos. Mensagens de erro: - Campo e-mail: "E-mail é obrigatório" - Campo senha: "Senha é obrigatória"
2	Caso de uso retorna para o passo 3 do fluxo principal.	

UC03 - ATUALIZAR INFORMAÇÕES PROFESSOR

Caso de Uso	Atualizar Informações Professor
Ator Principal	Professor
Ator Secundário	-
Resumo	Permite que um Professor atualize suas informações pessoais
Pré-condições	Professor precisa estar autenticado

Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Professor clica no botão “Meu perfil”.	Exibe as informações de perfil do usuário contendo: <ul style="list-style-type: none"> ● Nome; ● E-mail; ● Instituição; Botões: <ul style="list-style-type: none"> ● Editar perfil (ícone de pessoa com lápis); ● Editar senha (ícone de pessoa com chave); ● Sair (ícone de seta);
2	Professor clica no botão de lápis.	Exibe um modal com os campos: <ul style="list-style-type: none"> ● Nome; ● E-mail; ● Telefone; Botões: <ul style="list-style-type: none"> ● Alterar dados;
3	Professor preenche os campos com novos dados e seleciona “Alterar dados”.	
4		Exibe mensagem de sucesso e altera os dados no banco de dados.
5	Caso de Uso encerra.	
Fluxo Alternativo		
A (2.1) Atualizar senha		
Passo	Ação do Ator	Ação do Sistema
1	Professor clica no ícone de pessoa com chave.	Exibe um modal com os campos: <ul style="list-style-type: none"> ● Senha; ● Nova senha; ● Confirmação de nova senha; Botões: <ul style="list-style-type: none"> ● Alterar senha;
2	Professor preenche os	

	campos com novos dados e seleciona "Alterar dados".	
3	Professor preenche os campos com novos dados e seleciona "Alterar dados".	
4		Exibe mensagem de sucesso e altera os dados no banco de dados.
5	Caso de Uso encerra.	
A (3.1) Cancelar a atualização dos dados		
Passo	Ação do Ator	Ação do Sistema
1	Clica no ícone de "X"	
2		Fecha a tela de atualização de informações
3	Caso de Uso encerra.	
Fluxo de Exceção		
E (3.2) Email não disponível		
Passo	Ação do Ator	Ação do Sistema
1		Informa que o e-mail escolhido não está mais disponível.
2	Caso de uso retorna para o passo 3 do fluxo principal.	
E (3.3) Campo de telefone com menos de 11 dígitos		
Passo	Ação do Ator	Ação do Sistema
1		Sistema informa que o telefone precisa de, no mínimo 11 dígitos
2	Caso de uso retorna para o passo 3 do fluxo principal.	

UC04 - TELA INICIAL PROFESSOR

Caso de Uso		Tela inicial Professor
Ator Principal		Professor
Ator Secundário		-
Resumo		Permite que um professor acesse todas as suas opções e visualize informações de forma rápida
Pré-condições		Professor precisa estar cadastrado e autenticado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Realizar login	
2		Carrega e exibe a tela inicial do professor. Botões: <ul style="list-style-type: none"> • Minhas turmas; • Meus exercícios; • Gamificação; • Meu perfil; Seções: <ul style="list-style-type: none"> • Minha turma: contendo as 3 classes mais recentes e um botão "Ver todas" • Meus exercícios: contendo os 3 exercícios mais recentes e um botão "Ver todos"
3	Caso de Uso encerra.	
Fluxo Alternativo		
A (2.1) Clicar em Minhas turmas		
Passo	Ação do Ator	Ação do Sistema
1	Usuário clica no botão de "Minhas	Direcionar o usuário para a tela de "Minhas turmas"

	turmas”.	
2	Caso de Uso encerra.	
A (2.2) Clicar em Meus exercícios		
Passo	Ação do Ator	Ação do Sistema
1	Usuário clica no botão de “Meus exercícios”.	Direcionar o usuário para a tela de “Meus exercícios”
2	Caso de Uso encerra.	
A (2.3) Clicar em Gamificação		
Passo	Ação do Ator	Ação do Sistema
1	Usuário clica no botão de “Gamificação”.	Direcionar o usuário para a tela de “Gamificação”
2	Caso de Uso encerra.	
A (2.4) Clicar em Meu perfil		
Passo	Ação do Ator	Ação do Sistema
1	Usuário clica no botão de “Meu perfil”.	Direcionar o usuário para a tela de “Meu perfil”
2	Caso de Uso encerra.	
A (2.5) Clicar em “Ver todas” - seção Minhas turmas		
Passo	Ação do Ator	Ação do Sistema
1	Usuário clica no botão de “Ver todas” na seção de “Minhas turmas”	Direcionar o usuário para a tela de “Minhas turmas”
2	Caso de Uso encerra.	

A (2.6) Clicar em “Ver todos” - seção Meus exercícios		
Passo	Ação do Ator	Ação do Sistema
1	Usuário clica no botão de “Ver todos” na seção de “Meus Exercícios”	Direcionar o usuário para a tela de “Meus exercícios”
2	Caso de Uso encerra.	
A (2.7) Clicar em um card de turma		
Passo	Ação do Ator	Ação do Sistema
1	Usuário clica em um card de turma	Direcionar o usuário para a tela da turma selecionada
2	Caso de Uso encerra.	
A (2.8) Clicar em um card de exercício		
Passo	Ação do Ator	Ação do Sistema
1	Usuário clica em um card de exercício	Direcionar o usuário para a tela do exercício selecionado.
2	Caso de Uso encerra.	

UC05 - CRIAR TURMA

Caso de Uso	Criar Turma
Ator Principal	Professor
Ator Secundário	-
Resumo	Permite que um professor crie suas turmas
Pré-condições	Professor precisa estar cadastrado e autenticado

Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa a página de turmas.	
2		Carrega e exibe a tela de turmas do professor.
3	Seleciona a opção "Criar nova turma".	
4		Abre um modal com o formulário para criação de turma. Campos: <ul style="list-style-type: none"> • Nome da disciplina; • Nome da classe; • Código de acesso da turma; • Descrição da turma; Botões: <ul style="list-style-type: none"> • Situação da turma;
5	Preenche as informações solicitadas e seleciona a opção "Registrar".	
6		Insere as informações da turma no banco de dados e exibe mensagem de sucesso.
7		Caso de Uso encerra.
Fluxo Alternativo		
A (5.1) Cancelar criação da turma		
Passo	Ação do Ator	Ação do Sistema
1	Clica no ícone de "X"	
2		Fecha a tela de criação de turma.
Fluxo de Exceção		

E (5.1) Campos obrigatórios não preenchidos		
Passo	Ação do Ator	Ação do Sistema
1		Informa que os dados obrigatórios não foram preenchidos. Mensagens de erro: <ul style="list-style-type: none"> - Campo nome da disciplina: “Disciplina deve possuir pelo menos 5 caracteres” - Campo nome da classe: “Nome deve ter pelo menos 5 caracteres” - Campo código de acesso: “Código de acesso deve ter pelo menos 5 caracteres” - Campo descrição: “Descrição deve ter pelo menos 5 caracteres” - Status turma: “O status da turma é obrigatório”
2	Caso de uso retorna para o passo 5 do fluxo principal.	

