

UNIVERSIDADE FEDERAL DO PARANÁ

ALEXANDRE YUKIO DE SOUSA SEMBO

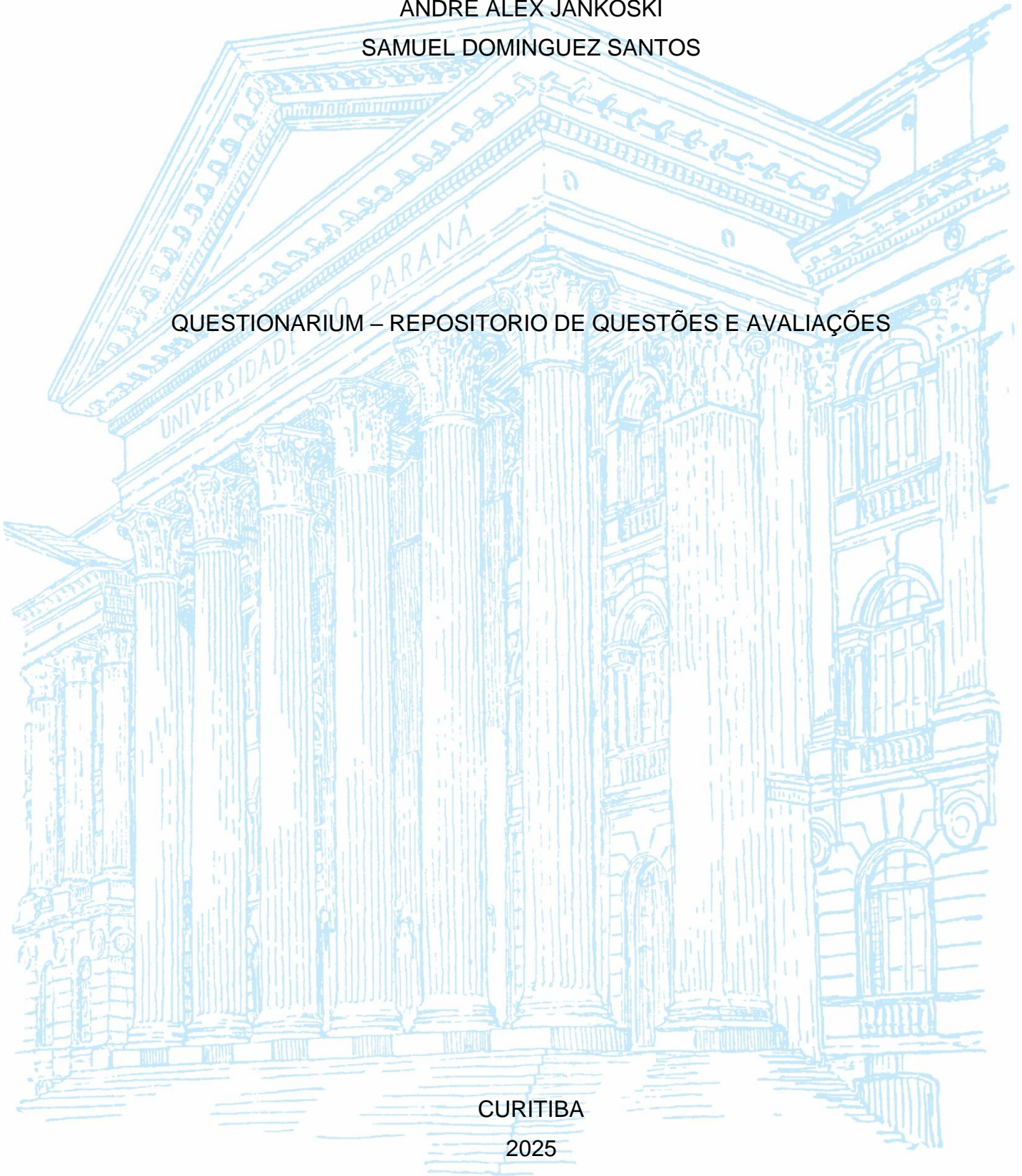
ANDRÉ ALEX JANKOSKI

SAMUEL DOMINGUEZ SANTOS

QUESTIONARIUM – REPOSITORIO DE QUESTÕES E AVALIAÇÕES

CURITIBA

2025



ALEXANDRE YUKIO DE SOUSA SEMBO  
ANDRÉ ALEX JANKOSKI  
SAMUEL DOMINGUEZ SANTOS

## QUESTIONARIUM – REPOSITORIO DE QUESTÕES E AVALIAÇÕES

Trabalho de conclusão de curso apresentando ao curso de Tecnologia em Análise e Desenvolvimento de Sistemas da Universidade Federal do Paraná, como requisito parcial à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: Prof. Dr. Razer Anthom Nizer Rojas Montaña

CURITIBA

2025



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 - <https://ufpr.br/>

Ata de Reunião

## **TERMO DE APROVAÇÃO**

ALEXANDRE YUKIO DE SOUSA SEMBO  
ANDRÉ ALEX JANKOSKI  
SAMUEL DOMINQUEZ SANTOS

## **QUESTIONARIUM – REPOSITÓRIO DE QUESTÕES E AVALIAÇÕES**

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. Razer Anthom Nizer Rojas Montaña  
Orientadora – SEPT/UFPR

Prof. Alexander Robert Kutzke  
SEPT/UFPR

Prof.a Rafaela Mantovani Fontana  
SEPT/UFPR

Curitiba, 03 de julho de 2025.

---



Documento assinado eletronicamente por **ALEXANDER ROBERT KUTZKE, PROFESSOR DO MAGISTERIO SUPERIOR**, em 03/07/2025, às 18:27, conforme art. 1º, III, "b", da Lei 11.419/2006.

---



Documento assinado eletronicamente por **RAZER ANTHOM NIZER ROJAS MONTANO, PROFESSOR DO MAGISTERIO SUPERIOR**, em 03/07/2025, às 18:28, conforme art. 1º, III, "b", da Lei 11.419/2006.

---



Documento assinado eletronicamente por **RAFAELA MANTOVANI FONTANA, PROFESSOR ENS BASICO TECN TECNOLOGICO**, em 03/07/2025, às 18:28, conforme art. 1º, III, "b", da Lei 11.419/2006.

---



A autenticidade do documento pode ser conferida [aqui](#) informando o código verificador **7921488** e o código CRC **B6234467**.

---

## **AGRADECIMENTOS**

Ao nosso professor orientador, Razer Anthom Nizer Rojas Montaña, pela paciência, incentivo e atenção que foram fundamentais não apenas na orientação deste trabalho, mas durante todo o curso.

Aos nossos familiares e amigos, por estarem presentes ao longo dessa jornada desafiadora, oferecendo apoio incondicional.

À Universidade Federal do Paraná, por nos proporcionar crescimento pessoal e nos moldar como cidadãos.

Ao curso de Tecnologia em Análise e Desenvolvimento de Sistemas e aos professores que compuseram nossa formação, por aprimorarem nossos conhecimentos e habilidades.

## RESUMO

A jornada de trabalho dos professores no Brasil é frequentemente exaustiva e desgastante, como evidenciado por diversos estudos e dados históricos. Professores frequentemente precisam lidar com turmas grandes e múltiplas escolas, o que amplia suas responsabilidades além das salas de aula, afetando sua qualidade de vida e eficiência profissional. Este trabalho propõe o desenvolvimento de uma plataforma online e um aplicativo móvel multiplataforma que apoiam docentes na criação, organização e gestão de bancos de questões, além da elaboração e correção automática de avaliações. O sistema foi desenvolvido utilizando a linguagem de programação Java, em conjunto com os frameworks Spring Boot para o *backend* e Angular para o *frontend*, seguindo a metodologia ágil Scrum. A plataforma visa otimizar o tempo docente, reduzir a carga administrativa e promover uma abordagem mais personalizada e eficaz no processo de ensino e aprendizagem. Também é proposto facilitar a criação de questões, organizar e gerenciar bancos de dados, elaborar avaliações ágeis, automatizar a correção, promover a personalização do ensino e reduzir a carga administrativa dos professores. Como resultados alcançados, o sistema permite a criação de forma organizada de avaliações e questões, automatiza processos de correção, gera relatórios detalhados e facilita a gestão de avaliações aplicadas.

Palavras-chave: avaliações; banco de questões; correção automática; repositório de questões;

## ABSTRACT

The workload of teachers in Brazil is frequently exhausting and demanding, as evidenced by various studies and historical data. Teachers often have to deal with large classes and multiple schools, extending their responsibilities beyond classroom activities and negatively impacting their quality of life and professional effectiveness. This study proposes the development of an online platform and a cross-platform mobile application to support educators in creating, organizing, and managing question banks, as well as developing and automatically grading assessments. The system was developed using the Java programming language, with Spring Boot for the backend and Angular for the frontend, following the Scrum agile methodology. The platform aims to optimize teachers' time, reduce administrative burdens, and promote a more personalized and effective teaching and learning approach. It also proposes facilitating question creation, organizing and managing databases, developing agile assessments, automating corrections, personalizing teaching, and reducing teachers' administrative workloads. The results achieved include organized creation of assessments and questions, automated correction processes, detailed report generation, and simplified management of applied assessments.

Keywords: assessments; question bank; automatic grading; question repository.

## LISTA DE FIGURAS

FIGURA 1 - CICLO DO SCRUM .....	25
FIGURA 2 - TELA DE CADASTRO PRUVO IA .....	29
FIGURA 3 – TELA DE CRIAR QUESTÃO PRUVO AI .....	30
FIGURA 4 – TELA DE TIPO DE QUESTÕES PRUVO AI .....	31
FIGURA 5 – TELA DE AVALIAÇÃO PRUVO AI .....	31
FIGURA 6 – TELA INICIAL GRADEPEN .....	34
FIGURA 7 – TELA DE AVALIAÇÕES GRADEPEN .....	34
FIGURA 8 – TELA DE GERENCIAMENTO DE CRÉDITOS GRADEPEN .....	35
FIGURA 9 – TELA DE DESCRIÇÃO DO WEBSITE GRADEPEN .....	35
FIGURA 10 – ARQUITETURA DO SISTEMA .....	60
FIGURA 11 – TELA LANDING PAGE .....	60
FIGURA 12 – TELA LOGIN .....	61
FIGURA 13 – TELA LOGIN APÓS CLICAR EM ESQUECI A SENHA .....	61
FIGURA 14 – TELA CADASTRO .....	62
FIGURA 15 - TELA INICIAL .....	62
FIGURA 16 - TELA REDEFINIÇÃO DE SENHA .....	63
FIGURA 17 - TELA EDIÇÃO DE PERFIL .....	63
FIGURA 18 - TELA EDIÇÃO DE SENHA .....	64
FIGURA 19 - TELA VISUALIZAÇÃO DE QUESTÃO .....	64
FIGURA 20 - TELA CADASTRO E EDIÇÃO DE QUESTÃO .....	65
FIGURA 21 - TELA DE GERAÇÃO AUTOMÁTICA DE QUESTÃO .....	66
FIGURA 22 - TELA DE LISTAGEM DE AVALIAÇÕES .....	66
FIGURA 23 - TELA DE CRIAÇÃO DE APLICAÇÃO .....	67
FIGURA 24 - TELA DE LISTAGEM DE APLICAÇÃO .....	67
FIGURA 25 - TELA DE CRIAÇÃO/EDIÇÃO DE AVALIAÇÃO .....	68
FIGURA 26 - TELA DE SELEÇÃO DE QUESTÕES .....	68
FIGURA 27 - TELA DE RELATÓRIOS DE AVALIAÇÃO .....	69
FIGURA 28 - TELA DE RELATÓRIO DE APLICAÇÃO .....	69
FIGURA 29 - TELA SOBRE .....	70
FIGURA 30 - TELA LISTAR CABEÇALHOS .....	70
FIGURA 31 - TELA CRIAR/EDITAR CABEÇALHOS .....	71
FIGURA 32 – TELA DE LOGIN .....	72

FIGURA 33 - TELA DE SCANNER .....	72
FIGURA 34- TELA DE SELEÇÃO DE IMAGEM .....	73
FIGURA 35 - TELA DE EDIÇÃO DO GABARITO .....	74
FIGURA 36 - TELA DE RESULTADO.....	75

## LISTA DE QUADROS

QUADRO 1 – COMPARAÇÃO ENTRE TECNOLOGIAS SEMELHANTES .....	36
QUADRO 2 – RESUMO DAS SPRINTS DA PRIMEIRA FASE.....	45
QUADRO 3 – PLANEJAMENTO DAS SPRINTS DA SEGUNDA FASE .....	45
QUADRO 4 – TAREFAS E RESPONSABILIDADES .....	46

## SUMÁRIO

<b>1 INTRODUÇÃO</b> .....	<b>10</b>
1.1 PROBLEMA .....	12
1.2 OBJETIVOS .....	13
1.2.1 Objetivo geral .....	13
1.2.2 Objetivos específicos.....	13
1.3 JUSTIFICATIVA .....	14
1.4 ESTRUTURA DO DOCUMENTO.....	15
<b>2 FUNDAMENTAÇÃO TEORICA</b> .....	<b>16</b>
2.1 PAPEL DAS AVALIAÇÕES NA EDUCAÇÃO.....	16
2.1.1 Impacto das Avaliações no Ensino.....	17
2.1.2 Tipos de avaliações.....	18
2.2 TRANSFORMAÇÃO DIGITAL NA EDUCAÇÃO .....	19
2.3 AVALIAÇÃO EDUCACIONAL E TECNOLOGIA .....	20
2.4 FUNDAMENTAÇÃO DA TECNOLOGIA .....	21
2.4.1 Engenharia de software.....	22
2.4.2 Desenvolvimento ágil .....	23
2.4.3 Scrum .....	24
2.4.4 Arquitetura de Microserviços .....	26
2.4.5 Padrões de Projeto.....	26
2.4.5.1 Database per service.....	27
2.4.5.2 API Gateway.....	27
2.4.5.3 SAGA.....	27
2.4.5.4 Circuit Breaker .....	28
2.5 SOFTWARES SEMELHANTES .....	28
2.5.1 Pruvo AI.....	29
2.5.2 Gradepen .....	32
2.5.3 Quadro Comparativo .....	35
2.6 CONSIDERAÇÕES .....	37
<b>3 MATERIAIS E MÉTODOS</b> .....	<b>38</b>
3.1 MÉTODO.....	38
3.2 MODELAGEM .....	39
3.2.1 Modelagem do Sistema.....	40

3.2.1.1 Entidades e Relacionamentos .....	40
3.2.1.2 Clonagem e Imutabilidade das Avaliações e Questões .....	41
3.2.2 Diagrama de Casos de Uso .....	41
3.2.3 Histórias de Usuário .....	42
3.2.4 Protótipos de Interface .....	42
3.2.5 Diagrama de Classes .....	43
3.2.6 Diagrama de Sequência .....	43
3.2.7 Diagrama Físico do Banco de Dados .....	44
3.3 CRONOGRAMA .....	44
3.3.1 Fase de Modelagem .....	44
3.3.2 Fase de Desenvolvimento .....	45
3.4 RESPONSABILIDADES .....	46
3.5 MATERIAIS .....	47
3.5.1 Figma .....	47
3.5.2 Draw.io e Astah .....	47
3.5.3 GitHub .....	48
3.5.4 Docker .....	48
3.5.5 Postman .....	49
3.5.6 VSCode .....	49
3.5.7 Linguagem de Modelagem Unificada .....	49
3.5.8 Angular .....	50
3.5.9 Java e Spring Boot .....	50
3.5.10 REST .....	51
3.5.11 PostgreSQL e pgAdmin .....	51
3.5.12 MongoDB .....	52
3.5.13 React Native .....	52
3.5.14 RabbitMQ .....	53
3.5.15 OpenCV .....	53
3.5.16 Hardware utilizado .....	54
3.6 DESENVOLVIMENTO DO PROJETO .....	54
3.6.1 Sprint 1 .....	54
3.6.2 Sprint 2 .....	55
3.6.3 Sprint 3 .....	55
3.6.4 Sprint 4 .....	55