UC06 - EDITAR TURMA

Caso de Uso		Editar Turma
Ator Principal		Professor
Ator Secundário		-
Resumo		Permite que um professor edite suas turmas
Pré-condições		Professor precisa estar cadastrado e autenticado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa a página de turmas.	
2		Carrega e exibe a tela de turmas do professor.
3	Seleciona a turma desejada	Carrega e exibe a turma selecionada contendo: <ul style="list-style-type: none"> ● Nome da turma; ● Professor responsável;

		<ul style="list-style-type: none"> Exercícios da turma; <p>Botões:</p> <ul style="list-style-type: none"> Editar (ícone de lápis); Excluir (ícone de lixeira); Voltar (seta);
4	Clica no ícone de lápis	<p>Abre um modal com o formulário para edição de turma.</p> <p>Campos:</p> <ul style="list-style-type: none"> Nome da disciplina; Nome da classe; Código de acesso da turma; Descrição da turma; <p>Botões:</p> <ul style="list-style-type: none"> Situação da turma;
5	Preenche as informações solicitadas e seleciona a opção "Registrar".	
6		Atualiza as informações da turma no banco de dados e exibe mensagem de sucesso.
7	Caso de Uso encerra.	
Fluxo Alternativo		
A (5.1) Cancelar edição da turma		
Passo	Ação do Ator	Ação do Sistema
1	Clica no ícone de "X" (ou fora da modal)	
2		Fecha a tela de criação de turma.
Fluxo de Exceção		
E (5.1) Campos obrigatórios não preenchidos		
Passo	Ação do Ator	Ação do Sistema

1		Informa que os dados obrigatórios não foram preenchidos
2	Caso de uso retorna para o passo 5 do fluxo principal.	

UC07 - REMOVER TURMA

Caso de Uso		Remover Turma
Ator Principal		Professor
Ator Secundário		-
Resumo		Permite que um professor remova suas turmas
Pré-condições		Professor precisa estar cadastrado e autenticado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa a página de turmas.	
2		Carrega e exibe a tela de turmas do professor.
3	Seleciona a turma desejada	Carrega e exibe a turma selecionada contendo: <ul style="list-style-type: none"> • Nome da turma; • Professor responsável; • Exercícios da turma; Botões: <ul style="list-style-type: none"> • Editar (ícone de lápis); • Excluir (ícone de lixeira); • Voltar (seta);
4	Clica no ícone de lixeira	Abre um modal de confirmação da remoção de turma. Botões: <ul style="list-style-type: none"> • Remover turma;
4	Clica no botão "Remover"	Remove as informações da turma no banco de dados e exibe mensagem de sucesso.

5	Caso de Uso encerra.	
Fluxo Alternativo		
A (4.1) Cancelar remoção da turma		
Passo	Ação do Ator	Ação do Sistema
1	Clica no ícone de "X" (ou fora da modal)	
2		Fecha a tela de remoção de turma.

UC08 - LISTAR TURMA

Caso de Uso	Listar Turma	
Ator Principal	Professor	
Ator Secundário	-	
Resumo	Permite que um professor liste suas turmas	
Pré-condições	Professor precisa estar cadastrado e autenticado	
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa a página de turmas	
2		Carrega e exibe a tela de turmas do professor
3	Caso de Uso encerra.	

UC09 - CRIAÇÃO DE EXERCÍCIOS

Caso de Uso	Criação de Exercícios
--------------------	------------------------------

Ator Principal	Professor	
Ator Secundário	-	
Resumo	Permite que um professor crie um exercício.	
Pré-condições	Professor precisa estar cadastrado e autenticado.	
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Seleciona o Menu de “Meus exercícios”.	
2		Exibe tela de exercícios.
4	Seleciona a opção “Criar novo exercício”.	
5		Exibe a tela para criar um novo exercício contendo: Campos: <ul style="list-style-type: none"> ● Turma; ● Título; ● Descrição; ● Enunciado; Botões: <ul style="list-style-type: none"> ● Checkbox de visibilidade para outros professores; ● Configurar linguagens; ● Configurar tags; ● Configurar dicas; ● Registrar;
6	Preenche todas as informações solicitadas.	
7	Seleciona a opção “Salvar”.	
8		Grava o novo registro de exercício no banco de dados.

9		Exibe mensagem de sucesso na criação do exercício.
10	Caso de Uso encerra.	
Fluxo Alternativo		
A (6.1) Criar exercício com dica		
Passo	Ação do Ator	Ação do Sistema
1	Seleciona a opção "Configurar dicas"	
2		Sistema abre uma nova tela para criação de dica.
3	Seleciona a opção "Adicionar nova dica"	Sistema abre uma modal com as informações para criar uma dica: Campos: <ul style="list-style-type: none"> • Título; • Descrição; • Número de tentativas para desbloquear dica; Botões: <ul style="list-style-type: none"> • Registrar;
3	Preenche os campos solicitados e seleciona a opção "Registrar".	
4		Grava as informações da dica no banco de dados
5		Fecha o modal e apresenta a dica na tela de dicas do exercício.
6	Caso de Uso encerra.	
A (6.2) Configurar linguagens		
Passo	Ação do Ator	Ação do Sistema
1	Seleciona a opção "Configurar linguagens"	

2		Sistema abre um modal para selecionar as linguagens aceitas no exercício proposto: Campos: <ul style="list-style-type: none"> • Lista com seleção múltipla para indicar as linguagens aceitas (C, Java 11, Javascript, Pascal, PHP, Python 3);
3	Seleciona uma ou mais opções.	
4		Grava as informações das linguagens no banco de dados
5	Fecha o modal.	
6	Caso de Uso encerra.	
A (6.2) Configurar tags		
1	Ação do Ator	Ação do Sistema
2	Seleciona a opção "Configurar tags"	
3		Sistema abre um modal para selecionar as tags relacionadas no exercício proposto: Campos: <ul style="list-style-type: none"> • Lista com seleção múltipla para indicar as tags do exercício;
4	Seleciona uma ou mais opções.	
5		Grava as informações das tags no banco de dados
6	Fecha o modal.	
7	Caso de Uso encerra.	
Fluxo de Exceção		
E (3.1) Campos obrigatórios não preenchidos		
Passo	Ação do Ator	Ação do Sistema

1		<p>Informa que os dados obrigatórios não foram preenchidos.</p> <p>Mensagens de erro:</p> <ul style="list-style-type: none"> - Campo nome da disciplina: “Disciplina deve possuir pelo menos 5 caracteres” - Campo nome da classe: “Nome deve ter pelo menos 5 caracteres” - Campo código de acesso: “Código de acesso deve ter pelo menos 5 caracteres” - Campo descrição: “Descrição deve ter pelo menos 5 caracteres” - Status turma: “O status da turma é obrigatório”
2	Caso de uso retorna para o passo 3 do fluxo principal.	

UC10 - LISTAR EXERCÍCIOS

Caso de Uso		Listar Exercícios
Ator Principal	Professor	
Ator Secundário	-	
Resumo	Permite que um professor liste seus exercícios	
Pré-condições	Professor precisa estar cadastrado e autenticado	
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa a página de exercícios.	
2		<p>Carrega e exibe a tela de exercícios contendo o título do exercício e o começo de sua descrição.</p> <p>Botões:</p> <ul style="list-style-type: none"> ● Criar novo exercício; ● Ver exercícios públicos (ícone de cadeado)
3	Caso de Uso encerra.	
Fluxo Alternativo		

A (6.1) Ver exercícios públicos		
Passo	Ação do Ator	Ação do Sistema
1	Acessa a página de exercícios.	
2		Carrega e exibe a tela de exercícios contendo o título do exercício e o começo de sua descrição. Botões: <ul style="list-style-type: none"> • Criar novo exercício; • Ver exercícios públicos (ícone de cadeado)
3	Clica no ícone de cadeado	Carrega e exibe todos os exercícios públicos existentes.
4	Clica em um exercício público	Deve aparecer as informações do exercício de maneira read-only. Não deve aparecer o botão "Editar e executar casos de teste" e "Editar exercício"
5		Caso de Uso encerra.

UC11 - ACESSAR EXERCÍCIOS

Caso de Uso		Acessar Exercícios
Ator Principal		Professor
Ator Secundário		-
Resumo		Permite que um professor acesse seus exercícios
Pré-condições		Professor precisa estar cadastrado e autenticado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa a página de exercícios.	
2		Carrega e exibe a tela de exercícios.

3	Seleciona o exercício desejado e clica no card.	
4		Redireciona para a página com os detalhes do exercício.
5	Caso de Uso encerra.	

UC12 - EDIÇÃO DE EXERCÍCIOS

Caso de Uso		Edição de Exercícios
Ator Principal		Professor
Ator Secundário		-
Resumo		Permite que um professor edite um exercício.
Pré-condições		Professor precisa estar cadastrado e autenticado.
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Seleciona o Menu de "Exercícios".	
2		Exibe tela de exercícios.
4	Seleciona o exercício desejado	Exibe tela de detalhes do exercício.
5	Clica no ícone de lápis	Exibe a tela de edição de exercício contendo: Campos: <ul style="list-style-type: none"> ● Turma; ● Título; ● Descrição; ● Enunciado; Botões: <ul style="list-style-type: none"> ● Checkbox de visibilidade para outros professores; ● Configurar linguagens;

		<ul style="list-style-type: none"> ● Configurar tags; ● Configurar dicas; ● Registrar;
6	Preenche todas as informações solicitadas.	
7	Seleciona a opção "Registrar".	
8		Grava a atualização do registro de exercício no banco de dados.
9		Exibe mensagem de sucesso na edição do exercício.
10	Caso de Uso encerra.	
Fluxo Alternativo		
A (6.1) Criar exercício com dica		
Passo	Ação do Ator	Ação do Sistema
1	Seleciona a opção "Configurar dicas"	
2		Sistema abre uma nova tela para criação de dica.
3	Seleciona a opção "Adicionar nova dica"	<p>Sistema abre uma modal com as informações para criar uma dica:</p> <p>Campos:</p> <ul style="list-style-type: none"> ● Título; ● Descrição; ● Número de tentativas para desbloquear dica; <p>Botões:</p> <ul style="list-style-type: none"> ● Registrar;
3	Preenche os campos solicitados e seleciona a opção "Registrar".	
4		Grava as informações da dica no banco de dados
5		Fecha o modal e apresenta a dica na tela de

		dicas do exercício.
6	Caso de Uso encerra.	
A (6.2) Configurar linguagens		
Passo	Ação do Ator	Ação do Sistema
1	Seleciona a opção "Configurar linguagens"	
2		Sistema abre um modal para selecionar as linguagens aceitas no exercício proposto: Campos: <ul style="list-style-type: none"> • Lista com seleção múltipla para indicar as linguagens aceitas (C, Java 11, Javascript, Pascal, PHP, Python 3);
3	Seleciona uma ou mais opções.	
4		Grava as informações das linguagens no banco de dados
5	Fecha o modal.	
6	Caso de Uso encerra.	
A (6.2) Configurar tags		
1	Ação do Ator	Ação do Sistema
2	Seleciona a opção "Configurar tags"	
3		Sistema abre um modal para selecionar as tags relacionadas no exercício proposto: Campos: <ul style="list-style-type: none"> • Lista com seleção múltipla para indicar as tags do exercício;
4	Seleciona uma ou mais opções.	
5		Grava as informações das tags no banco de

		dados
6	Fecha o modal.	
7	Caso de Uso encerra.	
Fluxo de Exceção		
E (3.1) Campos obrigatórios não preenchidos		
Passo	Ação do Ator	Ação do Sistema
1		Informa que os dados obrigatórios não foram preenchidos
2	Caso de uso retorna para o passo 3 do fluxo principal.	

UC13 - REMOVER EXERCÍCIO

Caso de Uso		Remover Exercício
Ator Principal		Professor
Ator Secundário		-
Resumo		Permite que um professor remova seus exercícios
Pré-condições		Professor precisa estar cadastrado e autenticado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Acessa a página de exercícios.	
2		Carrega e exibe a tela de exercícios.
3	Seleciona o exercício desejado.	Exibe a tela de detalhes do exercício.
4	Clica no ícone de lixeiro.	Abre um modal de confirmação da remoção de exercício.

		Botões: <ul style="list-style-type: none"> Remover exercício;
	Clica em "Remover"	Remove as informações do exercício no banco de dados e exibe mensagem de sucesso.
6	Caso de Uso encerra.	
Fluxo Alternativo		
A (4.1) Cancelar remoção do exercício		
Passo	Ação do Ator	Ação do Sistema
1	Clica no ícone de "X" (ou fora da modal)	
2		Fecha a tela de remoção de exercício.

UC14 - CRIAÇÃO DE CASO DE TESTE

Caso de Uso		Criação de Caso de Teste
Ator Principal		Professor
Ator Secundário		-
Resumo		Permite que um professor crie um caso de teste para o exercício.
Pré-condições		Professor precisa estar cadastrado e autenticado.
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Seleciona o Menu de "Exercícios".	
2		Exibe tela de exercícios.
3	Seleciona o exercício.	

4		Exibe as informações do exercício na tela.
5	Clica no botão “Editar e executar casos de teste”	Exibe o prompt. Botões: <ul style="list-style-type: none"> • Alterar linguagem (ícone com duas setas); • Executar caso de teste (ícone de play);
6	Clica no botão “Executar caso de teste” (ícone de play)	Sistema abre uma tela com a lista dos casos de teste com os campos: Campos: <ul style="list-style-type: none"> • Nome dos casos de teste; • Entrada; • Saídas esperada; Botões: <ul style="list-style-type: none"> • Criar novo caso de teste
7	Clica no botão “Criar novo caso de teste”	Sistema abre uma modal para criar o caso de teste com os campos: Campos: <ul style="list-style-type: none"> • Nome dos casos de teste; • Entrada; • Saídas esperada; Botões: <ul style="list-style-type: none"> • Registrar
8	Preenche os campos solicitados e seleciona a opção “Registrar”.	
9		Exibe mensagem de sucesso na criação do caso de teste.
10		Caso de Uso encerra.
Fluxo Alternativo		
A (7.1) Cancelar a criação do caso de teste		
Passo	Ação do Ator	Ação do Sistema
1	Seleciona o botão “X” no modal.	

2		Sistema fecha o modal e cancela a operação
3	Caso de Uso encerra.	
Fluxo de Exceção		
E (7.1) Campos obrigatórios em branco		
Passo	Ação do Ator	Ação do Sistema
1		Destaca os campos obrigatórios não preenchidos com a cor vermelha. Mensagens de erro: <ul style="list-style-type: none"> - Campo nome: "O título do caso de teste é obrigatório" - Campo saída: "A saída do caso de teste é obrigatória"
2	Retorna para o passo 7 do fluxo principal	

UC15 - LISTAR CASOS DE TESTE

Caso de Uso		Listar Casos de Teste
Ator Principal	Professor	
Ator Secundário	-	
Resumo	Permite que um professor liste casos de teste para o exercício.	
Pré-condições	Professor precisa estar cadastrado e autenticado.	
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Seleciona o Menu de "Exercícios".	
2		Exibe tela de exercícios.

3	Seleciona o exercício.	
4		Exibe as informações do exercício na tela.
5	Clica no botão “Editar e executar casos de teste”	Exibe o prompt. Botões: <ul style="list-style-type: none"> • Alterar linguagem (ícone com duas setas); • Executar caso de teste (ícone de play);
6	Clica no botão “Executar caso de teste” (ícone de play)	Apresenta a lista de casos de teste contendo: <ul style="list-style-type: none"> - Card com cada caso de teste, se houver; - O card deve conter ícone de lápis e lixeira; Botões: <ul style="list-style-type: none"> • Criar novo caso de teste
7	Caso de Uso encerra.	

UC16 - EDITAR CASOS DE TESTE

Caso de Uso		Editar Casos de Teste
Ator Principal		Professor
Ator Secundário		-
Resumo		Permite que um professor liste casos de teste para o exercício.
Pré-condições		Professor precisa estar cadastrado e autenticado.
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Seleciona o Menu de “Exercícios”.	
2		Exibe tela de exercícios.