3.6.5 Sprint 5.....	55
3.6.6 Sprint 6.....	56
3.6.7 Sprint 7.....	56
3.6.8 Sprint 8.....	56
3.6.9 Sprint 9.....	57
3.6.10 Sprint 10.....	57
3.6.11 Sprint 11.....	57
3.6.12 Sprint 12.....	58
3.6.13 Sprint 13.....	58
3.6.14 Sprint 14.....	58
<b>4 APRESENTAÇÃO DO SISTEMA.....</b>	<b>59</b>
4.1 TELAS DA APLICAÇÃO WEB.....	60
4.2 TELAS DA APLICAÇÃO MOBILE.....	71
<b>5 CONSIDERAÇÕES FINAIS.....</b>	<b>76</b>
5.1 ESTUDOS FUTUROS.....	77
<b>REFERÊNCIAS.....</b>	<b>78</b>
<b>APÊNDICE A – DIAGRAMA DE CASOS DE USO.....</b>	<b>85</b>
<b>APÊNDICE B – HISTÓRIAS DE USUÁRIO E PROTOTIPAÇÃO.....</b>	<b>86</b>
<b>APÊNDICE C – DIAGRAMAS DE CLASSES.....</b>	<b>121</b>
<b>APÊNDICE D – DIAGRAMAS DE SEQUÊNCIA.....</b>	<b>122</b>
<b>APÊNDICE E – DIAGRAMA FÍSICO DO BANCO DE DADOS.....</b>	<b>132</b>
<b>APÊNDICE F – IMPLEMENTAÇÃO DA INTELIGÊNCIA ARTIFICIAL NO SOFTWARE.....</b>	<b>133</b>

## 1 INTRODUÇÃO

A jornada de trabalho dos professores no Brasil é frequentemente descrita como exaustiva (Brito *et al.*, 2021) e desgastante por diversos autores, especialmente por aqueles situados nos grandes centros urbanos do país (Barbosa *et al.*, 2021). Este cenário não é recente, sendo evidenciado por dados históricos que mostram que, desde o século passado, as jornadas de trabalho poderiam ultrapassar as 44 horas semanais (Brasil, 1943).

Pela Lei nº 9.394, de 1996, que estabelece as Diretrizes e Bases da Educação Nacional, o artigo 31, inciso II, determina que a carga horária escolar mínima anual é de 800 horas, distribuídas em no mínimo 200 dias de trabalho educacional (Brasil, 1996). Dentro da jornada de trabalho do professor, há basicamente uma divisão entre o tempo gasto com atividades em sala de aula e atividades fora da sala de aula. Essas últimas são denominadas horas-atividade, período em que o professor desempenha funções da docência, como planejamento de aulas e atividades, atendimentos à comunidade escolar, correção de avaliações, reuniões pedagógicas e outras correlatas, devendo ser cumpridas no local de exercício. O tempo em sala de aula é estipulado pela Lei nº 11.738 de 2008, que descreve que dois terços da jornada de trabalho do professor devem ser voltados para atividades de interação com os educandos (Brasil, 2008). Contudo, as definições de tempo das horas-atividade podem variar conforme o estado, a rede (pública e privada) e a modalidade de ensino (infantil, fundamental, médio, técnico e superior).

No Paraná, de acordo com a Lei Ordinária nº 13.807, de 2002, é instituído um percentual de 20% de hora-atividade da jornada de trabalho para todos os professores do estado (Brasil, 2002). Em tese, em uma jornada de trabalho de 40 horas semanais, o professor deve cumprir cerca de 48 horas-aula (sendo 1 hora-aula, de forma geral, correspondente a 50 minutos), das quais cerca de 10 são horas-atividade. Porém, mesmo com essa definição, esses valores ainda podem variar de acordo com a instituição ou a rede de ensino. Conforme disponibilizado no portal da Secretaria de Educação do Estado do Paraná, uma jornada de trabalho de 40 horas é composta de 30 horas de regência e 10 horas de hora-atividade.

Desde 2012, o Projeto de Lei nº 4731 alterou o artigo 25 da Lei de Diretrizes e Bases (Brasil, 1996), definindo o número máximo de alunos por turma: 25 alunos na pré-escola e nos dois primeiros anos do ensino fundamental, e 35 alunos nos anos

subsequentes do ensino fundamental e no ensino médio. No entanto, não há restrições quanto ao número máximo de turmas ou instituições em que um professor pode trabalhar simultaneamente. Além disso, na prática, os resultados frequentemente divergem dos previstos em lei, variando de acordo com a rede, estado e modalidade de ensino, assim como ocorre com as definições de hora-atividade.

Em um estudo realizado por Moriconi *et al.* (2021) sobre a quantidade de turmas e alunos por professores com base nos dados do Censo Escolar de 2020, eles observaram que as diferenças significativas na carga de trabalho dos professores dos anos finais do ensino fundamental no Brasil em comparação com os Estados Unidos, França e Japão é demasiadamente alta. No Brasil, quase metade dos professores trabalham em mais de uma escola e 30% atuam em mais de uma rede, enquanto a cultura dos outros países do estudo demonstra que a maioria dos professores trabalham em apenas uma instituição. Isso implica que quase metade dos professores brasileiros do Ensino Fundamental precisam de muito mais tempo para atividades envolvidas com seu trabalho que vão além de dar aula.

No trabalho da D3E (Moriconi *et al.*, 2021), é analisado o número de alunos por professor no Brasil, e esses dados devem ser tratados com cautela, considerando o tamanho do país e as grandes variações em realidades, estruturas educacionais e políticas. Não é adequado utilizar um número absoluto da média de alunos por professor em todo o país, pois enquanto algumas redes têm uma média inferior a 50 alunos por professor, em outras áreas do conhecimento e redes educacionais, essa média pode ultrapassar 500 alunos por docente.

A pesquisa conduzida por Alvarenga sobre o tempo de trabalho e o tempo de ensino aplicados à realidade do professor brasileiro revela que o tempo de trabalho é significativamente maior do que o tempo dedicado ao ensino. As entrevistas realizadas pelos pesquisadores demonstram que os professores utilizam mais tempo do que o previsto em lei para a jornada de trabalho, a fim de gerenciar atividades fora da sala de aula (Alvarenga, 2012). Entre essas atividades estão: elaboração de atividades e planos de aula; correção de avaliações e trabalhos; pesquisa, estudo e atualização em temas a serem abordados em aula; cumprimento de deveres burocráticos do sistema de ensino, entre outras tarefas.

Uma maneira de reduzir a carga de trabalho dos professores é por meio da utilização das Tecnologias da Informação e Comunicação (TIC), que, com o avanço significativo das tecnologias, estão cada vez mais presentes no cotidiano docente.

Pesquisas sobre o uso de Inteligência Artificial (IA) destacam não apenas a crescente presença dessas ferramentas no dia a dia, mas também o vasto potencial que elas oferecem para a educação. Entre as possibilidades proporcionadas pelas TIC e IA estão a automação de tarefas administrativas, a personalização do ensino, a análise de dados educacionais e o apoio na elaboração de conteúdo, contribuindo para um ensino mais eficiente e menos oneroso para os professores.

Entre essas possibilidades estão a adaptação da aprendizagem, a acessibilidade, a análise de dados e a otimização de diversos processos (Ferreira *et al.*, 2023).

Assim, este trabalho tem por objetivo contribuir para a melhoria das condições de trabalho dos professores por meio do desenvolvimento de uma solução tecnológica que otimize processos avaliativos.

## 1.1 PROBLEMA

O sistema educacional brasileiro é caracterizado por uma notável ênfase em avaliações no formato de provas, as quais desempenham um papel central em diversas etapas da jornada acadêmica dos estudantes. Começando no ensino fundamental e se estendendo a provas de maior porte, como o vestibular e Exame Nacional do Ensino Médio, as avaliações são uma constante na vida dos estudantes brasileiros (Castro, 2009).

Dessa forma, as provas deixam de ser apenas uma avaliação de conhecimento, e se transformam em um fator crucial para o sucesso acadêmico, e até mesmo em uma porta de entrada para o ensino superior. Todavia, o processo de criação, aplicação e correção dessas avaliações pode se mostrar complexo, desafiador e demorado para educadores e profissionais do ensino. Isso ocorre devido à importância de uso de questões variadas e condizentes com as necessidades de cada turma, à necessidade de garantir uma correção justa e padronizada, e à importância do *feedback* direcionado, por exemplo (Medeiros, 2009).

Com isso, surge a necessidade de um sistema capaz de auxiliar os professores, educadores e profissionais de ensino em todo o processo de criação, aplicação e correção de uma avaliação.

Baseado nesta situação, este trabalho propõe o desenvolvimento de um sistema web e um aplicativo móvel multiplataforma, que oferece uma solução para

simplificar os processos envolvidos na elaboração, aplicação, correção e análise de avaliações.

Desde a montagem das provas até a correção e geração de relatórios, o sistema visa auxiliar os professores a poupar tempo e esforço em relação às tarefas repetitivas e trabalhosas deste processo. Ao fornecer ferramentas eficientes e fáceis de usar, o sistema ajuda os professores a gerenciar melhor seu tempo, permitindo que se concentrem mais em planejamento de aulas e foco individual em seus alunos, o que, por sua vez, contribui para uma experiência de ensino e aprendizagem mais eficaz e satisfatória para todos.

## 1.2 OBJETIVOS

Esta seção busca apresentar os objetivos gerais e específicos que permeiam o desenvolvimento da plataforma.

### 1.2.1 Objetivo geral

Desenvolver uma plataforma online para auxiliar professores na criação, organização e gestão de bancos de questões, bem como na elaboração e correção automática de avaliações. A plataforma otimiza o tempo docente ao centralizar materiais de ensino e simplificar processos avaliativos.

### 1.2.2 Objetivos específicos

1. **Facilitar a criação de questões:** Proporcionar uma interface com navegação simplificada, que permita aos professores criar e categorizar questões com diferentes formatos (objetivas, dissertativas etc.) e associá-las ao conteúdo das disciplinas, utilizando um número reduzido de etapas;
2. **Organizar e gerenciar bancos de questões:** Desenvolver funcionalidades que permitam armazenar, classificar e recuperar questões de forma estruturada, possibilitando sua reutilização em avaliações futuras;
3. **Elaborar avaliações de forma ágil:** Implementar ferramentas para montagem de avaliações com base nas questões armazenadas, com opções de seleção, ordenação e edição conforme critérios definidos pelo usuário;

4. **Automatizar a correção de avaliações:** Integrar um sistema de correção automática que permita a avaliação das respostas dos alunos, com *feedback* instantâneo e relatórios detalhados para os professores, tudo isso por um aplicativo móvel, tornando a solução mais acessível;
5. **Promover a personalização do ensino:** Oferecer recursos que possibilitem ajustes nas avaliações e nas questões com base no desempenho e nas necessidades individuais dos alunos, permitindo uma abordagem mais personalizada e eficaz no processo educativo;
6. **Reduzir a carga administrativa dos professores:** Desenvolver soluções que diminuam o tempo e o esforço necessários para a criação, organização e correção de avaliações, permitindo que os professores se concentrem mais em atividades pedagógicas e no desenvolvimento de estratégias de ensino;
7. **Gerar relatórios de desempenho de alunos,** avaliações e até mesmo de questões específicas, sendo utilizada como uma ferramenta que possibilita o aprimoramento do conteúdo dos profissionais do ensino.

### 1.3 JUSTIFICATIVA

A crescente demanda por atividades avaliativas e a complexidade associada à criação e correção de questões têm resultado em uma sobrecarga significativa para os professores. Esse cenário não apenas afeta negativamente a qualidade do ensino, mas também o bem-estar e a saúde dos docentes, que frequentemente enfrentam jornadas de trabalho extenuantes e excesso de tarefas administrativas.

A implementação de uma plataforma online para a gestão de questões e avaliações surge como uma solução inovadora para enfrentar esses desafios. Ao automatizar a criação, organização e correção de avaliações, a plataforma tem o potencial de aliviar a carga de trabalho dos professores, permitindo-lhes dedicar mais tempo a atividades pedagógicas e ao desenvolvimento profissional.

Além disso, a plataforma contribuirá para a personalização do ensino, proporcionando ferramentas para ajustar as avaliações de acordo com as necessidades e o desempenho dos alunos. Isso pode levar a uma melhoria significativa na eficácia do ensino e na experiência de aprendizagem dos alunos.

Em um contexto educacional cada vez mais dinâmico e exigente, a integração de tecnologias que simplifiquem e otimizem processos administrativos é essencial

para garantir que os professores possam se concentrar em sua principal missão: promover a aprendizagem e o desenvolvimento dos alunos. A plataforma proposta visa, portanto, melhorar a qualidade do ensino e o bem-estar dos docentes, alinhando-se às tendências tecnológicas e às necessidades contemporâneas da educação.

#### 1.4 ESTRUTURA DO DOCUMENTO

O documento objetiva detalhar todo o processo de desenvolvimento do sistema, assim como as ferramentas usadas para tal e as fundamentações e motivos para a elaboração.

Este primeiro capítulo, Capítulo 1, trata dos motivos pelos quais o sistema foi desenvolvido, os objetivos do projeto e a necessidade de ferramentas para auxiliar os professores no seu trabalho.

O Capítulo 2, fundamenta o desenvolvimento do software proposto, apresentando os conceitos relacionados ao ensino e avaliações. Conceitua os tipos de avaliações e as transformações com relação ao trabalho exercido pelos professores ao longo do tempo.

No Capítulo 3 são apresentados os modelos de processo de Engenharia de Software, as etapas, cronogramas e as tecnologias utilizadas.

O Capítulo 4, traz a arquitetura do sistema por meio de um fluxo de dados, comunicação entre componentes, mostrando o mapeamento das funcionalidades e suas respectivas telas.

No Capítulo 5, são apresentadas as considerações finais, com uma reflexão sobre a solução proposta, os resultados obtidos e sua importância para agregar valor ao trabalho do professor. Também são discutidas a contínua evolução do sistema a introdução de novas funcionalidades.

## 2 FUNDAMENTAÇÃO TEORICA

Para fundamentar o desenvolvimento do software proposto, apresenta-se neste capítulo, os conceitos relacionados ao ensino e avaliação no Brasil. É fundamental a conceituação dos tipos de avaliações e suas transformações ao longo do tempo, além de compreender como as tecnologias são introduzidas e afetam na educação.

### 2.1 PAPEL DAS AVALIAÇÕES NA EDUCAÇÃO

No contexto escolar atual, as avaliações são instrumentos fundamentais na aferição de aprendizagem e retenção de conhecimento. De acordo com Black e William (1998), o núcleo da atividade de avaliação se baseia na percepção dos objetivos de aprendizagem do aluno e nas ações tomadas para alcançá-los. Os autores revelam que o uso de provas por professores em uma frequência apropriada contribui para o aumento da performance dos alunos, além disso, a qualidade do feedback se mostra essencial em dirigir os estudantes em busca de seus objetivos de aprendizagem, os quais podem ser fornecidos pela própria avaliação.