3	Seleciona o exercício.	
4		Exibe as informações do exercício na tela.
5	Clica no botão “Editar e executar casos de teste”	Exibe o prompt. Botões: <ul style="list-style-type: none"> • Alterar linguagem (ícone com duas setas); • Executar caso de teste (ícone de play);
6	Clica no botão “Executar caso de teste” (ícone de play)	Apresenta a lista de casos de teste contendo: <ul style="list-style-type: none"> - Card com cada caso de teste, se houver; - O card deve conter ícone de lápis e lixeira; Botões: <ul style="list-style-type: none"> • Criar novo caso de teste
7	Seleciona o ícone de lápis	
8		Sistema abre um modal para editar o caso de teste com os campos: <p>Campos:</p> <ul style="list-style-type: none"> • Nome dos casos de teste; • Entrada; • Saídas esperada; Botões: <ul style="list-style-type: none"> • Salvar
9	Preenche os campos solicitados e seleciona a opção “Salvar”.	
10		Exibe mensagem de sucesso na criação do caso de teste.
11		Caso de Uso encerra.
Fluxo Alternativo		
A (9.1) Cancelar a edição do caso de teste		

Passo	Ação do Ator	Ação do Sistema
1	Seleciona a opção "X" no modal.	
2		Sistema fecha o modal e cancela a operação
3	Caso de Uso encerra.	
Fluxo de Exceção		
E (9.1) Campos obrigatórios em branco		
Passo	Ação do Ator	Ação do Sistema
1		Destaca os campos obrigatórios não preenchidos com a cor vermelha. Mensagens de erro: <ul style="list-style-type: none"> - Campo nome: "O título do caso de teste é obrigatório" - Campo saída: "A saída do caso de teste é obrigatória"
2	Retorna para o passo 9 do fluxo principal	

UC17 - REMOVER CASOS DE TESTE

Caso de Uso	Remover Casos de Teste
Ator Principal	Professor
Ator Secundário	-
Resumo	Permite que um professor remova casos de teste para o exercício.
Pré-condições	Professor precisa estar cadastrado e autenticado.
Fluxo Principal	

Passo	Ação do Ator	Ação do Sistema
1	Seleciona o Menu de "Exercícios".	
2		Exibe tela de exercícios.
3	Seleciona o ícone de abrir o exercício.	
4		Exibe as informações do exercício na tela.
5	Clica no botão "Editar e executar casos de teste"	Exibe o prompt. Botões: <ul style="list-style-type: none"> • Alterar linguagem (ícone com duas setas); • Executar caso de teste (ícone de play);
6	Clica no botão "Executar caso de teste" (ícone de play)	Apresenta a lista de casos de teste contendo: <ul style="list-style-type: none"> - Card com cada caso de teste, se houver; - O card deve conter ícone de lápis e lixeira; Botões: <ul style="list-style-type: none"> • Criar novo caso de teste
7	Seleciona o ícone de lixeira	
8		Remove as informações do banco de dados e exibe mensagem de sucesso.
		Caso de Uso encerra.

UC18 - EXECUTAR CASOS DE TESTE

Caso de Uso	Executar Casos de Teste
Ator Principal	Professor
Ator Secundário	-
Resumo	Permite que um professor execute casos de teste para o exercício.

Pré-condições		Professor precisa estar cadastrado e autenticado.
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Seleciona o Menu de “Exercícios”.	
2		Exibe tela de exercícios.
3	Seleciona o ícone de abrir o exercício.	
4		Exibe as informações do exercício na tela.
5	Clica no botão “Editar e executar casos de teste”	Exibe o prompt. Botões: <ul style="list-style-type: none"> ● Alterar linguagem (ícone com duas setas); ● Executar caso de teste (ícone de play);
6	Clica no botão “Executar caso de teste” (ícone de play)	Apresenta a lista de casos de teste contendo: <ul style="list-style-type: none"> - Card com cada caso de teste, se houver; - O card deve conter ícone de lápis e lixeira; Botões: <ul style="list-style-type: none"> ● Criar novo caso de teste
7	Seleciona a aba “Execução”.	Deve mostrar a lista de casos de teste. Botões: <ul style="list-style-type: none"> ● Executar casos de teste
8	Clique no botão “Executar casos de teste”	Executa os casos de teste e retorna para os resultados para o usuário em cada caso de teste cadastrado, retorna também a “Saída obtida”.
9		Caso de Uso encerra.
Fluxo Alternativo		

A (6.1) Alterar a linguagem da tentativa		
Passo	Ação do Ator	Ação do Sistema
1	Clica no botão "Linguagem da tentativa" (icone de setas)	Um modal é aberto contendo: Botões: <ul style="list-style-type: none"> • Dropdown de linguagens; • Iniciar tentativa;
2	Clica no dropdown e selecione a linguagem desejada e clique em "Iniciar tentativa"	O texto do console deve mudar para a linguagem selecionada.
3	Caso de Uso encerra.	

UC19 - GAMIFICAÇÃO PROFESSOR
UC20 - CADASTRO ALUNO

Caso de Uso		Cadastrar Professor
Ator Principal	Usuário	
Ator Secundário		
Resumo	Permite que um usuário se cadastre no sistema como aluno	
Pré-condições	-	
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema

1	Acessa a página de cadastro.	
2		Carrega a tela de cadastro
3		<p>Exibe o formulário do perfil desejado.</p> <p>Discente:</p> <ul style="list-style-type: none"> ● Nome Completo; ● Email; ● Instituição; ● GRR; ● Senha de acesso; ● Confirmar senha de acesso. <p>Botões:</p> <ul style="list-style-type: none"> ● Login.
4	Preenche o formulário e seleciona a opção "Login".	
5		Verifica as informações inseridas e registra as informações no banco de dados.
6		Caso de Uso encerra.
Fluxo de Exceção		
E (5.1) Email já cadastrado		
Passo	Ação do Ator	Ação do Sistema
1		Sistema informa que o e-mail já foi cadastrado
2		Caso de uso retorna para o passo 4 do fluxo principal.
E (5.2) Senha não confere com confirmação		

Passo	Ação do Ator	Ação do Sistema
1		Sistema informa que as senhas não conferem
2	Caso de uso retorna para o passo 4 do fluxo principal.	
E (3.3) Campos obrigatórios não preenchidos		
Passo	Ação do Ator	Ação do Sistema
1		Sistema informa que os dados obrigatórios não foram preenchidos. Mensagens de erro: <ul style="list-style-type: none"> - Campo nome: "O nome precisa possuir pelo menos 5 caracteres" - Campo e-mail: "E-mail é obrigatório" - Campo GRR: "GRR precisa começar com GRR e conter 8 números" - Campo senha: "Senha é obrigatória" - Confirmação de senha: "Confirmação de senha é obrigatória"
2	Caso de uso retorna para o passo 4 do fluxo principal.	
E (3.4) Nome incompleto		
Passo	Ação do Ator	Ação do Sistema
1		Sistema informa que o nome está incompleto
2	Caso de uso retorna para o passo 4 do fluxo principal.	

UC21 - TELA INICIAL ALUNO

Caso de Uso	Tela inicial aluno
--------------------	---------------------------

Ator Principal	Aluno	
Ator Secundário	-	
Resumo	Permite que um aluno acesse todas as suas opções e visualize informações de forma rápida	
Pré-condições	Aluno precisa estar cadastrado e autenticado	
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Realizar login	
2		Carrega e exibe a tela inicial do professor. Botões: <ul style="list-style-type: none"> ● Minhas turmas; ● Meus exercícios; ● Gamificação; ● Meu perfil; Seções: <ul style="list-style-type: none"> ● Gamificação: contendo o resumo com o nível do aluno e sua pontuação. ● Minha turma: contendo as 3 classes mais recentes que os alunos registraram e um botão “Ver todas” ● Meus exercícios: contendo os 3 exercícios mais recentes cadastrados nas turmas que o aluno está registrado e um botão “Ver todos”
3	Caso de Uso encerra.	
Fluxo Alternativo		
A (2.1) Clicar em Minhas turmas		
Passo	Ação do Ator	Ação do Sistema
1	Usuário clica no botão de “Minhas turmas”.	Direcionar o usuário para a tela de “Minhas turmas”
2	Caso de Uso encerra.	

A (2.2) Clicar em Meus exercícios		
Passo	Ação do Ator	Ação do Sistema
1	Usuário clica no botão de “Meus exercícios”.	Direcionar o usuário para a tela de “Meus exercícios”
2	Caso de Uso encerra.	
A (2.3) Clicar em Gamificação		
Passo	Ação do Ator	Ação do Sistema
1	Usuário clica no botão de “Gamificação”.	Direcionar o usuário para a tela de “Gamificação”
2	Caso de Uso encerra.	
A (2.4) Clicar em Meu perfil		
Passo	Ação do Ator	Ação do Sistema
1	Usuário clica no botão de “Meu perfil”.	Direcionar o usuário para a tela de “Meu perfil”
2	Caso de Uso encerra.	
A (2.5) Clicar em “Ver todas” - seção Minhas turmas		
Passo	Ação do Ator	Ação do Sistema
1	Usuário clica no botão de “Ver todas” na seção de “Minhas turmas”	Direcionar o usuário para a tela de “Minhas turmas”
2	Caso de Uso encerra.	
A (2.6) Clicar no resumo - seção Gamificação		
Passo	Ação do Ator	Ação do Sistema

1	Usuário clica no resumo	Direcionar o usuário para a tela de Gamificação
2	Caso de Uso encerra.	
A (2.6) Clicar em “Ver todos” - seção Meus exercícios		
Passo	Ação do Ator	Ação do Sistema
1	Usuário clica no botão de “Ver todos” na seção de “Meus Exercícios”	Direcionar o usuário para a tela de “Meus exercícios”
2	Caso de Uso encerra.	
A (2.8) Clicar em um card de turma		
Passo	Ação do Ator	Ação do Sistema
1	Usuário clica em um card de turma	Direcionar o usuário para a tela da turma selecionada
2	Caso de Uso encerra.	
A (2.9) Clicar em um card de exercício		
Passo	Ação do Ator	Ação do Sistema
1	Usuário clica em um card de exercício	Direcionar o usuário para a tela do exercício selecionado.
2	Caso de Uso encerra.	