Assim, as avaliações, quando pensadas e adaptadas ao contexto da sala de aula, são ferramentas poderosas na evidenciação de avanços e dificuldades dos discentes em relação aos objetivos educacionais estabelecidos. Segundo Libâneo (2013), “é importante ressaltar que a avaliação é uma reflexão sobre o nível de qualidade do trabalho escolar, tanto do professor, como dos alunos”.

Porém, as avaliações muitas vezes acabam tomando o seu papel tradicional em sala de aula, ou seja, atuam como ferramenta de controle, quantificação, comparação e classificação. Segundo Freitas, Costa e Miranda (2014, p. 88) “Nessa visão tradicional o professor restringe a avaliação a um instrumento de controle que é utilizado para medir os conteúdos memorizados pelo aluno”. Assim, a prova perde seu papel como mediadora e estimuladora de aprendizagem, passando para um dispositivo de governo, e em casos extremos instaurando medo e controle social em educandos.

Todavia, ainda que educadores fujam do papel punitivo e controlador das avaliações, o potencial como mediadora de aprendizagem das provas dificilmente é explorado em sua totalidade. Fatores como sobrecarga de trabalho, jornadas duplas

e triplas, entre outros, dificultam a criação de provas voltadas ao contexto específico de cada situação. A pesquisa de Viegas (2022) fornece um panorama sobre a situação de trabalho dos professores, contando com entrevistas e discussões envolvendo 204 docentes de 18 municípios e abrangendo diversas áreas da educação, na região do Vale do Rio Pardo, Rio Grande do Sul. Os dados revelam que 83,3% dos professores têm carga horária de 40 horas semanais ou mais e 43,7% lecionam em mais de uma instituição. Entretanto, de acordo com o autor, este último dado é acentuadamente maior que de outras pesquisas com maior escopo regional, como Ministério da Educação e Instituto Nacional de Estudos e Pesquisas (MEC/INEP) e demais autores.

### 2.1.1 Impacto das Avaliações no Ensino

Naturalmente, devido à importância das avaliações no ensino, a atmosfera à volta do processo avaliativo é composta por grande ansiedade, e seus resultados trazem uma gama de emoções, desde a felicidade até a frustração. Como apontado por Gatti (2003), muitos professores criam avaliações de extrema dificuldade e consideram-nas bem-sucedidas se poucos alunos obtiveram boas notas, assim como o oposto é verdade, onde professores criam provas tão fáceis que alunos não precisam sequer se empenhar em sua realização. No primeiro caso, gera-se ansiedade, frustração e injustiça, enquanto no segundo, gera-se falta de disposição e indiferença, mas em ambos os casos, o processo de aprendizagem é afetado negativamente. “Sentimentos negativos em relação às provas vão sendo desenvolvidos ao longo dos anos de escolarização e, indubitavelmente, se quiser mudá-los leva-se algum tempo” (Gatti, 2003, p.7).

Contudo, a avaliação não impacta apenas os alunos que a realizam: ela também avalia a prática de ensino do próprio avaliador, como explicitado por Vieira (2022). Baseado na mesma obra de Gatti e utilizando a visão moderna de avaliação tratada anteriormente, o avaliador é o responsável direto pelo processo avaliativo, ou seja, o docente que ensina é o mesmo que avalia, e assim, mesmo que incidentalmente, acaba por gerar uma reflexão de seu respectivo trabalho. Dessa forma, é evidenciado o que procurou se ensinar e a forma empregada para isso, revelando o processo de docência e, de forma mais significativa, a qualidade deste.

Assim, quando a prova cumpre seu papel reflexivo, gera um *feedback* essencial tanto para alunos, quanto para professores, permitindo que estes considerem e reorientem suas ações de ensino e aprendizagem.

### 2.1.2 Tipos de avaliações

A obra de Freitas, Costa e Miranda (2014) discute as diferentes abordagens avaliativas, condensando-as em três principais tipos de avaliações, sendo elas:

1. Diagnóstica, avaliando pré-requisitos como conhecimento pré-estabelecido
2. Formativa, avaliando se objetivos propostos foram alcançados
3. Somativa, classificar o aluno e quantificar seus resultados

Nota-se que esses tipos de avaliação têm suas funções definidas conforme os objetivos estabelecidos e podem ser usadas em diferentes momentos do ciclo de vida do processo avaliativo.

A visão moderna da avaliação como mediadora da aprendizagem foca no uso de avaliações do tipo formativa, as quais são fundamentais no processo de aprendizado contínuo, e ainda fornecem uma ponte para o alcance dos objetivos de ensino dos alunos, professores, e até mesmo das instituições.

Ao longo desta seção, explorou-se o papel fundamental das avaliações no contexto educacional, desde a sua importância na medição da aprendizagem até o seu impacto na prática instrucional dos professores. Demonstrou-se como as avaliações podem ser ferramentas poderosas quando adequadamente adaptadas ao ambiente de sala de aula, permitindo não apenas a medição do conhecimento do aluno, mas também a identificação de áreas de crescimento e o estímulo à reflexão do professor. No entanto, também se reconheceu os desafios que os educadores enfrentam, que vão desde a tendência de usar as avaliações como meio de controle até as dificuldades práticas associadas à criação de avaliações formativas e específicas para o contexto. Apesar desses obstáculos, é evidente que as avaliações são essenciais na promoção de uma aprendizagem significativa e no aprimoramento contínuo da prática educacional. Ao reconhecer e enfrentar prontamente esses desafios, pode-se otimizar o potencial das avaliações como ferramentas de apoio instrucional.

## 2.2 TRANSFORMAÇÃO DIGITAL NA EDUCAÇÃO

Para Stolternam e Fors (2004, p. 689) “A transformação digital pode ser entendida como as mudanças que a tecnologia digital causa ou influencia em todos os aspectos da vida humana”. A transformação digital na sociedade contemporânea engloba esferas como a tecnológica, organizacional e social, e ainda que facilitado pela primeira não é limitado a ela (Pacheco; Santos; Wahrhaftig, 2020). Porém, ao pensar neste fenômeno, constata-se que a esfera tecnológica é a mais popular, se não a mais importante na transformação digital.

Para Schwab (2016), um dos fundadores do Fundo Monetário Internacional, os tempos atuais constituem a Quarta Revolução Industrial, a qual começou em meados de 2010. Uma das características dessa revolução, também chamada de Indústria 4.0, é a velocidade com que se tem inovações tecnológicas, impulsionadas pela convergência de tecnologias digitais, físicas e biológicas. Conseqüentemente, a educação procura se adaptar a esse ambiente tecnológico, desfrutando das novas tecnologias e criando sua própria revolução, a Educação 4.0, marcada pela inovação, informatização e uso de tecnologias de informação e comunicação (TIC). “Na atualidade, a tecnologia está cada vez mais presente em todas as áreas de nossas vidas e tem sido utilizada em salas de aula tanto por alunos quanto por professores.” (Siqueira; Bittencourt; Novaes; Avelar, 2021, p. 1264).

Contudo, existem diversas formas de entender e conceituar a tecnologia. De acordo com Chaves (2007), ela é qualquer ferramenta, método ou processo criado pelo homem para facilitar seu trabalho, desde locomoção, comunicação, ou apenas para tornar sua vida mais agradável. Entretanto, as tecnologias mais relevantes para a educação são aquelas que ampliam os poderes sensoriais do homem, a comunicação e o poder mental, e de forma interessante, os sistemas computacionais centralizam essas características em apenas um produto.

Assim, o computador toma uma posição como ferramenta didático-pedagógica, o qual pode melhorar a qualidade do ensino e aprendizagem, de forma que o próprio aluno se torna o construtor de conhecimento e o professor se torna o mediador entre o aluno, o computador e o conhecimento (Peixoto; Araújo, 2012).

Um exemplo prático da incorporação de tecnologias computacionais na educação é o Moodle, um ambiente virtual de aprendizado (AVA) oficialmente adotado

pela Universidade Federal do Paraná (UFPR, 2024). Como exposto pela Coordenadoria de Integração de Políticas de Educação a Distância (CIPEAD), a UFPR Virtual, uma versão modificada do Moodle, foi “pensada e desenvolvida para funcionar como um campus virtual da nossa universidade” (CIPEAD, 2024, p. 1). Ela permite criação de cursos, interação entre professores e alunos, disponibilização de conteúdo, realização de atividades e avaliações e mais, ou seja, a UFPR Virtual fornece plataforma para educação à distância (EAD), aulas híbridas e apoio ao ensino presencial.

Portanto, conclui-se que a transformação digital na educação é uma realidade incontestável, impulsionada pela convergência de tecnologias digitais que permeiam todos os aspectos da vida humana. Dessa forma, surge a Educação 4.0 como resposta a esse cenário, marcando uma era de inovação, informatização e uso de tecnologias de informação e comunicação. Exemplos concretos, como a adoção do Moodle pela Universidade Federal do Paraná, ilustram como as instituições de ensino estão se adaptando e aproveitando as novas tecnologias para aprimorar a qualidade do ensino e da aprendizagem, preparando os estudantes para os desafios e oportunidades do mundo conectado digitalmente e usando a transformação digital a seu favor.

### 2.3 AVALIAÇÃO EDUCACIONAL E TECNOLOGIA

O uso de tecnologias de aplicação de provas, especialmente exames eletrônicos, apresenta um grande potencial no aprimoramento e auxílio de ensino e aprendizagem. Como explicitado por Frankl, Napetschnig e Schartner (2017, p. 200) “Exames eletrônicos têm o potencial de melhorar a didática, eficiência, objetividade, flexibilidade, acessibilidade e até mesmo sustentabilidade comparados a exames no papel.”

Essas vantagens refletem uma série de benefícios associados ao uso de exames eletrônicos, os quais serão explorados em detalhes:

1. Didática: fornecem recursos interativos como questões de múltipla escolha e de texto aberto com feedback imediato, que é essencial na melhora de performance dos alunos e promove um aprendizado autorregulado (Nicol; Macfarlane-Dick, 2007).

2. Eficiência: aplicação rápida de provas em grupos maiores de alunos, escalonamento facilitado e possibilidade de administração remota, além de diminuir a carga de trabalho para docentes.
3. Objetividade: critérios de avaliação mais explícitos e claros, além de correção automática (Williamson; Breyer, 2019). Também dificulta a cola por meio de embaralhamento de questões e alternativas.
4. Flexibilidade: agendamento, formato, dispositivos e locais de realização.
5. Acessibilidade: para alunos com necessidades especiais, há a possibilidade de adaptação de fontes, cores e formatos, narração, lupas etc.
6. Sustentabilidade: diminuição do consumo de papel e materiais envolvidos na criação de provas escritas. Estudantes podem usar seus próprios dispositivos, diminuindo o uso de recursos por parte das instituições.

Todavia, esses benefícios não são exclusivos a exames eletrônicos. Como será discutido na Seção 2.5, tecnologias para geração de provas escritas também estão evoluindo para se beneficiar de algumas vantagens supracitadas, especialmente em termos de didática, eficiência e objetividade.

Ademais, outra grande vantagem no uso de tecnologias de avaliação é o fornecimento de dados e estatísticas sobre o desempenho dos alunos. O uso dessas tecnologias permite abordagens analíticas para fornecimento de *feedback* de qualidade tanto para discentes quanto para educadores (Heil; Ifenthaler, 2023).

Logo, as tecnologias de avaliação fornecem excelentes ferramentas para o aprimoramento da educação, oferecendo benefícios como a melhoria da didática, eficiência, objetividade e acessibilidade. Tanto os exames eletrônicos quanto as tecnologias para a geração de provas escritas estão em constante evolução para atender às demandas modernas de ensino e aprendizagem. Dessa forma, somado a disponibilidade de dados detalhados sobre o desempenho dos alunos, acredita-se que essas tecnologias têm o potencial de impulsionar ainda mais o sucesso educacional.

## 2.4 FUNDAMENTAÇÃO DA TECNOLOGIA

Exposto o embasamento teórico da fundamentação de negócio do sistema proposto, é essencial compreender os processos subjacentes que possibilitaram a

solução proposta. O objetivo desta seção é fundamentar a engenharia de software assim como seus processos e aplicações.

#### 2.4.1 Engenharia de software

A Engenharia de Software é uma disciplina da computação voltada à especificação, desenvolvimento e manutenção de software, envolvendo todas as áreas de sua concepção. Para Sommerville (2011), essa engenharia “É uma disciplina de engenharia que se preocupa com todos os aspectos de produção de softwares.” Segundo Pressman e Maxim (2016), ela é uma abordagem sistemática e disciplinada para o desenvolvimento de software.

Como explicitado no SWEBOK (*Guide to the Software Engineering Body of Knowledge*), um processo de engenharia é um conjunto de atividades que transformam entradas em saídas utilizando recursos para alcançar esta transformação (SWEBOK, 2014). Na Engenharia de Software, os processos de engenharia, também chamados processos de software, dizem respeito à especificação de atividades de trabalho relacionadas ao desenvolvimento, manutenção e operação de software, como requisitos, design, construção, e demais processos.

Assim, esses processos são especificados para facilitar o entendimento humano, comunicação e coordenação, auxiliar no gerenciamento de projetos, medir e melhorar a qualidade de produtos e processos e mais, ou seja, a Engenharia de Software e seus processos são essenciais na gestão de um projeto de software, garantindo a qualidade e alcance dos objetivos estabelecidos de forma eficiente e bem planejada (SWEBOK, 2014).

Para Sommerville (2011), existem diversos processos de software, porém algumas atividades fundamentais são comuns a todos eles:

- Especificação: define a funcionalidade do software e as restrições sobre sua operação;
- Projeto e implementação: o software que atenda a especificação deve ser produzido;
- Validação: o software deve ser validado para garantir que ele faça o que o cliente deseja;

- Evolução: o software deve evoluir para atender aos novos requisitos que naturalmente surgirão.

Apesar da aparente simplicidade da categorização, os processos de software na prática são complexos e, como todos os processos intelectuais e criativos, são dependentes das decisões e julgamentos individuais dos envolvidos. Além disso, não existe um processo universalmente ideal, logo, é natural que as organizações criem e adaptem seus processos e atividades de acordo com seu contexto. Dessa forma, um processo de software pode ser dirigido a plano, no qual as atividades são planejadas previamente e são bem definidas, ou processos ágeis, onde o planejamento é progressivo e mais flexível em relação a mudança das necessidades do cliente (Sommerville, 2011).

#### 2.4.2 Desenvolvimento ágil

A agilidade no contexto da Engenharia de Software é estimulada principalmente pelas mudanças recorrentes em projetos. Porém, mais que isso, ela abrange os princípios filosóficos do Manifesto Ágil apresentado anteriormente neste trabalho. Esses princípios enfatizam a estruturação e ações de equipes para facilitar a comunicação, entrega rápida de software de valor, inclui o cliente como parte da equipe de desenvolvimento e reconhecem as limitações do planejamento, sugerindo um plano mais flexível (Pressman; Maxim, 2016).

Os projetos de software atuais são caracterizados pela falta de certeza sobre quais requisitos se manterão estáveis e quais mudarão, intercalação do projeto e da construção e falta de previsibilidade da análise, projeto, construção e teste, e, portanto, nota-se que a imprevisibilidade permeia a maioria dos projetos. Logo, para que o processo ágil lide com a imprevisibilidade adequadamente é necessário que ele se adapte de forma contínua e progressiva, ou seja, de modo incremental. Assim, faz-se necessário uma abordagem iterativa baseada em curtos períodos de entrega, a qual capacita o cliente a avaliar os produtos de cada incremento, fornecer feedback para estimular as adaptações e superar a imprevisibilidade (Pressman; Maxim, 2016).

Dessa forma, compartilhando das características comuns supracitadas e embasadas no Manifesto Ágil, surgem diversos métodos ágeis de desenvolvimento

como o Extreme Programming, Scrum, Crystal etc. que buscam alcançar o objetivo da agilidade, porém com processos diversificados.

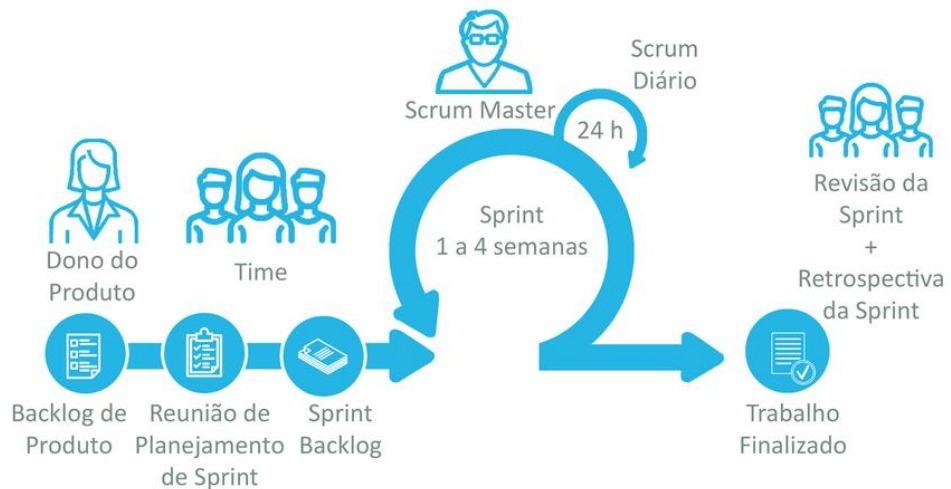
### 2.4.3 Scrum

O Scrum, criado por Ken Schwaber e Jeff Sutherland no início dos anos 1990, é uma metodologia empregada para organizar e gerenciar projetos, combinando os valores e princípios do manifesto ágil com os elementos e fluxo definidos no *framework* Scrum. “Scrum é um *framework* leve que ajuda pessoas, times e organizações a gerar valor por meio de soluções adaptativas para problemas complexos” (Schwaber; Sutherland, 2020, p. 4).

Em resumo, o Scrum exige a presença de um Scrum Master para criar um ambiente no qual:

- *Product Owner* organiza o trabalho para resolver um problema complexo, criando um *Product Backlog*;
- O *Scrum Team* transforma uma seleção desse trabalho em um incremento de valor durante uma Sprint;
- O *Scrum Team* e seus *stakeholders* inspecionam os resultados e fazem ajustes para a próxima Sprint;
- Repete-se esse ciclo de trabalho, ilustrado pela FIGURA 1.

FIGURA 1 - CICLO DO SCRUM



Fonte: Nésio; Freitas; Freitas (2020)

O *Scrum Team* é um grupo pequeno de pessoas que trabalham juntas para criar um Incremento utilizável. Esse time é composto por:

- Desenvolvedores: Eles são responsáveis por criar qualquer aspecto do produto;
- *Product Owner*: responsável por criar e gerenciar os itens no *Product Backlog* e definir a meta do produto;
- *Scrum Master*: O *Scrum Master* é responsável por estabelecer e manter o processo Scrum;

Os eventos do Scrum são:

- *Sprint*, eventos de duração fixa de um mês ou menos. “Todo o trabalho necessário para atingir a meta do Produto, incluindo *Sprint Planning*, *Daily Scrums*, *Sprint Review* e *Sprint Retrospective*, acontece dentro de *Sprints*” (Schwaber; Sutherland, 2020, p. 8);
- *Sprint Planning*, usado para definir o trabalho a ser realizado naquela *Sprint* por meio de definições do valor da *Sprint*, o que será feito na *Sprint* e como o trabalho será realizado;
- *Daily Scrum*, reunião diária para averiguar o progresso da *Sprint* e adaptar o *Sprint Backlog* conforme necessidade;
- *Sprint Review*, reunião no final da *Sprint* para inspecionar o resultado da *Sprint* e determinar futuras adaptações;

- *Sprint Retrospective*, reunião no final da Sprint para estabelecer maneiras de melhorar a qualidade e eficácia;

Os artefatos do Scrum representam valor ou trabalho, sendo eles o *Product Backlog*, que é lista do que é necessário para melhorar o produto; *Sprint Backlog*, sendo composto pela Meta da Sprint, itens do *Product Backlog* selecionados e plano de ação para entrega; e os Incrementos, partes de software potencialmente entregável.

#### 2.4.4 Arquitetura de Microserviços

A arquitetura de microserviços é um padrão arquitetural que estrutura aplicações em serviços, cuja função é tipicamente delimitada a uma capacidade de negócio. Cada microserviço é independente e levemente acoplado a outros, realizando sua comunicação por processos leves como HTTP (*Hypertext Transfer Protocol*), AMQP (*Advanced Message Queuing Protocol*) etc. Essa arquitetura surge em contraste ao monolito, onde todas as capacidades de negócio se encontram em apenas uma aplicação (Fowler, 2002).

O uso desse padrão arquitetural traz diversas implicações como independência de tecnologias, escalabilidade granular e flexibilidade de infraestrutura, e por outro lado traz desafios como coesão arquitetural, elevada complexidade operacional e de rede além de coordenação de equipes (XP Inc., 2020).

#### 2.4.5 Padrões de Projeto

Padrões de projeto são soluções típicas e reutilizáveis para problemas comuns de software, sendo aplicáveis em praticamente qualquer linguagem de programação e trazendo uma linguagem comum e consistente para as equipes de desenvolvimento.

O livro "*Design Patterns: Elements of Reusable Object-Oriented Software*", publicado em 1994 pelos autores conhecidos como *Gang of Four* (Erich Gamma, Richard Helm, Ralph Johnson e John Vlissides), é considerado a principal referência clássica sobre padrões de projeto. De acordo com esse livro, os padrões são organizados em três categorias principais: criacionais, estruturais e comportamentais.

Essa divisão é feita com base na natureza do problema que cada padrão busca resolver. Os padrões definidos por esses autores dizem respeito a softwares orientados a objetos.

Seguindo o conceito dos padrões de projetos, a arquitetura de microsserviços tem seus próprios padrões de acordo com as necessidades deste modelo. O site [Microservices.io](https://microservices.io), mantido por Chris Richardson, é uma das principais referências práticas sobre padrões e arquiteturas baseadas em microsserviços. De acordo com o autor, os padrões podem ter diversas classificações, englobando limites de serviço, comunicação, testagem, implantação e mais.

#### 2.4.5.1 Database per service

Em uma aplicação onde cada serviço precisa armazenar seus próprios dados, o padrão de projeto Database per Service oferece uma solução que garante a independência de cada serviço em relação ao banco de dados. Esse padrão propõe que cada serviço tenha seu próprio banco de dados privado, que não é acessado diretamente por outros serviços. Em vez disso, o acesso aos dados é feito exclusivamente por meio da API do próprio serviço, o que reforça o encapsulamento e a autonomia de cada componente do sistema (Richardson, 2025).

#### 2.4.5.2 API Gateway

Na arquitetura de microsserviços, um cliente pode precisar de dados que estão divididos em diferentes serviços, cada qual com sua localização e função. A fim de solucionar este problema, o padrão API Gateway traz um ponto de entrada singular e comum para todos os clientes, que age como um roteador de requisições ou divide uma requisição em várias conforme a necessidade. De forma geral, este padrão abstrai a complexidade de coordenar requisições a diferentes serviços por parte do cliente (Richardson, 2025).

#### 2.4.5.3 SAGA

Como consequência do uso do padrão Database per service, transações distribuídas representam um desafio quando uma única operação de negócio requer a atualização de dados em múltiplos serviços. Para lidar com essa complexidade, o

padrão de Saga Orquestrada é utilizado, onde um orquestrador central coordena a execução sequencial das etapas da transação, acionando cada serviço envolvido e definindo as ações compensatórias em caso de falhas. O uso desse padrão facilita o controle do fluxo, garantindo consistência eventual sem depender de bloqueios ou transações distribuídas tradicionais. O orquestrador pode ser implementado como um serviço dedicado que gerencia o estado e o progresso da SAGA, aliviando os serviços individuais dessa responsabilidade. (Richardson, 2025).

#### 2.4.5.4 Circuit Breaker

Quando a lógica de negócio é separada em microsserviços, há a necessidade de comunicação entre eles em certas funcionalidades. Todavia existe a possibilidade que o serviço requisitado está indisponível ou inutilizável, consumindo recursos do requisitante, podendo causar falhas em cascata em demais áreas da aplicação.

Para superar este problema, o padrão Circuit Breaker atua semelhante a um disjuntor elétrico, que é ativado caso um número limite de falhas ou tempo seja excedido, fazendo com que todas as requisições para tal serviço falhem imediatamente por um período determinado (Richardson, 2025).

## 2.5 SOFTWARES SEMELHANTES

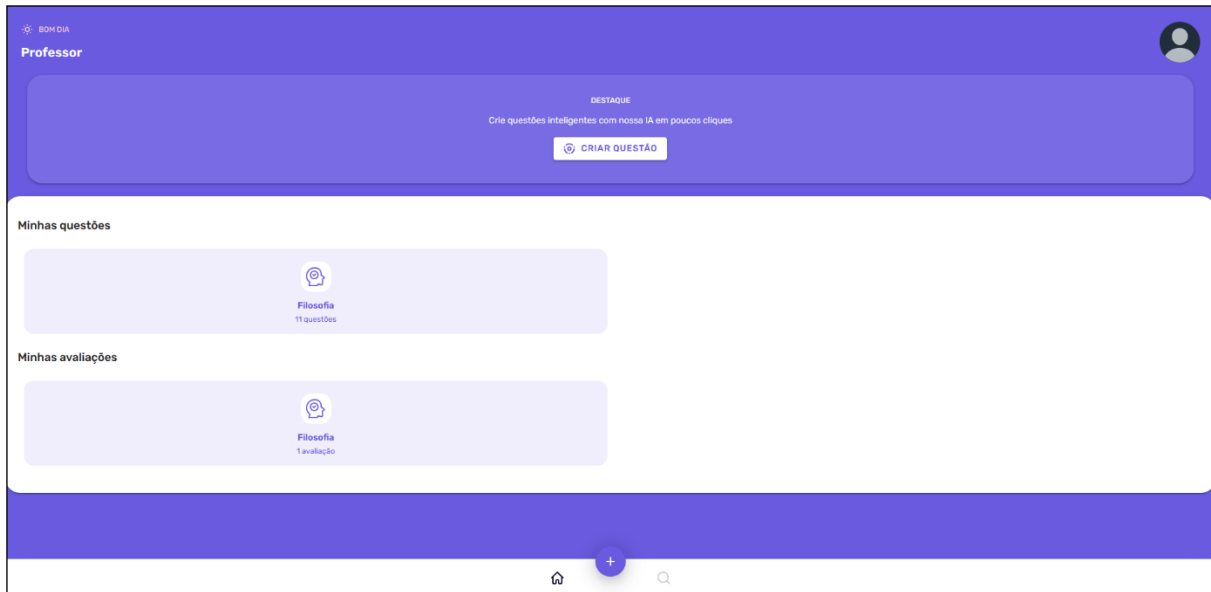
Após entender a importância da avaliação no ensino e como a tecnologia está transformando a educação, pode-se inferir que o uso de softwares para facilitar a criação de provas e automatizar sua correção tem o potencial de otimizar significativamente o processo de avaliação. Essas aplicações podem agilizar a elaboração de questões, oferecer uma variedade de formatos de perguntas, permitir personalização de avaliações de acordo com as necessidades específicas dos alunos e proporcionar uma correção rápida e precisa.

Assim, foram buscadas tecnologias semelhantes a esta proposta neste trabalho, e foram encontradas quatro aplicações que apresentam similaridades suficientes para serem comparadas. Todavia, duas dessas tecnologias, Minha Prova e Fábrica de Provas, são soluções corporativas que exigem uma licença anual para sua utilização, sem período de teste grátis ou exemplos de utilização, e assim não puderam ser testadas.

### 2.5.1 Pruvo AI

A primeira aplicação de criação, correção e gerenciamento de provas encontrada foi a Pruvo AI (PRUVO AI, 2025). Esse sistema oferece duas principais funcionalidades: criar questões e criar avaliação, conforme mostra a FIGURA 2.

FIGURA 2 - TELA DE CADASTRO PRUVO IA



FONTE: O autor (2025).

A criação de questões, como mostrado na FIGURA 3, oferece diferentes formatos de perguntas, etapas de ensino, começando no sexto ano do ensino fundamental até o terceiro ano do ensino médio, diversas disciplinas, assuntos e sugestões de assunto, dificuldade e a parte mais interessante: criação da questão por Inteligência Artificial (IA) utilizando os parâmetros definidos.

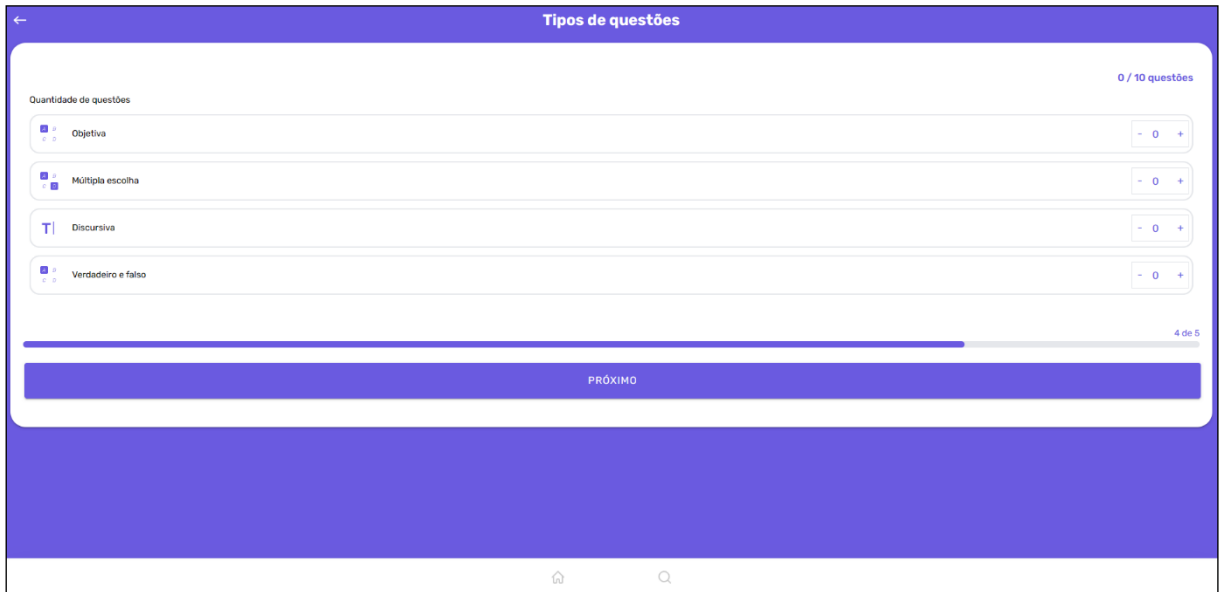
FIGURA 3 – TELA DE CRIAR QUESTÃO PRUVO AI

FONTE: O autor (2025).