UC22 - INSCRIÇÃO NA TURMA

Caso de Uso	Inscrição Na Turma
Ator Principal	Aluno
Ator Secundário	-

Resumo		Permite que um aluno se inscreva em uma turma
Pré-condições		O aluno precisa estar cadastrado e autenticado. A turma precisa estar cadastrada. A turma e o aluno devem pertencer à mesma instituição.
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Seleciona o botão "Minhas Turmas"	
2		Exibe tela de turmas que o aluno já se inscreveu.
3	Seleciona a opção "Registrar-se em uma nova turma"	
4		Exibe o modal com o campo código de acesso e botão "Registrar"
5	Preenche com o código da turma e clica em "Registrar".	
6		Sistema cadastra o aluno na turma e exibe mensagem de sucesso.
7		Caso de Uso encerra.
Fluxo de Exceção		
E (5.1) Campos obrigatórios em branco		
Passo	Ação do Ator	Ação do Sistema
1		Destaca os campos obrigatórios não preenchidos com a cor vermelha. Mensagem de erro: "O código da turma é obrigatório".

2	Retorna para o passo 5 do fluxo principal	
E (6.1) Código de turma não encontrado		
Passo	Ação do Ator	Ação do Sistema
1		Envia mensagem de erro para o usuário. Mensagem de erro: "Verifique o código da turma informado e tente novamente".
2	Retorna para o passo 2 do fluxo principal	

UC23 - VER PERFIL

Caso de Uso		Visitar Perfil de Aluno
Ator Principal	Usuário	
Ator Secundário	-	
Resumo	Permite que um usuário visualize informações do seu perfil	
Pré-condições	Usuário precisa estar cadastrado e autenticado	
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Clica sobre o botão "Meu Perfil"	
2		Carrega informações do usuário logado e apresenta informações na tela. Informações: <ul style="list-style-type: none"> • Nome; • Email; • Instituição; Botões: <ul style="list-style-type: none"> • Editar perfil (ícone de pessoa com lápis); • Editar senha (ícone de pessoa com chave);

		<ul style="list-style-type: none"> ● Sair (ícone de seta);
3	Caso de Uso encerra.	

UC24 - ATUALIZAR INFORMAÇÕES ALUNO

Caso de Uso		Atualizar Informações Aluno
Ator Principal	Aluno	
Ator Secundário	-	
Resumo	Permite que um aluno atualize suas informações pessoais	
Pré-condições	Aluno precisa estar autenticado	
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Aluno clica no botão "Meu perfil".	Exibe as informações de perfil do usuário contendo: <ul style="list-style-type: none"> ● Nome; ● E-mail; ● Instituição; Botões: <ul style="list-style-type: none"> ● Editar perfil (ícone de pessoa com lápis); ● Editar senha (ícone de pessoa com chave); ● Sair (ícone de seta);
2	Aluno clica no ícone de pessoa com lápis.	Exibe um modal com os campos: <ul style="list-style-type: none"> ● Nome; ● E-mail; Botões: <ul style="list-style-type: none"> ● Alterar dados;
3	Aluno preenche os campos com novos dados e seleciona "Alterar dados".	
4		Exibe mensagem de sucesso e altera os dados no banco de dados.

5	Caso de Uso encerra.	
Fluxo Alternativo		
A (2.1) Atualizar senha		
Passo	Ação do Ator	Ação do Sistema
1	Aluno clica no ícone de pessoa com chave.	Exibe um modal com os campos: <ul style="list-style-type: none"> • Senha; • Nova senha; • Confirmação de nova senha; Botões: <ul style="list-style-type: none"> • Alterar senha;
2	Aluno preenche os campos com novos dados e seleciona "Alterar dados".	
3	Aluno preenche os campos com novos dados e seleciona "Alterar dados".	
4		Exibe mensagem de sucesso e altera os dados no banco de dados.
5	Caso de Uso encerra.	
A (3.1) Cancelar a atualização dos dados		
Passo	Ação do Ator	Ação do Sistema
1	Clica no ícone de "X"	
2		Fecha a tela de atualização de informações
3	Caso de Uso encerra.	
Fluxo de Exceção		
E (3.2) Email não disponível		
Passo	Ação do Ator	Ação do Sistema

1		Informa que o e-mail escolhido não está mais disponível.
2	Caso de uso retorna para o passo 3 do fluxo principal.	

UC25 - VISITAR RANKING

Caso de Uso		Visitar Ranking
Ator Principal		Aluno
Ator Secundário		-
Resumo		Permite que um aluno visualize o ranking de pontuações
Pré-condições		Aluno precisa estar cadastrado e autenticado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Seleciona o botão de "Gamificação".	
2		Exibe a tela do ranking de alunos. Informações: <ul style="list-style-type: none"> ● Nível do aluno; ● Posição geral no ranking; ● Saldo de pontos; ● Quantidade de pontos faltantes para o próximo nível; ● Medalhas; ● Visualização dos 10 primeiros do ranking e a posição do usuário.
3	Caso de Uso encerra.	
Fluxo Alternativo		
A (2.1) Clicar em uma medalha		
Passo	Ação do Ator	Ação do Sistema

1	Aluno clica na medalha	Exibe um modal com os dados: <ul style="list-style-type: none"> • Nome da medalha; • Descrição da medalha; • Raridade da badge (Consultar Tabela 1 – Conquistas, pontuação e raridade);
2	Caso de Uso encerra.	

UC26 - CRIAR RESPOSTA PARA EXERCÍCIO

Caso de Uso		Inscrição Na Turma
Ator Principal		Aluno
Ator Secundário		-
Resumo		Permite que um aluno se inscreva em uma turma
Pré-condições		O aluno precisa estar cadastrado e autenticado. A turma precisa estar cadastrada. A turma e o aluno devem pertencer à mesma instituição. A turma deve ter pelo menos um exercício.
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Seleciona o botão "Minhas Turmas".	
2		Exibe tela de turmas que o aluno já se inscreveu.
3	Seleciona a turma desejada.	
4		Exibe os exercícios da turma.
5	Seleciona o exercício desejado.	

6		Carrega a tela com detalhes e informações do exercício.
7	Clica em “Iniciar tentativa”	Abre uma modal para seleção de linguagem da tentativa com um dropdown que contém as linguagens disponíveis.
8	Seleciona a linguagem da tentativa e clica em “Iniciar tentativa”	Abre o console para a tentativa ser codificada.
9	Seleciona a opção de “Execução” (ícone de play).	Abre a página de execução que contém duas abas: <ul style="list-style-type: none"> • Casos de teste; • Execução;
10	Seleciona a aba “Execução”	Abre a página de execução onde o aluno pode “Executar” e “Executar e salvar”
11	Clica no botão “Executar e salvar”	Registra a resposta e salva no banco de dados. E direciona o usuário para a página inicial do exercício.
12		A página inicial do exercício não deve mais conter o botão “Iniciar tentativa”
13		Caso de Uso encerra.
Fluxo Alternativo		
A (9.1) Clicar em Executar		
Passo	Ação do Ator	Ação do Sistema
1	Aluno clica no botão “Executar”	O caso de teste é executado e retorna à validação dele. O botão executar pode ser clicado na quantidade de tentativas que o exercício foi cadastrado para possuir.
2		Caso de Uso encerra.