O sistema também oferece a possibilidade da criação manual da questão, com formatação de texto e criação de alternativas.

A criação de avaliação do Pruvo AI, pode ser feita de duas formas: Minhas questões ou IA. Na primeira forma, o site irá pedir os parâmetros de questão da mesma forma na função de criação de questão, e então mostrará as perguntas com aquelas características. A segunda forma possibilita a criação de avaliação com inteligência artificial, onde o usuário define os parâmetros de etapas de ensino e disciplina, e então a aplicação permite a escolha do número de perguntas de cada formato, exemplificado na FIGURA 4.

FIGURA 4 – TELA DE TIPO DE QUESTÕES PRUVO AI



FONTE: O autor (2025).

Então, para cada pergunta, o usuário define o assunto e a dificuldade, e após a parametrização de todas as questões o sistema gera automaticamente a avaliação, suas respostas corretas e comentários de correção. As avaliações criadas são mostradas como na FIGURA 5.

FIGURA 5 – TELA DE AVALIAÇÃO PRUVO AI



FONTE: O autor (2025).

Ademais, o sistema oferece a possibilidade de exportação para extensão de documento .docx ou para PDF, assim como a exportação do gabarito para preenchimento em PDF. A correção é automática por meio de fotografia do gabarito preenchido.

Pontos positivos:

- Criação de avaliação facilitada por IA
- Plano gratuito com todos os recursos do sistema, ainda que limitados (exemplo: apenas uma exportação para PDF por mês)
- Interfaces simples

Pontos negativos:

- Escolher questão própria para a avaliação pode ser complicado, pois é necessário lembrar de todos os parâmetros
- Parâmetro de dificuldade não influencia de forma relevante a questão
- Criação de questão manual não permite inserção de imagens e apresentam poucas opções de formatação de texto

### 2.5.2 Gradepen

A segunda aplicação encontrada foi o Gradepen (GRADEPEN, 2025). É um aplicativo de criação e correção automática de avaliações por meio de um banco de questões público e pessoal. É utilizado por um aplicativo de correção em *smartphones* e *tablets* que utilizam IOS ou Android e por meio do website tanto em computadores quanto smartphones.

Algumas das funcionalidades são:

1. Criação de avaliações com questões discursivas e objetivas;
2. Correção de avaliação por meio de um QR Code em conjunto com uma senha de acesso;
3. Formatação das avaliações e criação de cabeçalhos;
4. Geração de um relatório das notas;
5. Compartilhamento de Questões;
6. Conferência de gabaritos anteriores;
7. Compra e gerenciamento dos créditos de correção;
8. Pesquisa pública de gabaritos.

O público-alvo do Gradepen são usuários em geral que desejam criar avaliações personalizadas como professores, educadores e alunos que necessitem visualizar sua nota por meio dos gabaritos publicados no site.

Principais interfaces:

- Tela inicial: apresentação do site e suas características relevantes; (FIGURA 6);
- Tela de avaliação: o usuário pode criar uma avaliação do zero ou aproveitar avaliações anteriores para criar um documento novo. O usuário pode selecionar questões de acordo com alguns atributos como disciplinas, assuntos, autores, tipos de questões (discursivas ou múltipla escolha) (FIGURA 7);
- Tela de gabarito: o usuário pode acessar todos os gabaritos de suas avaliações criadas;
- Tela de créditos: o usuário utiliza e gerencia créditos para obter certas funcionalidades do sistema como correções automáticas e privatização de questões (FIGURA 8);
- Tela de compra: local onde o usuário pode comprar créditos;
- Tela de alteração de dados cadastrais: onde o usuário pode atualizar dados como nome, sobrenome, instituição, e-mail e senha;
- Tela de descrição do site: Contém dados sobre principais funcionalidades e respostas as perguntas mais comuns (FIGURA 9);

O público-alvo do Gradepen são usuários em geral que desejam criar avaliações personalizadas como professores, educadores e alunos que necessitem visualizar sua nota por meio dos gabaritos publicados no site.

Pontos positivos:

- Gabarito único para cada aluno;
- Banco de questões públicas e privadas;
- Formatação da avaliação.

Pontos negativos:

- Quando se tem muitas questões criadas é difícil achá-las na criação de avaliação;
- Algumas interfaces apresentam muita informação e funcionalidades, tornando-as complexas e de difícil navegação;
- Para alguns ícones não há palavras explicativas e o usuário não consegue deduzir sua função.

FIGURA 6 – TELA INICIAL GRADEPEN



FONTE: O autor (2025).

FIGURA 7 – TELA DE AVALIAÇÕES GRADEPEN



FONTE: O autor (2025).

FIGURA 8 – TELA DE GERENCIAMENTO DE CRÉDITOS GRADEPEN



FONTE: O autor (2025).

FIGURA 9 – TELA DE DESCRIÇÃO DO WEBSITE GRADEPEN



FONTE: O autor (2025).

### 2.5.3 Quadro Comparativo

Após a análise de tecnologias semelhantes à proposta, criou-se um quadro comparativo de funcionalidades e demais características destas. Os resultados estão descritos no QUADRO 1.

QUADRO 1 – COMPARAÇÃO ENTRE TECNOLOGIAS SEMELHANTES

<b>Funcionalidades</b>	<b>Questionarium</b>	<b>Pruvo AI</b>	<b>Gradenpen</b>
Software gratuito	X	X	X
Créditos de uso ilimitados	X		
Criação de questões	X	X	X
Banco de questões privadas	X	X	X
Banco de questões públicas	X		X
Criação de avaliações	X	X	X
Correção de gabarito por imagem	X	X	X
Exportação para PDF	X	X	X
Geração de questões por IA	X	X	

FONTE: O autor (2025).

## 2.6 CONSIDERAÇÕES

Neste capítulo, foram expostos os principais conceitos que sustentam o desenvolvimento do software proposto, evidenciando desde a fundamentação do negócio, com ênfase nas conexões entre educação, avaliação e tecnologia, até a fundamentação das tecnologias envolvidas, onde apresentam-se as linguagens de programação, *frameworks*, metodologias ágeis e outras tecnologias relevantes. Além disso, foi realizado um estudo comparativo de tecnologias semelhantes no mercado, destacando suas funcionalidades, vantagens e desvantagens. Dessa forma, o próximo capítulo se dedicará ao detalhamento da metodologia utilizada para o desenvolvimento do software, revelando as etapas do processo de desenvolvimento, cronograma, tecnologias aplicadas e demais estratégias.

### 3 MATERIAIS E MÉTODOS

Após a apresentação dos conceitos, motivações e problemas que englobam este trabalho, o próximo passo é o processo de desenvolvimento do software proposto. Este capítulo tem por objetivo evidenciar o modelo de processo de Engenharia de Software utilizado e a partir deste as etapas realizadas, além dos cronogramas de confecção deste documento e do desenvolvimento da solução e as tecnologias empregadas.

#### 3.1 MÉTODO

Durante a execução deste projeto, foram realizadas adaptações no *framework* Scrum, considerando as necessidades específicas e a disponibilidade da equipe. As modificações foram as seguintes:

- *Scrum Team*: O *Scrum Team*, composto por desenvolvedores, assumiu papéis multifuncionais. Não houve um *Scrum Master* dedicado, em vez disso, este papel foi rotativo a cada Sprint, com diferentes membros da equipe assumindo essa responsabilidade.
- *Product Owner*: O papel de *Product Owner* também foi compartilhado entre os membros da equipe e o orientador, criando o *Product Backlog* e definindo a meta do produto.
- Eventos da Sprint:
  - *Sprint Planning*: Reuniões de planejamento no início de cada Sprint para definir o trabalho a ser realizado.
  - *Sprint Review*: No final de cada Sprint, inspecionaram-se os resultados e determinaram-se ajustes para a próxima iteração.
  - *Sprint Retrospective*: Realizou-se apenas uma retrospectiva no final da primeira fase do projeto para identificar maneiras de melhorar a qualidade e eficácia e refletir esses aprimoramentos na segunda fase do projeto.
- *Daily Scrum*: Optou-se por não realizar reuniões diárias devido à disponibilidade da equipe.

Essas adaptações permitiram a aplicação flexível e eficiente do Scrum, considerando as circunstâncias específicas do projeto. A Seção 3.3 do documento evidencia as Sprints realizadas e as planejadas para a segunda fase do trabalho, enquanto a Seção 3.5 detalha cada Sprint.

## 3.2 MODELAGEM

Visto como processo, o design ou modelagem de software é uma atividade pertencente ao ciclo de vida de Engenharia de Software que consiste na análise dos requisitos de forma a produzir uma descrição da estrutura interna e comportamento externo do software. De acordo com o SWEBOK (2014, p.2) “o design de software é definido tanto como o processo de definição de arquitetura, componentes, interfaces e outras características de um sistema ou componente e o resultado deste processo.”

A importância da modelagem, como evidenciado por Pressman e Maxim (2016), vem do fornecimento de um roteiro para atingir um objetivo com sucesso a todos os envolvidos na criação do software, provendo detalhamento necessário para seguir o caminho e auxiliando a compreensão dos conceitos e princípios que devem ser entendidos.

Para Sommerville (2011), os produtos da modelagem devem ser abstrações do software considerado, além disso, apresentam várias funções durante o processo, como auxílio na extração de requisitos, descrição para os responsáveis pela implementação e documentação da estrutura e operação do software. Ademais, a modelagem geralmente representa o software por meio de notação gráfica, no caso deste trabalho por meio da UML – *Unified Modeling Language*. A UML é uma linguagem que permite a criação de diagramas para representar sistemas complexos.

A UML foi escolhida pois se tornou uma linguagem de modelagem padrão na orientação a objetos, além dos diversos tipos de diagramas que podem ser representados por ela, visto que o design de software engloba o design arquitetural, que descreve como o este é organizado em componentes e o design detalhado, que descreve o comportamento desejado desses componentes (SWEBOK, 2014).

A UML foi utilizada na criação de Diagramas de Casos de Uso, Diagrama de Classes e Diagrama de Sequência, que são encontrados nos Apêndices A, C e D respectivamente, a fim de representar diversos aspectos do sistema proposto, como interações, comportamentos, estruturas e arquiteturas.

### 3.2.1 Modelagem do Sistema

O sistema foi desenvolvido para possibilitar a criação, aplicação e o registro de avaliações de forma flexível e ao mesmo tempo garantindo a integridade histórica das aplicações mesmo diante de eventuais alterações em questões ou modelos de avaliações.

#### 3.2.1.1 Entidades e Relacionamentos

A modelagem do sistema contempla as seguintes entidades principais:

- Question (Questão): Representa as perguntas criadas pelo usuário, ou geradas pela automaticamente por IA (apêndice F). As questões podem ser discursivas (sem alternativas) ou de múltipla escolha, podendo conter de duas a cinco alternativas associadas (entidade Alternative).
- Alternative (Alternativa): Relaciona-se com a entidade Question e representa as possíveis respostas para questões de múltipla escolha.

O usuário cria modelos de avaliação, por meio da entidade AssessmentModel, os quais funcionam como um repositório estruturado de questões para serem utilizadas em avaliações futuras.

- AssessmentModel (Modelo de Avaliação): Estrutura composta por um conjunto de questões (e suas alternativas, quando aplicável), servindo como base para aplicações de avaliações.

Quando um modelo de avaliação é aplicado, gera-se uma nova entidade AppliedAssessment, responsável por registrar cada aplicação daquele modelo. Para cada aplicação, é criada uma instância independente da avaliação aplicada (entidade RecordAssessment), que representa o registro individual de cada avaliação feita.

- AppliedAssessment (Avaliação Aplicada): Refere-se a cada aplicação de um modelo de avaliação.
- RecordAssessment (Registro de Avaliação): Registra cada ocorrência individual de aplicação de avaliação, permitindo o acompanhamento dos resultados e respostas específicas.

### 3.2.1.2 Clonagem e Imutabilidade das Avaliações e Questões

É fundamental a garantia da imutabilidade das avaliações aplicadas. Para isso, sempre que uma avaliação é aplicada, todas as questões e alternativas do modelo de avaliação são “clonadas” no momento da aplicação, originando as entidades QuestionSnapshot (questão foto) e AlternativeSnapshot (alternativa foto).

- QuestionSnapshot (Questão Foto): Cópia da questão original no momento da aplicação.
- AlternativeSnapshot (Alternativa Foto): Cópia da alternativa vinculada à questão snapshot.

Essas "fotos" das questões e alternativas são vinculadas às entidades AppliedAssessment e RecordAssessment. Essa modelagem é feita para garantir que eventuais alterações feitas posteriormente nas questões originais, alternativas ou modelos de avaliação não afetem as avaliações já aplicadas. Assim, o sistema assegura que o histórico das avaliações permaneça fiel ao seu estado no momento da aplicação.

### 3.2.2 Diagrama de Casos de Uso

A modelagem de caso de uso foi desenvolvida por Booch, Jacobson e Rumbaugh e foi apresentada no primeiro lançamento da UML em 1999. Como explicitado por Sommerville (2011, p. 86) “Um caso de uso pode ser tomado como um cenário simples que descreve o que o usuário espera de um sistema”. Este tipo de modelagem é usado para levantamento de requisitos e cada caso de uso representa uma tarefa a interação externa com um software. Tal interação é proveniente de atores que representam tanto humanos quanto sistemas e máquinas.

Para representar os casos de uso, é usado o Diagrama de Caso de Uso da UML, e assim como descrito por Booch, Rumbaugh e Jacobson (2005, p. 95):

Um diagrama de caso de uso mostra um conjunto de casos de uso e atores (um tipo especial de classe) e seus relacionamentos. Você aplica diagramas de caso de uso para ilustrar a visualização estática de caso de uso de um sistema. Diagramas de casos de uso são especialmente importantes em organizar e modelar os comportamentos de um sistema.

Para este trabalho, foi desenvolvido um Diagrama de Caso de Uso que representa o ator Usuário, além dos comportamentos modelados para o sistema a ser desenvolvido. O diagrama pode ser encontrado no Apêndice A.

Todavia, os diagramas de caso de uso, por design, carecem de detalhes da interação representada, e para isso podem ser usadas descrições textuais, detalhadas ou outros tipos de diagrama. Com esse objetivo, este trabalho utilizou de Histórias de Usuário, apresentados na Seção 3.2.2 e Diagramas de Sequência, apresentados na Seção 3.2.5.

### 3.2.3 Histórias de Usuário

Como citado anteriormente, os diagramas de caso de uso não contêm detalhamento necessário de uma interação e comportamento do sistema estudado. Logo, as histórias de usuário foram usadas para detalhar os casos de uso e demais características a serem implementadas no sistema a ser desenvolvido.

As histórias de usuário se encaixam muito bem no desenvolvimento ágil, visto que representam uma parte de uma funcionalidade e não uma extensa descrição do sistema, além de contar com uma estrutura flexível e informal e dessa forma podem ser usadas como artefatos do Scrum. As histórias criadas para este software seguem a estrutura SENDO (ator), QUERO (ação), PARA (objetivo), além de critérios de aceitação detalhados, desenhos de tela que serão citados na próxima seção e notas adicionais (Cohn, 2004). As histórias podem ser encontradas no Apêndice B.