UC27 - LISTAR EXERCÍCIOS DA TURMA

Caso de Uso		Inscrição Na Turma
Ator Principal		Aluno
Ator Secundário		-
Resumo		Permite que um aluno visualize os exercícios da turma
Pré-condições		O aluno precisa estar cadastrado e autenticado. A turma precisa estar cadastrada. A turma e o aluno devem pertencer à mesma instituição. A turma deve ter pelo menos um exercício.
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Seleciona o botão "Minhas Turmas".	
2		Exibe tela de turmas que o aluno já se inscreveu.
3	Seleciona a turma desejada.	
4		Exibe os exercícios da turma.
11	Caso de Uso encerra.	

UC28 - ACESSAR TURMA

Caso de Uso		Acessar Turma
Ator Principal		Aluno
Ator Secundário		-

Resumo	Permite que um aluno visualize os detalhes da turma	
Pré-condições	<p>O aluno precisa estar cadastrado e autenticado.</p> <p>A turma precisa estar cadastrada.</p> <p>A turma e o aluno devem pertencer à mesma instituição.</p> <p>A turma deve ter pelo menos um exercício.</p>	
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Seleciona o botão "Minhas Turmas".	
2		Exibe tela de turmas que o aluno já se inscreveu.
3	Seleciona a turma desejada.	
4		<p>Exibe:</p> <ul style="list-style-type: none"> ● Nome da turma; ● Nome do professor; ● Exercícios cadastrados <p>Botões:</p> <ul style="list-style-type: none"> ● Info;
5	Caso de Uso encerra.	
Fluxo Alternativo		
A (4.1) Clicar no botão de info e em enviar e-mail		
Passo	Ação do Ator	Ação do Sistema
1	Aluno clica no botão "Info"	<p>Abre um modal com:</p> <ul style="list-style-type: none"> ● Nome do professor <p>Botões:</p> <ul style="list-style-type: none"> ● Enviar e-mail;
2	Aluno clica em "Enviar e-mail"	O usuário é direcionado ao e-mail com o destinatário sendo o professor da disciplina.



3	Caso de Uso encerra.
---	----------------------




UC29 - VER RESPOSTAS

Caso de Uso		Ver respostas
Ator Principal		Professor
Ator Secundário		-
Resumo		Permite que um Professor veja a lista de alunos com suas respectivas respostas do exercício proposto
Pré-condições		O Professor deve estar cadastrado e autenticado
Fluxo Principal		
Passo	Ação do Ator	Ação do Sistema
1	Seleciona o botão "Meus Exercícios".	
2		Exibe lista de exercícios.
3	Seleciona o um card para abrir um exercício.	Exibe a tela do exercício selecionado contendo: <ul style="list-style-type: none"> ● Título do exercício; ● Descrição do exercício; ● Enunciado do exercício; ● Casos de teste (se houver); Botões: <ul style="list-style-type: none"> ● Editar exercício (ícone de lápis); ● Excluir exercício (ícone de lixeira); ● Editar e executar casos de teste; ● Ver respostas;
4	Clica no botão "Ver respostas"	Exibe uma lista com todos os alunos contendo as Informações: <ul style="list-style-type: none"> ● Card com nome do aluno e a situação do exercício para o aluno ("RESPONDIDO", "NÃO RESPONDIDO" E "EM TENTATIVA") Botões: <ul style="list-style-type: none"> ● Filtro por status "(ícone de pessoa com check por default, ao clicar para outra forma de visualização ícone de várias pessoas)";

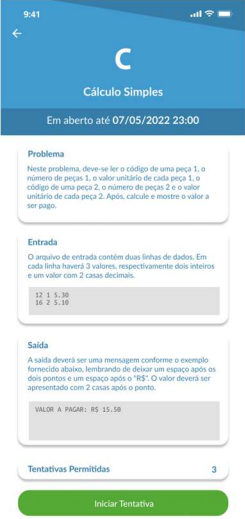


5	Clica no card do aluno com status "RESPONDIDO" ou "EM TENTATIVA"	Exibe uma tela com: <ul style="list-style-type: none"> • Card com nome do aluno e a situação do exercício para o aluno ("RESPONDIDO", "NÃO RESPONDIDO" E "EM TENTATIVA") • Tentativa com: input, output e saída obtida. Botões: <ul style="list-style-type: none"> • Ver código (em cada tentativa);
6	Clica em "Ver código"	Exibe o prompt com o código da tentativa selecionada como read-only.
7	Caso de Uso encerra.	
Fluxo Alternativo		
A (5.1) Clica no card do aluno com status "NÃO RESPONDIDO"		
Passo	Ação do Ator	Ação do Sistema
1		Sistema informa que o aluno não respondeu e nem fez nenhuma tentativa para o exercício.
2	Caso de uso retorna para o passo 5 do fluxo principal.	

APÊNDICE G - LISTA DE PROTÓTIPOS DE TELA DESENVOLVIDOS PARA A VERSÃO MOBILE

Protótipo da tela	Nome	Funcionalidade base
	Splash Screen	Tela de splash do aplicativo, duração estimada de 1,5s na maioria dos dispositivos
	Seleção de Login	Tela de seleção de cargo. Caso a seleção "sou docente" seja selecionada, o tema dourado é aplicado em todo o App.

 <p>9:41 ← farmaAlg Login Registro</p>	<p>Seleção de autenticação ou cadastro</p>	<p>Seleção para login ou registro em caso de novo acesso.</p>
 <p>9:41 ← farmaAlg Login E-mail Senha Entrar Esqueceu sua senha?</p>	<p>Login</p>	<p>Tela de login com e-mail institucional UFPR e senha. Um botão de esquecimento de senha é destacado na parte inferior.</p>
 <p>9:41 ← farmaAlg Cadastro Nome completo Email de contato Instituição de ensino GRR Senha de acesso Confirmar senha de acesso Cadastrar</p>	<p>Cadastro</p>	<p>Tela de cadastro de alunos, os campos de entrada de dados são semelhantes aos presentes na versão web.</p>

	<h2>Home</h2>	<p>Tela principal do aplicativo. Destaca atalhos para todas as páginas no topo, seção de gamificação com pontuação e nível atual, listagem de turmas matriculadas pelo aluno.</p>
	<h2>Turmas</h2>	<p>Listagem de todas as turmas disponíveis para esse usuário. Na versão para docentes, se destacam as turmas de disciplinas ministradas pelo professor.</p>
	<h2>Turma Selecionada</h2>	<p>Tela de turma individual, contendo entregas e listagem de exercícios pendentes. No topo, se destaca o nome da disciplina e o docente da turma.</p>

	<h3>Exercício Selecionado</h3>	<p>Tela do exercício selecionado, contendo data de entrega, problema, entrada a ser executada e saída prevista.</p>
	<h3>Resposta do exercício</h3>	<p>Tela para resposta do aplicativo, contendo saída esperada, área para inserção do código com formatação personalizada e botões “Executar” para testar o código na sandbox e “Executar e enviar” para execução e envio do exercício selecionado para o professor.</p>
	<h3>Gamificação</h3>	<p>Tela inicial da gamificação, contendo nível atual, posição no ranking geral, saldo de pontos da conta, posição no ranking de cada linguagem com exercícios respondidos pelo aluno, pontos necessários para o próximo nível, extrato de pontos e primeiras posições do ranking mensal.</p>

 <p>9:41 ← Extrato de pontos</p> <p>Responder um exercício 26/04/2022 23:46 +1100 pts</p> <p>Finalizou seu primeiro exercício em C 26/04/2022 13:46 +200 pts</p> <p>Fez login no FarmaAlg Mobile 26/04/2022 05:46 +150 pts</p> <p>Responder um exercício 25/04/2022 20:46 +1100 pts</p> <p>Finalizou seu primeiro exercício em C 24/04/2022 18:46 +200 pts</p> <p>Fez login no FarmaAlg Mobile 23/04/2022 19:46 +150 pts</p>	<h2>Extrato de pontos</h2>	<p>Extrato de pontos do usuário por período selecionável via filtro</p>																																																
 <p>9:41 ← Ranking</p> <p>Java - Abril/2022</p> <table><tbody><tr><td>1*</td><td>Marcos Fernandes</td><td>1500 pts</td></tr><tr><td>2*</td><td>Thiago Silva</td><td>1500 pts</td></tr><tr><td>3*</td><td>Maria Souza</td><td>1400 pts</td></tr><tr><td>4*</td><td>Mariana Lagos</td><td>1400 pts</td></tr><tr><td>5*</td><td>Karolina Paz</td><td>1350 pts</td></tr><tr><td>6*</td><td>Richard Vieira</td><td>1300 pts</td></tr><tr><td>7*</td><td>Nair Souza</td><td>1200 pts</td></tr><tr><td>8*</td><td>Thayla Ferro</td><td>1100 pts</td></tr><tr><td>9*</td><td>Carmo Dourado</td><td>1000 pts</td></tr><tr><td>10*</td><td>Caietana Freixo</td><td>950 pts</td></tr><tr><td>11*</td><td>Danielle Leal</td><td>900 pts</td></tr><tr><td>12*</td><td>Isacc Silva</td><td>850 pts</td></tr><tr><td>13*</td><td>Geovana Carminha</td><td>850 pts</td></tr><tr><td>14*</td><td>Charlotte Marcos</td><td>800 pts</td></tr><tr><td>15*</td><td>Rayane Quintas</td><td>750 pts</td></tr><tr><td>16*</td><td>Carmo Cerqueira</td><td>700 pts</td></tr></tbody></table>	1*	Marcos Fernandes	1500 pts	2*	Thiago Silva	1500 pts	3*	Maria Souza	1400 pts	4*	Mariana Lagos	1400 pts	5*	Karolina Paz	1350 pts	6*	Richard Vieira	1300 pts	7*	Nair Souza	1200 pts	8*	Thayla Ferro	1100 pts	9*	Carmo Dourado	1000 pts	10*	Caietana Freixo	950 pts	11*	Danielle Leal	900 pts	12*	Isacc Silva	850 pts	13*	Geovana Carminha	850 pts	14*	Charlotte Marcos	800 pts	15*	Rayane Quintas	750 pts	16*	Carmo Cerqueira	700 pts	<h2>Ranking</h2>	<p>Ranking de usuários com mais pontos conforme filtros definidos por linguagem, tempo ou geral.</p>
1*	Marcos Fernandes	1500 pts																																																
2*	Thiago Silva	1500 pts																																																
3*	Maria Souza	1400 pts																																																
4*	Mariana Lagos	1400 pts																																																
5*	Karolina Paz	1350 pts																																																
6*	Richard Vieira	1300 pts																																																
7*	Nair Souza	1200 pts																																																
8*	Thayla Ferro	1100 pts																																																
9*	Carmo Dourado	1000 pts																																																
10*	Caietana Freixo	950 pts																																																
11*	Danielle Leal	900 pts																																																
12*	Isacc Silva	850 pts																																																
13*	Geovana Carminha	850 pts																																																
14*	Charlotte Marcos	800 pts																																																
15*	Rayane Quintas	750 pts																																																
16*	Carmo Cerqueira	700 pts																																																

APÊNDICE H - SPRINTS DE DESENVOLVIMENTO DO PROJETO

Atividades	Data inicial	Data final
Sprint 1	06/06/2022	20/06/2022
<ul style="list-style-type: none"> - Configuração de Ambiente de desenvolvimento - Estudo do código da implementação anterior 		
Sprint 2	20/06/2022	04/07/2022
<ul style="list-style-type: none"> - Estudo do código da implementação anterior - Hospedagem dos projetos Front-end e Back-end no Heroku - Definição de bibliotecas de estilo a serem utilizadas nos projetos mobile 		
Sprint 3	04/07/2022	18/07/2022
<ul style="list-style-type: none"> - Hospedagem do serviço Sandbox no Heroku - Início do desenvolvimento do APP Aluno - UC02 (LOGIN): Desenvolvimento 		
Sprint 4	18/07/2022	01/08/2022
<ul style="list-style-type: none"> - UC20 (CADASTRO ALUNO): Desenvolvimento - UC02 (LOGIN): Revisão - UC21 (TELA INICIAL ALUNO): Desenvolvimento 		
Sprint 5	01/08/2022	15/08/2022
<ul style="list-style-type: none"> - UC20 (CADASTRO ALUNO): Revisão - UC21 (TELA INICIAL ALUNO): Desenvolvimento - UC22 (INSCRIÇÃO NA TURMA): Desenvolvimento 		
Sprint 6	15/08/2022	29/08/2022
<ul style="list-style-type: none"> - APP ALUNO: Padronização das requests para API - UC21 (TELA INICIAL ALUNO): Revisão - UC22 (INSCRIÇÃO NA TURMA): Revisão - UC28 (ACESSAR TURMA): Desenvolvimento 		
Sprint 7	29/08/2022	12/09/2022
<ul style="list-style-type: none"> - UC27 (LISTAR EXERCÍCIOS DA TURMA): Desenvolvimento - UC28 (ACESSAR TURMA): Revisão 		
Sprint 8	12/09/2022	26/09/2022
<ul style="list-style-type: none"> - UC27 (LISTAR EXERCÍCIOS DA TURMA): Revisão - UC26 (CRIAR RESPOSTA PARA EXERCÍCIO): Desenvolvimento 		

Sprint 9	26/09/2022	03/09/2022
<ul style="list-style-type: none"> - UC26 (CRIAR RESPOSTA PARA EXERCÍCIO): Revisão - UC23 (VER PERFIL): Desenvolvimento e Revisão - UC24 (ATUALIZAR INFORMAÇÕES ALUNO): Desenvolvimento e Revisão 		
Intervalo de Sprints (Recesso)		
Sprint 10	24/10/2022	07/11/2022
<ul style="list-style-type: none"> - Início do desenvolvimento do APP Docente - Configuração do projeto APP Docente - Remoção de lógica de professor do APP Aluno 		
Sprint 11	07/11/2022	21/11/2022
<ul style="list-style-type: none"> - UC01 (CADASTRAR PROFESSOR): Desenvolvimento - UC04 (TELA INICIAL PROFESSOR): Desenvolvimento - UC08 (LISTAR TURMA): Desenvolvimento 		
Sprint 12	21/11/2022	05/12/2022
<ul style="list-style-type: none"> - UC01 (CADASTRAR PROFESSOR): Revisão - UC04 (TELA INICIAL PROFESSOR): Desenvolvimento - UC08 (LISTAR TURMA): Revisão 		
Sprint 13	05/12/2022	19/12/2022
<ul style="list-style-type: none"> - UC03 (ATUALIZAR INFORMAÇÕES PROFESSOR): Desenvolvimento e revisão - UC04 (TELA INICIAL PROFESSOR): Revisão - UC05 (CRIAR TURMA): Desenvolvimento e revisão 		
Intervalo de Sprints (Recesso)		
Sprint 14	17/01/2023	31/01/2023
<ul style="list-style-type: none"> - Definição do sistema de gamificação (pontuações, níveis, tarefas e medalhas) - UC06 (EDITAR TURMA): Desenvolvimento - UC10 (LISTAR EXERCÍCIOS): Desenvolvimento 		
Sprint 15	31/01/2023	14/02/2023
<ul style="list-style-type: none"> - Implementação do sistema de gamificação no Back-end - UC06 (EDITAR TURMA): Revisão - UC07 (REMOVER TURMA): Desenvolvimento e Revisão - UC10 (LISTAR EXERCÍCIOS): Revisão - UC11 (ACESSAR EXERCÍCIOS): Desenvolvimento 		
Sprint 16	14/02/2023	28/02/2023