### 3.2.4 Protótipos de Interface

Um protótipo é um tipo de demonstração que apresenta às partes interessadas que tipos de funcionalidade o sistema pode oferecer. É uma excelente forma de elicitar requisitos pois possibilita aos usuários finais experimentar uma versão representativa do sistema e a partir refinar suas ideias sobre os requisitos.

Para Sommerville e Sawyer (1997), uma das formas de prototipação é a prototipagem em papel, onde uma maquete das telas desenhadas em papel é criada e usada para experimentos do sistema estudado. Todavia, com o advento da tecnologia o uso de papéis de fato não é mais necessário, podendo substituí-los por

software de prototipação como o Figma, utilizado neste trabalho e apresentado na Seção 3.5, mas o conceito deste tipo de prototipagem permanece em alta.

Assim, foram criados diversos protótipos de interfaces para representar o sistema e as navegações entre telas. Esses protótipos foram integrados às histórias de usuário (Apêndice B) para esclarecer e facilitar o entendimento destas.

### 3.2.5 Diagrama de Classes

O diagrama de Classes é um tipo de diagrama da UML categorizado como estrutural, considerado o esqueleto ou planta de um sistema. Este tipo de diagrama é usado para representar o conjunto de classes, interfaces, colaborações e seus relacionamentos e é o diagrama mais comum na modelagem de sistemas orientados a objetos (Booch; Rumbaugh; Jacobson, 2005). Uma classe de objeto pode ser entendida como a definição geral de um objeto do sistema, que nas fases iniciais do processo de software tais objetos representam algo do mundo real (Sommerville, 2011). Além da estrutura do sistema estudado, o diagrama de classes pode ser usado como base para a construção de um modelo lógico de banco de dados.

Segundo Booch (2005, p.106), “você usa diagramas de classes para modelar a visualização do design estático de um sistema. Esta visualização primariamente apoia os requisitos funcionais de um sistema - os serviços que o sistema deve providenciar para seus usuários finais”.

O diagrama de classes deste trabalho se encontra no Apêndice C.

### 3.2.6 Diagrama de Sequência

Até este ponto, os diagramas mostrados e até mesmo protótipos oferecem uma visão estática do sistema, ainda que representando suas interações e comportamentos. Para ilustrar a visão dinâmica de um sistema é usado o diagrama de sequência, que mostra a interação de elementos do sistema. “Um diagrama de sequência enfatiza a ordenação cronológica de mensagens.” (Booch; Rumbaugh; Jacobson, 2005, p. 252).

Como explicitado por Sommerville (2011), o diagrama de sequência mostra a sequência de interações que ocorrem em um caso de uso. Os diagramas de sequência da UML são muito ricos em recursos para demonstrar as diversas

possibilidades de eventos durante a execução de um processo ou interação, tornando fácil o entendimento dos ciclos de vida de objetos, troca de mensagens e demais especificidades de um processo em particular do sistema. Os diagramas de sequência modelados para este trabalho se encontram no Apêndice D.

### 3.2.7 Diagrama Físico do Banco de Dados

O diagrama físico do banco de dados descreve a estrutura de armazenamento dos dados em nível de implementação, especificando com precisão como os dados são organizados em tabelas, colunas, tipos de dados, índices e restrições em um Sistema Gerenciador de Banco de Dados (SGBD). Esse tipo de diagrama é resultado direto da transformação dos modelos conceitual e lógico em uma estrutura compatível com o ambiente de execução escolhido.

Segundo Pressman e Maxim (2021), a modelagem física de dados define as estruturas internas necessárias para suportar as operações exigidas pelo sistema, sendo um componente fundamental da fase de projeto de banco de dados. O diagrama físico do banco de dados para este trabalho se encontra no Apêndice E.

## 3.3 CRONOGRAMA

Como citado na Seção 3.1, a concepção deste trabalho se deu em ciclos baseados e adaptados da metodologia Scrum. Esta seção busca apresentar as Sprints em dois aspectos: os ciclos que culminaram na produção deste documento, na modelagem e proposta da arquitetura do software, e os ciclos planejados para a fase de desenvolvimento do sistema proposto. O maior detalhamento de artefatos e eventos das Sprints pode ser encontrado na Seção 3.6 deste capítulo.

### 3.3.1 Fase de Modelagem

A primeira fase deste trabalho teve o propósito de modelar e projetar a arquitetura do software, além da concepção deste documento. As Sprints duraram duas semanas e foram compostas por Sprint Planning e Sprint Review, normalmente acompanhadas pelo orientador da equipe para planejar os Incrementos a serem

desenvolvidos e verificação do trabalho concluído. O QUADRO 2 apresenta as Sprints realizadas, datas de início e fim e o tema.

QUADRO 2 – RESUMO DAS SPRINTS DA PRIMEIRA FASE

<b>Sprint</b>	<b>Data</b>	<b>Tema</b>
Sprint 1	26/03/2024 - 08/04/2024	Levantamento e apresentação dos requisitos iniciais. Discussão de tecnologias
Sprint 2	09/04/2024 - 22/04/2024	Elaboração de Diagramas e Refinamento dos Requisitos
Sprint 3	23/04/2024 - 06/05/2024	Discussão e Ajustes dos Diagramas
Sprint 4	07/05/2024 - 20/05/2024	Apresentação e Validação de Artefatos
Sprint 5	21/05/2024 - 03/06/2024	Desenvolvimento da Prova de Conceito e Continuação de Documentação
Sprint 6	04/06/2024 - 17/06/2024	Revisão e Preparação para a Segunda Fase

FONTE: Os autores (2025).

### 3.3.2 Fase de Desenvolvimento

Com os requisitos, modelagem e arquitetura (apresentada no próximo capítulo) prontos, a próxima fase diz respeito ao desenvolvimento do software proposto neste documento. Para tanto, foram planejadas Sprints de duas semanas, cada qual com seus artefatos, que neste trabalho são as Histórias de Usuário, além da finalização do documento.

Para o planejamento dos artefatos das Sprints foram priorizadas as histórias de usuário com as funcionalidades mais importantes e com maior fator de complexidade. O QUADRO 3 evidencia o planejamento.

QUADRO 3 – PLANEJAMENTO DAS SPRINTS DA SEGUNDA FASE

<b>Sprint</b>	<b>Data</b>	<b>Artefatos</b>
Sprint 1	08/03/2025 - 21/03/2025	HU01 Cadastrar Usuário, HU02 Logar Usuário
Sprint 2	22/03/2025 - 04/04/2025	HU03 Cadastrar Questão, HU04 Recuperar Senha
Sprint 3	05/04/2025 - 18/04/2025	HU07 Cadastrar Avaliação, HU06 Interagir com ChatBot de Suporte
Sprint 4	19/04/2025 - 02/05/2025	HU08 Cadastrar Cabeçalhos,

Sprint 5	03/05/2025 - 16/05/2025	HU11 Corrigir Avaliações Aplicadas
Sprint 6	17/05/2025 - 30/05/2025	HU09 Criar Aplicação de Avaliação
Sprint 7	31/05/2025 - 13/06/2025	HU05 Usar Funções Automatizadas, HU10 Gerar e Visualizar Relatórios
Sprint 8	14/06/2025 - 30/06/2025	Correções, finalização do documento, preparação para a defesa

FONTE: Os autores (2025).

### 3.4 RESPONSABILIDADES

A fim de otimizar a concepção deste trabalho, a separação de tarefas e responsabilidades se deu conforme o QUADRO 4:

QUADRO 4 – TAREFAS E RESPONSABILIDADES

<b>Responsabilidade</b>	<b>Alexandre</b>	<b>André</b>	<b>Samuel</b>
Histórias de Usuário	X	X	X
Prototipação de Interfaces	X	X	X
Diagrama de Classes	X		
Diagrama de Casos de Uso		X	
Diagramas de Sequência	X	X	X
Diagramas do Banco de Dados	X		
Desenvolvimento Front-End	X	X	X
Desenvolvimento Back-End	X	X	X
Escrita do Documento	X	X	X

FONTE: O autor (2025).

Todavia, todos os membros da equipe foram ativos em todas as responsabilidades, fornecendo ideias e auxílio quando necessário. Essa divisão permitiu a flexibilidade no desenvolvimento de tarefas, permitindo o foco em uma mesma responsabilidade por todos da equipe ou múltiplas tarefas de forma concorrente, conforme necessidade e contexto.

### 3.5 MATERIAIS

Esta seção tem por objetivo apresentar as tecnologias usadas para o desenvolvimento deste trabalho.

#### 3.5.1 Figma

O Figma é uma plataforma de design e prototipagem de interfaces de usuário que se destaca por suas diversas funcionalidades e recursos. A escolha dessa ferramenta se deu por dois principais motivos: capacidade de criar protótipos interativos, que possibilita transformar interfaces estáticas em demonstrações funcionais, como interação com botões, navegação de telas, animações etc; e colaboração em tempo real entre membros da equipe, permitindo não apenas a colaboração no mesmo arquivo, mas a colaboração em tempo real, tornando possível que todos os participantes vejam as criações e alterações de protótipos conforme ocorrem (Figma, 2024). A plataforma foi utilizada em sua versão *web* para este projeto.

#### 3.5.2 Draw.io e Astah

Para criação de diagramas e modelagem do sistema, foram usados o Draw.io e o Astah. O Draw.io é uma ferramenta de diagramação *online* gratuita e poderosa para criação de diagramas de aplicações de forma simples e rápida. Conta com uma aplicação baseada em navegador, usada neste trabalho, e uma versão *desktop*, permitindo a alteração de qualquer arquivo em qualquer lugar. Ademais, a ferramenta tem uma ampla variedade de ícones e formas, porém mais importante tem notações UML, essenciais para este trabalho (Draw.io, 2024).

O Astah é uma ferramenta de modelagem e diagramação UML e SysML notório por sua simplicidade, qualidade e variedade de recursos. Ele oferece diversas ferramentas completas para concepção de software, criação de diagramas UML, modelagem SysML e engenharia de sistemas. Com funcionalidades como diagramas ER, fluxogramas, mapas mentais e muito mais, o Astah é uma boa opção para a criação de diagramas de software com clareza e eficiência (Astah, 2024). Para este trabalho, a versão usada foi a 9.

### 3.5.3 GitHub

Um sistema de controle de versão é um sistema que gerencia diferentes versões de um documento ou projeto de software. Ele permite acompanhar o histórico de desenvolvimento, trabalhar em paralelo em diferentes funcionalidades e resolver conflitos de edição entre desenvolvedores. Os sistemas de controle de versão são essenciais para organizar projetos, especialmente quando várias pessoas estão trabalhando juntas, ajudando a evitar problemas como a sobreposição de alterações e garantem que todos estejam trabalhando com as versões mais recentes do projeto (Chacon, 2014).

O GitHub é um serviço em nuvem que hospeda o Git, um sistema de controle de versão criado por Linus Torvalds. Este serviço incorpora os recursos disponíveis no Git, oferecendo uma plataforma de repositórios remotos que permitem a colaboração entre desenvolvedores e registro de alterações, funções essenciais para desenvolvimento de software e uma escolha natural para o gerenciamento de versões deste trabalho (GitHub, 2024).

### 3.5.4 Docker

Os contêineres são pacotes executáveis compostos por código de aplicação e suas dependências, que utilizam uma forma de virtualização para que sua execução seja possível em praticamente qualquer ambiente computacional (IBM, 2024).

A fim de se beneficiar desta tecnologia, adotou-se o Docker, uma plataforma de código aberto utilizada para desenvolvimento, entrega e execução de aplicações por meio de contêineres (Docker, 2024). O Docker simplifica a criação de imagens que definem ambientes completos, viabilizando a consistência entre os ambientes de desenvolvimento, teste e produção, sendo muito útil na arquitetura de microsserviços visto que cada módulo pode ser encapsulado e reproduzido em qualquer ambiente.

Para a definição, orquestração e gerenciamento de múltiplos contêineres, empregou-se o Docker Compose, uma ferramenta complementar ao Docker que permite declarar a infraestrutura de aplicações multicontêiner por meio de arquivos no formato YAML. Essa abordagem facilita a configuração de serviços interdependentes, como bancos de dados, servidores de aplicação e demais serviços, por meio de uma única interface de controle (Docker Compose, 2024).

### 3.5.5 Postman

Para a realização de testes e validações da API REST desenvolvida, foi utilizada a ferramenta Postman, uma plataforma amplamente empregada no desenvolvimento e consumo de APIs. O Postman oferece uma interface gráfica que permite a construção, envio e análise de requisições HTTP, facilitando o teste de *endpoints*, a verificação de respostas e o controle de parâmetros, cabeçalhos e autenticação (Postman, 2024).

### 3.5.6 VSCode

Durante o desenvolvimento da aplicação, foi utilizado o Visual Studio Code (VS Code) como ambiente de desenvolvimento integrado (IDE). O VS Code é uma ferramenta gratuita e de código aberto, desenvolvida pela Microsoft, que oferece uma interface leve e altamente extensível (Microsoft, 2024). Ele possui suporte nativo a diversas linguagens de programação, além de recursos como realce de sintaxe, IntelliSense (autocompletação inteligente), controle de versão integrado via Git e uma vasta gama de extensões disponíveis em seu marketplace. Essas funcionalidades contribuíram para aumentar a produtividade, facilitar a depuração de código e melhorar a organização do projeto durante o desenvolvimento da aplicação.

### 3.5.7 Linguagem de Modelagem Unificada

A Linguagem de Modelagem Unificada, do inglês *Unified Modeling Language* (UML) é uma linguagem visual utilizada para modelar sistemas de software de forma padronizada e compreensível. Criada com base na colaboração entre Grady Booch, Ivar Jacobson e James Rumbaugh, a UML se tornou uma linguagem padrão na Engenharia de Software.

A UML fornece uma forma padronizada de escrever a planta de um sistema, cobrindo coisas conceituais como processos de negócios e funções de sistema, e também coisas concretas como classes escritas em uma linguagem de programação específica, esquemas de base de dados, e componentes de software reutilizáveis (Booch; Rumbaugh; Jacobson, 2005, p. 13).

Essa linguagem fornece uma variedade de diagramas que permitem aos desenvolvedores representar diferentes aspectos de um sistema, desde a estrutura estática até o comportamento dinâmico.

### 3.5.8 Angular

Angular é uma plataforma de desenvolvimento e framework para design de aplicações web desenvolvida pela Google. Como plataforma, ele inclui um *framework* baseado em componentes para construções de aplicações escaláveis, coleção de bibliotecas cobrindo aspectos fundamentais de uma aplicação como rotas, manipulação de formulários, comunicação cliente-servidor e uma suíte de ferramentas de desenvolvimento (Angular, 2024).

O desenvolvimento do Front-end ou lado do cliente, se deu por meio deste *framework*. Usado em sua versão 17, o Angular fornece uma plataforma completa para criação de aplicações web de forma fácil e rápida. A sua arquitetura MVC (Model, View, Controller) possibilita a separação de responsabilidades entre os modelos de dados, serviços, visualizações e controladores, tornando o código organizado, de fácil entendimento e manutenção (Angular, 2024).

### 3.5.9 Java e Spring Boot

Java é uma linguagem de programação orientada a objetos e plataforma de computação lançada pela Sun Microsystems em 1995, posteriormente adquirida pela Oracle. É uma linguagem altamente popular e presente em grande parte do mundo digital atual por meio do fornecimento de uma plataforma confiável sobre a qual serviços e aplicações são criados (Java, 2024).

O Spring é um *framework* Java criado para simplificar o desenvolvimento de aplicações web e serviços. Criado a partir da dificuldade de desenvolvedores em criar aplicações corporativas em Java, o Spring usa conceitos como Inversão de Controle e Injeção de Dependência, além de fornecer muitos recursos integrados necessários a maioria das aplicações como módulos de segurança, testes, desenvolvimento web, e mais (Spring, 2024).

Para construção do Back-end ou lado do servidor, foi utilizado o framework Spring Boot em sua versão 3.2.5 com o Java em suas versões 17 e 22. O Spring Boot

permite a criação rápida de aplicações e tem grande variedade de bibliotecas, plugins e dependências, como a do PostgreSQL usada para integração com banco de dados. Outras importantes dependências são a Spring Web, que permite a criação de APIs REST e contém auto-configuração, servidores embutidos e anotações, além do Spring Data JPA, que facilita a implementação de persistência de dados com a Java Persistence API, permitindo ao Spring criar aplicações com camadas de acesso de dados, queries, auditoria, validação e mais (Spring Framework, 2024).

### 3.5.10 REST

*Representational State Transfer* (REST), em português, Transferência Representacional de Estado, é um estilo arquitetural para sistemas distribuídos, popularizado por Roy Fielding (2000) em sua tese de doutorado. Segundo Fowler, um dos principais especialistas em padrões de design de software, REST é um estilo de arquitetura de software para sistemas hipermídia distribuídos, caracterizado por um conjunto de restrições de design. Essas restrições incluem o uso de verbos HTTP (GET, POST, PUT, DELETE) para manipular recursos, a identificação de recursos usando URIs, o uso de representações autossuficientes e a comunicação sem estado entre cliente e servidor.

Fowler (2009) diz que a adoção de REST pode simplificar a arquitetura de sistemas distribuídos e melhorar a escalabilidade e desempenho, tornando-o uma escolha popular para o desenvolvimento de APIs web modernas.

### 3.5.11 PostgreSQL e pgAdmin

O PostgreSQL, também conhecido como *Postgres*, é um sistema gerenciador de banco de dados relacional de código aberto amplamente utilizado por sua confiabilidade, robustez e extensibilidade (PostgreSQL, 2024). Apresenta suporte para recursos avançados como transações ACID (atomicidade, consistência, integridade e disponibilidade), chaves estrangeiras, procedimentos armazenados e replicação, o PostgreSQL é uma boa escolha para aplicações que exigem persistência de dados e desempenho escalável.

Este sistema vem com muitos recursos para ajudar desenvolvedores na construção de aplicativos, além de métodos de administração para proteger integridade de dados e ambientes tolerantes a falhas.

Para a administração e o desenvolvimento do PostgreSQL, foi empregado o pgAdmin, uma plataforma de código aberto que oferece uma interface gráfica robusta, destinada à gestão e ao desenvolvimento de bancos de dados PostgreSQL. O pgAdmin simplifica operações como a criação, manutenção e interação com o banco de dados, além de oferecer um conjunto extenso de ferramentas, tanto básicas quanto avançadas (PgAdmin, 2024).

### 3.5.12 MongoDB

O MongoDB é um sistema de gerenciamento de banco de dados NoSQL orientado a documentos, amplamente utilizado por sua flexibilidade, escalabilidade horizontal e facilidade de uso. Ele armazena dados em documentos no formato BSON (*Binary JSON*, uma extensão do *JavaScript Object Notation*), permitindo maior agilidade na manipulação e consulta de dados semi-estruturados.

No projeto, o MongoDB foi utilizado para armazenar senhas de forma segura, aproveitando sua capacidade de lidar com grandes volumes de dados e sua integração com bibliotecas de criptografia. A administração do banco pode ser realizada com ferramentas como o MongoDB Compass ou diretamente via terminal, o que facilita o gerenciamento e a visualização dos dados armazenados (MONGODB, 2024).

### 3.5.13 React Native

O React Native é um framework de código aberto desenvolvido pelo Facebook que permite a criação de aplicativos móveis nativos usando JavaScript e React. Com o React Native, os desenvolvedores podem criar aplicativos para iOS e Android usando uma única base de código, economizando tempo e recursos. Ao utilizar os mesmos princípios do React, como componentes reutilizáveis e uma arquitetura baseada em fluxo de dados unidirecional, o React Native oferece uma experiência de desenvolvimento eficiente e familiar.

Além disso, o framework permite o acesso a recursos nativos do dispositivo, como câmera, GPS e sensores, proporcionando aplicativos com desempenho e aparência nativos (React Native, 2024).

O desenvolvimento do software apresentado utilizou a versão 0.74.5 do React Native e a versão 18.2 do React.

#### 3.5.14 RabbitMQ

O RabbitMQ é um sistema *open-source* de intermediação de comunicação de mensagens, operando sobre o protocolo AMQP (*Advanced Message Queue Protocol*). Seu uso permite uma comunicação assíncrona entre microsserviços de forma robusta, segura e escalável.

O RabbitMQ utiliza o padrão de filas (*queues*) e *exchanges* para roteamento de mensagens. Isso permite que um serviço publique uma mensagem sem se preocupar com quem irá consumi-la, promovendo uma arquitetura mais flexível e resiliente a falhas (RabbitMQ, 2024).

Ademais, sua integração ao Spring é facilitada pela ferramenta Spring AMQP, que abstrai complexidade e facilita o desenvolvimento de produtores e consumidores de mensagens (Spring, 2024). A versão usada do RabbitMQ foi a 3.13.7.

#### 3.5.15 OpenCV

O OpenCV (*Open Source Computer Vision Library*) é uma biblioteca de software de código aberto voltada para visão computacional e aprendizado de máquina. Desenvolvida inicialmente pela Intel, a biblioteca agora é suportada por Willow Garage e Itseez. Multiplataforma, o OpenCV oferece mais de 2.500 algoritmos otimizados, que podem ser utilizados para uma vasta gama de aplicações, como reconhecimento facial, detecção de objetos, classificação de ações humanas em vídeos, rastreamento de movimento, extração de modelos 3D, criação de panoramas, realidade aumentada, entre outros.

A biblioteca é conhecida por sua extensa coleção de algoritmos, sendo uma ferramenta essencial para tarefas de visão computacional e aprendizado de máquina. Funciona em sistemas operacionais como Windows, Linux, macOS, Android e iOS, e suporta várias linguagens de programação, incluindo C++, Python, Java e

MATLAB/Octave, com C++ e Python sendo as mais populares. O OpenCV se integra bem com outras bibliotecas, como NumPy, facilitando a manipulação de arrays e a integração com ferramentas de ciência de dados. Além disso, utiliza otimizações de hardware para acelerar o processamento de imagens e vídeos, aproveitando instruções SIMD, multi-threading e aceleração por GPU, através de tecnologias como CUDA e OpenCL.

Devido à sua versatilidade e desempenho, o OpenCV é amplamente utilizado tanto na indústria quanto na academia, facilitando o desenvolvimento de aplicativos que exigem processamento de imagens e vídeos em tempo real (OpenCV, 2024). Para este trabalho, foi utilizado a dependência do OpenCV em Python, na versão 4.11.

### 3.5.16 Hardware utilizado

Para o desenvolvimento, execução e teste das aplicações, foram utilizados notebooks e desktops com as seguintes especificações:

- Notebook 1: Processador AMD Ryzen 7 5700U, memória RAM de 12 GB e SSD NVMe;
- Notebook 2: Processador Intel i5 12450H, memória RAM de 8GB e SSD NVMe;
- Notebook 3: Processador AMD Ryzen 5 5600G, memória RAM de 16GB e SSD SATA;
- Desktop 1: Processador Intel i7 9700KF, memória RAM de 16GB e SSD NVMe.

## 3.6 DESENVOLVIMENTO DO PROJETO

Esta seção busca detalhar as Sprints realizadas para o desenvolvimento deste trabalho.

### 3.6.1 Sprint 1

A Sprint 1, iniciada em 26/03/24 e finalizada em 08/04/24, teve por objetivo principal apresentar os requisitos iniciais levantados, centralizando e alinhando ideias e no geral definindo as principais funcionalidades desejadas. Para facilitar o entendimento dos requisitos, nesta Sprint foi iniciada a criação de Histórias de Usuário

(Apêndice B), porém em sua versão inicial, com menos detalhes e sem os desenhos de telas.

### 3.6.2 Sprint 2

A Sprint 2, de 09/04/2024 a 22/04/2024, se focou em refinar os requisitos por meio do aprimoramento das Histórias de Usuário, e para apoiar essa meta foi iniciada a criação do Diagrama de Casos de Uso (Apêndice A) e protótipos de tela. Além disso, com o objetivo de definir a estrutura do sistema foi dado início ao desenvolvimento do Diagrama de Classes (Apêndice C). Ademais, a concepção do Capítulo 1 se deu nessa Sprint.

### 3.6.3 Sprint 3

Na Sprint 3, iniciada em 23/04/2024 e finalizada em 06/05/2024, definiu-se as metas de revisar e ajustar os diagramas de Casos de Uso e Diagrama de Classes para sua versão final, também a discussão da prova de conceito a ser implementada. No geral, essa Sprint focou-se em revisar diagramas e iniciar a escrita do Capítulo 2.

### 3.6.4 Sprint 4

A avaliação dos diagramas criados até então e dos Capítulos 1 e 2, somados ao início do desenvolvimento de protótipos foram as metas principais da Sprint 4. Durante o Sprint Review deste ciclo foi detectado a necessidade de agilizar a escrita do documento, o que culminou na meta de desenvolvimento e correção simultânea dos capítulos escritos até então e início do Capítulo 3 para o ciclo seguinte. Além disso, o esquema físico do Banco de Dados (Apêndice E) derivado do Diagrama de Classes (Apêndice C) foi uma meta adicionada posteriormente a essa Sprint. Sua data de início foi em 07/05/2024 e fim em 20/05/2024.

### 3.6.5 Sprint 5

Iniciada em 21/05/2024 a 03/06/2024, a Sprint 5 buscou entregar a versão definitiva dos Capítulos 1 a 3, Diagramas de Classes (Apêndice C) e de Casos de Uso

(Apêndice A), e as Histórias de Usuário detalhadas e com desenhos de telas (Apêndice B). Nessa Sprint também foi iniciado o desenvolvimento da prova de conceito.

### 3.6.6 Sprint 6

Para finalizar a primeira fase do trabalho, a Sprint 6 contou com a finalização dos documentos relacionados à concepção dos Capítulos 4 e 5 e dos diagramas de sequência (Apêndice D), a conclusão da prova de conceito e os ajustes necessários. Também incluiu uma Sprint Retrospective adaptada, na qual foram discutidas não apenas a última Sprint, mas o desenvolvimento da primeira fase no geral, além do planejamento para a segunda fase. Realizada em 04/06/2024 a 17/06/2024.

### 3.6.7 Sprint 7

Durante a Sprint 7, já na fase de desenvolvimento de software, foi iniciada a implementação das funcionalidades básicas de acesso ao sistema, com foco no cadastro de novos usuários (HU01) e login (HU02). Essa etapa marcou o início da estruturação técnica do sistema, com a configuração do ambiente de desenvolvimento, testes iniciais de autenticação e início da integração com serviços de e-mail para verificação de conta. Realizada em 08/03/2025 - 21/03/2025.

### 3.6.8 Sprint 8

Na Sprint 8, o trabalho se concentrou no início da funcionalidade de cadastro de questões (HU03), uma das mais complexas do sistema, e na funcionalidade de recuperação de senha (HU04). Devido à amplitude das operações de CRUD necessárias para o gerenciamento completo de questões, essa HU exigiu bastante esforço técnico e acabou se estendendo para a Sprint seguinte. Já a funcionalidade de recuperação de senha foi desenvolvida de forma paralela, garantindo suporte aos usuários em caso de esquecimento ou alteração de senha. Ocupou o período de 22/03/2025 a 04/04/2025.

### 3.6.9 Sprint 9

A Sprint 9, de 05/04/2025 até 18/04/2025, deu continuidade à implementação do cadastro de questões (HU03), finalizando a lógica de CRUD e validando a interface com o banco de dados. Também foi iniciada a funcionalidade do ChatBot de Suporte (HU06), que demandou estudos sobre integração com serviços de IA e linguagem natural, estabelecendo as primeiras interações automatizadas com o usuário. Além disso, iniciou-se o desenvolvimento do cadastro de avaliações (HU07), etapa que se estenderia por mais Sprints devido à sua complexidade.

### 3.6.10 Sprint 10

Durante a Sprint 10, o foco foi aprimorar o cadastro de avaliações (HU07) com a adição do cadastro de cabeçalhos personalizados (HU08), oferecendo maior flexibilidade para a criação das provas. Paralelamente, iniciou-se o desenvolvimento da funcionalidade de correção de avaliações através da câmera (HU11), tarefa que se adiantou nesta Sprint devido à necessidade de testes mais aprofundados com tecnologias como OpenCV e OCR. A complexidade dessa integração trouxe desafios significativos relacionados ao reconhecimento de marcações, exigindo ajustes contínuos. Realizada em 19/04/2025 a 02/05/2025.

### 3.6.11 Sprint 11

A Sprint 11, de 03/05/2025 a 16/05/2025, foi marcada pela dedicação ao desenvolvimento da funcionalidade de correção automatizada via câmera (HU11), cuja base foi preparada na Sprint anterior. Os esforços foram voltados para a finalização dos testes com imagens reais de avaliações e refinamento da detecção de respostas. O sistema passou a ser capaz de interpretar os dados capturados e compará-los com o gabarito, fechando um importante ciclo de automação. Essa Sprint também avançou o trabalho na criação da aplicação de avaliação (HU09), que exigiu integração entre outros módulos do sistema.

### 3.6.12 Sprint 12

Na Sprint 12, deu-se continuidade ao desenvolvimento da funcionalidade de criação da aplicação de avaliação (HU09), assegurando a geração de provas com configuração personalizada. Também foram iniciadas as implementações das funções automatizadas (HU05), como sugestões de questões e correção gramatical, e melhorias no ChatBot de suporte (HU06), já que ambos compartilham componentes de processamento de linguagem natural. Essas funcionalidades exigiram integração com APIs externas e ajustes de desempenho para garantir a usabilidade do sistema. Ocorreu entre 17/05/2025 e 30/05/2025.

### 3.6.13 Sprint 13

A Sprint 13, iniciada em 31/05/2025 até 13/06/2025, deu sequência às funcionalidades de automação (HU05) e foi concluída a funcionalidade de geração e visualização de relatórios (HU10). O sistema passou a fornecer indicadores sobre as avaliações, como desempenho dos alunos, taxa de acertos por questão etc., permitindo ao professor uma análise mais rica das aplicações realizadas. Essa Sprint consolidou o ciclo de feedback para o usuário e reforçou o uso de inteligência de dados no sistema.

### 3.6.14 Sprint 14

A última Sprint teve como foco a correção de bugs, ajustes visuais e a finalização do documento do projeto. Nessa etapa, também foi feita uma preparação para a apresentação final e a defesa do trabalho. A equipe realizou testes finais de usabilidade, consolidou todos os artefatos desenvolvidos nas Sprints anteriores e documentou os aprendizados obtidos, encerrando a segunda fase do desenvolvimento com estabilidade e funcionalidade completas. Realizada em 14/06/2025 até 30/06/2025.