<ul style="list-style-type: none"> - UC09 (CRIAÇÃO DE EXERCÍCIOS): Desenvolvimento - UC11 (ACESSAR EXERCÍCIOS): Revisão - UC25 (VISITAR RANKING): Desenvolvimento 		
Sprint 17	28/02/2023	14/03/2023
<ul style="list-style-type: none"> - UC09 (CRIAÇÃO DE EXERCÍCIOS): Revisão - UC12 (EDIÇÃO DE EXERCÍCIOS): Desenvolvimento - UC13 (REMOVER EXERCÍCIO): Desenvolvimento - UC25 (VISITAR RANKING): Revisão 		
Sprint 18	14/03/2023	28/03/2023
<ul style="list-style-type: none"> - UC12 (EDIÇÃO DE EXERCÍCIOS): Revisão - UC13 (REMOVER EXERCÍCIO): Revisão - UC14 (CRIAÇÃO DE CASO DE TESTE): Desenvolvimento - UC15 (LISTAR CASOS DE TESTE): Desenvolvimento 		
Sprint 19	28/03/2023	11/04/2023
<ul style="list-style-type: none"> - UC14 (CRIAÇÃO DE CASO DE TESTE): Revisão - UC15 (LISTAR CASOS DE TESTE): Revisão - UC16 (EDITAR CASOS DE TESTE): Desenvolvimento e Revisão - UC17 (REMOVER CASOS DE TESTE): Desenvolvimento e Revisão 		
Sprint 20	11/04/2023	25/04/2023
<ul style="list-style-type: none"> - UC18 (EXECUTAR CASOS DE TESTE): Desenvolvimento e Revisão 		
Sprint 21	25/04/2023	09/05/2023
<ul style="list-style-type: none"> - UC19 (GAMIFICAÇÃO PROFESSOR): Desenvolvimento e Revisão 		
Sprint 22	09/05/2023	23/05/2023
<ul style="list-style-type: none"> - Implementação de tela de ranking no Front-end Web - Hospedagem do projeto sandbox na DigitalOcean 		
Sprint 23	23/05/2023	06/06/2023
<ul style="list-style-type: none"> - Correção de bugs APP Docente - Desenvolvimento do Documento - UC29 (VER RESPOSTAS): Desenvolvimento e Revisão 		
Sprint 24	06/06/2023	20/06/2023
<ul style="list-style-type: none"> - Correção de bugs APP Docente - Revisão do Documento 		
Sprint 25	20/06/2023	28/0/2023

- Correção de bugs APP Aluno
- Revisão do Documento

APÊNDICE I - ROTAS DA API

usuario-controller <small>Usuario Controller</small>	
GET	/usuario ListarUsuarios
POST	/usuario cadastrarUsuario
GET	/usuario/{codigo} buscarUsuario
PUT	/usuario/{codigo} atualizarUsuario
DELETE	/usuario/{codigo} removerUsuario
PUT	/usuario/admin/senha adminAtualizarSenhaUsuario
GET	/usuario/mim buscarUsuarioPorToken
GET	/usuario/ranking buscarRankingUsuarios
PUT	/usuario/senha atualizarSenhaUsuario

turma-controller <small>Turma Controller</small>	
GET	/turma ListarTurmas
POST	/turma cadastrarTurma
GET	/turma/{codigo} buscarTurmaPorCodigo
PUT	/turma/{codigo} atualizarTurma
DELETE	/turma/{codigo} removerTurma
DELETE	/turma/{codigoTurma}/aluno/{codigoAluno} removerAlunoTurma
POST	/turma/aluno/{codigoAluno} cadastrarAlunoTurma
GET	/turma/codigo/acao/{codigoAcesso} consultarDisponibilidadeCodigoAcesso
GET	/turma/mim buscarTurmasDoAluno

tag-controller <small>Tag Controller</small>	
GET	/tag ListarTags
POST	/tag cadastrarTag
GET	/tag/{codigo} buscarTag
PUT	/tag/{codigo} atualizarTag
DELETE	/tag/{codigo} removerTag

resposta-controller <small>Resposta Controller</small>	
POST	/resposta cadastrarResposta
GET	/resposta/{codigo} buscarResposta
PUT	/resposta/{codigo} atualizarResposta
DELETE	/resposta/{codigo} removerResposta
GET	/resposta/aluno/{codigoAluno} ListarRespostasDeUmAluno
GET	/resposta/aluno/{codigoAluno}/exercico/{codigoExercicio} ListarRespostasDeUmAlunoAUmExercicio
GET	/resposta/exercicio/{codigoExercicio} ListarRespostasDeUmExercicio

mensagem-controller Mensagem Controller		▼
POST	/mensagem	enviarMensagem
DELETE	/mensagem/{id}	deletarMensagem
GET	/mensagem/{idUserario}	listarMensagens
PUT	/mensagem/alterar/{id}	atualizarMensagem

medalha-controller Medalha Controller		▼
GET	/medalha	ListarMedalhas
POST	/medalha	cadastrarMedalha
PUT	/medalha/{id}	atualizarMedalha
DELETE	/medalha/{id}	removerLinguagem

lista-exercicio-turma-controller Lista Exercicio Turma Controller		▼
POST	/lista/exercicio/	cadastrarListaExercicioTurma
PUT	/lista/exercicio/{codigo}	atualizarListaExercicioTurma
DELETE	/lista/exercicio/{codigo}	removerListaExercicioTurma
GET	/lista/exercicio/{codigolista}	buscarListaExercicioTurma
GET	/lista/exercicio/turma/{codigoTurma}	ListarListaExerciciosTurma

linguagem-controller Linguagem Controller		▼
GET	/linguagem	ListarLinguagem
POST	/linguagem	cadastrarLinguagem
GET	/linguagem/{codigo}	buscarLinguagem
PUT	/linguagem/{codigo}	atualizarLinguagem
DELETE	/linguagem/{codigo}	removerLinguagem

instituicao-controller Instituicao Controller		▼
GET	/instituicao	ListarInstituicoes
POST	/instituicao	cadastrarInstituicao
GET	/instituicao/{codigo}	buscarInstituicao
PUT	/instituicao/{codigo}	atualizarInstituicao
DELETE	/instituicao/{codigo}	removerInstituicao
GET	/instituicao/combo	buscarComboInstituicao
POST	/instituicao/completa	cadastrarInstituicaoCompleta

instituicao-contato-controller Instituicao Contato ControllerGET /**instituicao/contato** ListarInstituicoesContatosPOST /**instituicao/contato** cadastrarInstituicaoContatoGET /**instituicao/contato/{codigo}** buscarInstituicaoContatoPUT /**instituicao/contato/{codigo}** atualizarInstituicaoContatoDELETE /**instituicao/contato/{codigo}** removerInstituicaoContato**exercicio-controller** Exercicio ControllerGET /**exercicio** ListarExerciciosPublicosPOST /**exercicio** cadastrarExercicioGET /**exercicio/{codigo}** buscarExercicioPUT /**exercicio/{codigo}** atualizarExercicioDELETE /**exercicio/{codigo}** removerExercicioPOST /**exercicio/{codigoExercicio}/linguagem/{codigoLinguagem}** cadastrarLinguagemEmUmExercicioDELETE /**exercicio/{codigoExercicio}/linguagem/{codigoLinguagem}** removerLinguagemDeUmExercicioPOST /**exercicio/{codigoExercicio}/lista/{codigoLista}** cadastrarExercicioEmUmaListaDELETE /**exercicio/{codigoExercicio}/lista/{codigoLista}** removerExercicioDeUmaListaDELETE /**exercicio/{codigoExercicio}/tag/{codigoTag}** removerTagDeUmExercicioPOST /**exercicio/{codigoExercicio}/tag/{nomeTag}** cadastrarTagEmUmExercicioPOST /**exercicio/{codigoExercicio}/turma/{codigoTurma}** cadastrarExercicioParaOutraTurmaPOST /**exercicio/completo** cadastrarExercicioCompleto**dica-exercicio-controller** Dica Exercicio ControllerPOST /**exercicio/dica** cadastrarDicaExercicioGET /**exercicio/dica/{codigo}** buscarDicaExercicioPUT /**exercicio/dica/{codigo}** atualizarDicaExercicioDELETE /**exercicio/dica/{codigo}** removerDicaExercicioGET /**exercicio/dica/exercicio/{codigoExercicio}** ListarDicasExercicio

caso-teste-controller		Caso Teste Controller	▼
POST	/caso/teste	cadastrarCasoTeste	
GET	/caso/teste/{codigo}	buscarCasoTeste	
POST	/caso/teste/{codigo}	testarCasoTeste	
PUT	/caso/teste/{codigo}	atualizarCasoTeste	
DELETE	/caso/teste/{codigo}	removerCasoTeste	
GET	/caso/teste/exercicio/{codigoExercicio}	ListarCasosTesteDeUmExercicio	

basic-error-controller		Basic Error Controller	▼
GET	/error	erroHtml	
HEAD	/error	erroHtml	
POST	/error	erroHtml	
PUT	/error	erroHtml	
DELETE	/error	erroHtml	
OPTIONS	/error	erroHtml	
PATCH	/error	erroHtml	