## 4 APRESENTAÇÃO DO SISTEMA

Neste capítulo apresenta-se a arquitetura proposta para este sistema. Ela é baseada em tecnologias atuais e amplamente utilizadas, como Angular, React Native, Spring Boot e PostgreSQL, que permite a construção de um sistema robusto, confiável e fácil de manter.

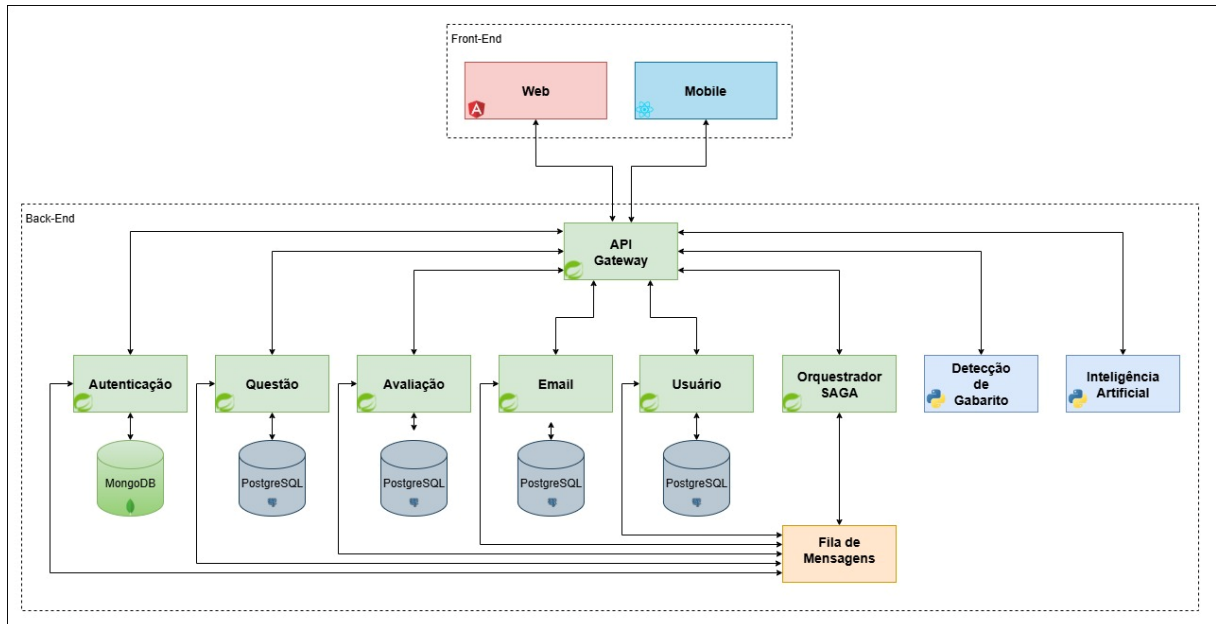
A arquitetura de microsserviços é um estilo de design de software onde o sistema é dividido em pequenos serviços independentes, que se comunicam entre si por meio de APIs. Esse modelo permite que cada serviço seja desenvolvido, implantado e escalado de forma isolada, facilitando a manutenção, a atualização e a resiliência do sistema como um todo. Além disso, ela contribui para uma maior flexibilidade tecnológica, permitindo o uso de diferentes linguagens ou bancos de dados em diferentes serviços, por exemplo, os serviços de inteligência artificial e processamento digital de imagens foram desenvolvidos em Python, contrastando com os demais desenvolvidos em Java.

A arquitetura do sistema, como visto na FIGURA 10, foi projetada para seguir um modelo de comunicação via internet por HTTP entre camadas ou AMQP via RabbitMQ entre microsserviços, garantindo escalabilidade e portabilidade. O usuário interage com o sistema por uma interface web (navegador) ou mobile (aplicativo) acessível por um dispositivo necessariamente conectado à internet. Essas interações acontecem no *frontend*, onde as solicitações dos usuários são processadas, ilustrados pelos elementos Web e Mobile na primeira camada da figura.

Quando um usuário faz uma requisição, ela é enviada ao servidor da aplicação. No *backend*, o sistema utiliza um API Gateway, para gerenciar e direcionar essas requisições, encontrado na segunda camada na FIGURA 10. O API Gateway atua como um ponto central de entrada, que encaminha as solicitações para os microsserviços correspondentes (Richardson, 2025).

Após o API Gateway há a camada de microsserviços, onde cada um é responsável por uma parte específica da lógica de negócios e interage com seu respectivo banco de dados. Para transações que envolvem múltiplos serviços, é utilizado um Orquestrador SAGA, que ordena e comunica via mensageria os passos necessários da operação para cada módulo, também enviando ordens de reversão em caso de erro.

FIGURA 10 – ARQUITETURA DO SISTEMA

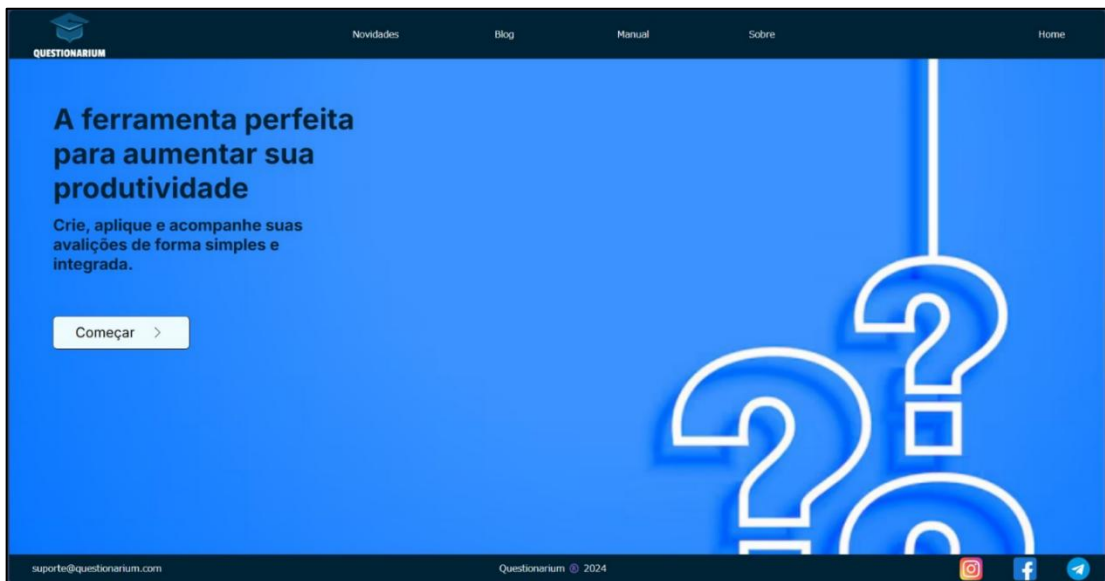


FONTE: O autor (2025).

#### 4.1 TELAS DA APLICAÇÃO WEB

A seção a seguir tem por objetivo mostrar as telas da aplicação e suas funcionalidades.

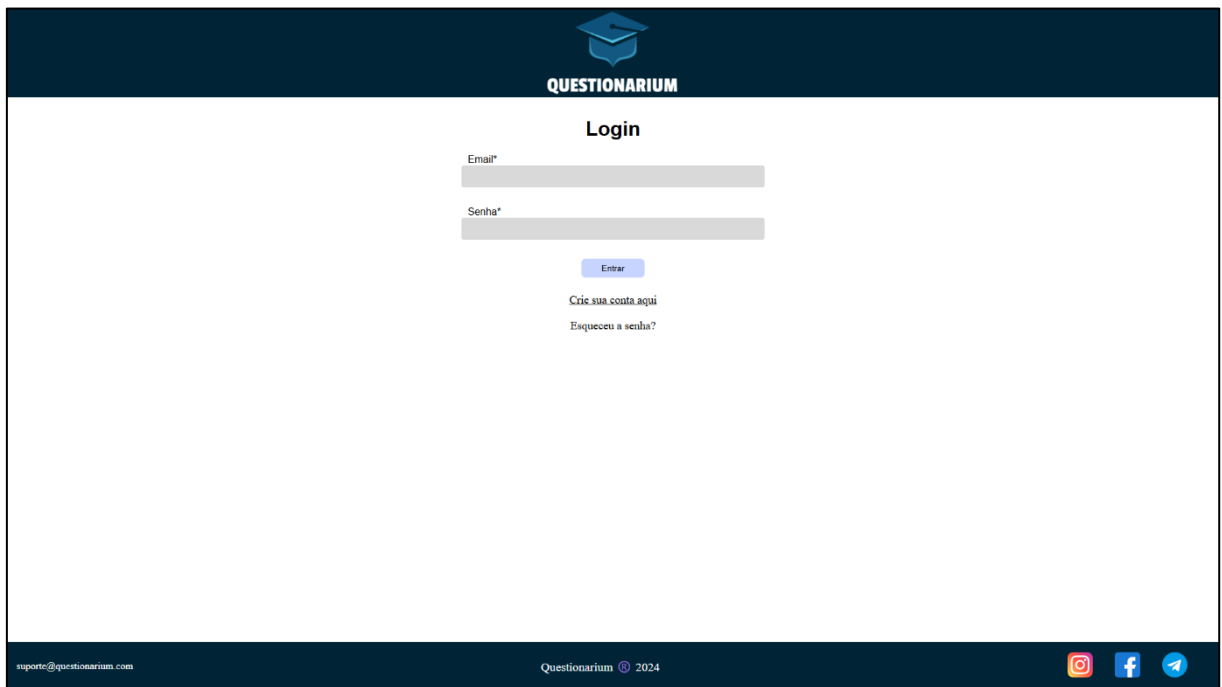
FIGURA 11 – TELA LANDING PAGE



FONTE: O autor (2025).

A tela Landing Page (FIGURA 11) tem objetivo apresentar o sistema e convencer visitantes a conhecerem mais sobre as funcionalidades do site.

FIGURA 12 – TELA LOGIN



QUESTIONARIUM

Login

Email\*

Senha\*

Entrar

[Crie sua conta aqui](#)

[Esqueceu a senha?](#)

suporte@questionarium.com

Questionarium © 2024

FONTE: O autor (2025).

FIGURA 13 – TELA LOGIN APÓS CLICAR EM ESQUECI A SENHA



Alteração de Senha X

Digite o seu email:

Enviar

FONTE: O autor (2025).

A tela de Login (FIGURA 12) tem a função de logar o usuário no sistema por meio dos campos de email e senha. Ademais, apresenta links para tela de criação de conta e de recuperação de senha que traz um novo campo, ilustrado pela FIGURA 13.

FIGURA 14 – TELA CADASTRO

QUESTIONARIUM

**Cadastro**

Login\*

Email\*

Senha\*

Cadastrar

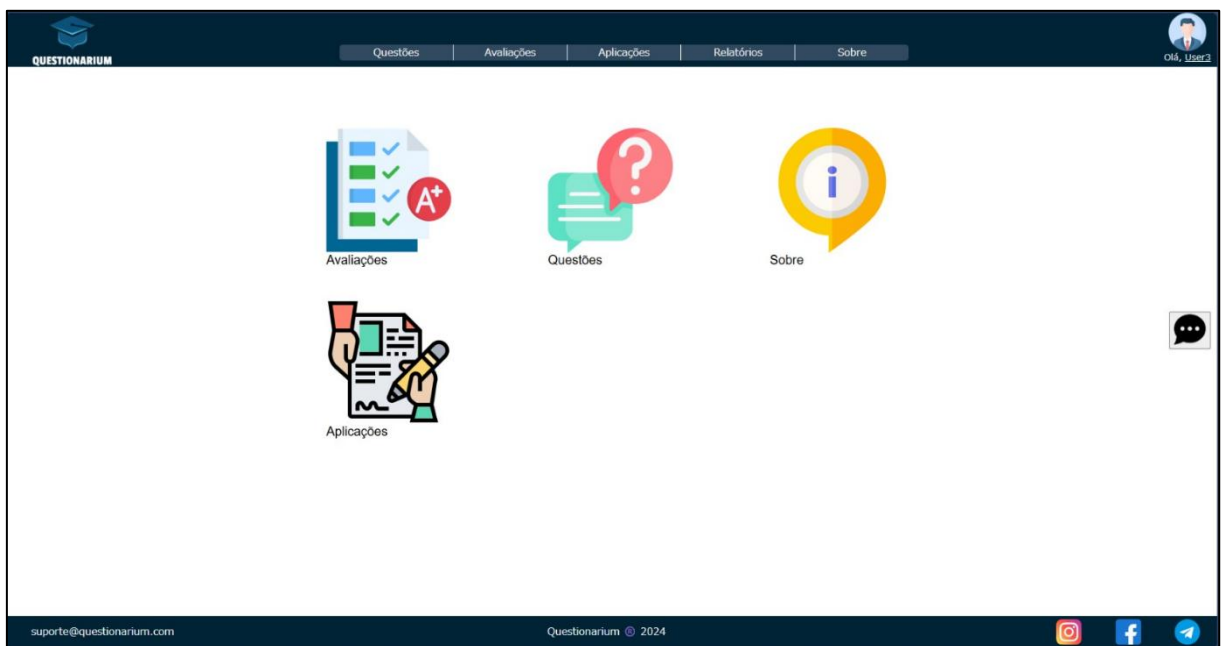
Voltar

suporte@questionarium.com Questionarium © 2024

FONTE: O autor (2025).

A tela Cadastro (FIGURA 14) traz campos para registro de usuário, como login, email e senha. Também apresenta um link que redireciona o usuário a tela Landing Page.

FIGURA 15 - TELA INICIAL



FONTE: O autor (2025).

A tela Inicial (FIGURA 15) traz de forma simplificada os módulos principais do sistema, através de ícones ilustrativos, facilitando a navegação. Além disso, conta com um ícone para abrir o modal do ChatBot, que por meio de inteligência artificial e banco de dados personalizado responde perguntas do usuário sobre funcionalidades do sistema e desenvolvedores.

FIGURA 16 - TELA REDEFINIÇÃO DE SENHA

A imagem mostra a interface de redefinição de senha do sistema Questionarium. O cabeçalho contém o logo 'QUESTIONARIUM' e um menu com as opções 'Questões', 'Avaliações', 'Aplicações', 'Relatórios' e 'Sobre'. No canto superior direito, há um ícone de perfil de usuário com o nome 'Olá, User2'. O título centralizado da página é 'Redefinição de Senha'. Abaixo do título, há dois campos de entrada de texto: 'Nova Senha\*' e 'Confirmar a Senha\*', ambos com barras cinzas para o texto. Abaixo dos campos, há um botão azul com o texto 'Salvar'. Na barra de rodapé, há o endereço 'suporte@questionarium.com', o copyright 'Questionarium © 2024' e ícones para Instagram, Facebook e WhatsApp.

FONTE: O autor (2025).

A partir da solicitação de recuperação de senha na tela de login, o usuário é encaminhado a tela de redefinição de senha, contando com os campos de senha e confirmação, como mostrado na FIGURA 16.

FIGURA 17 - TELA EDIÇÃO DE PERFIL

A imagem mostra a interface de edição de perfil do sistema Questionarium. O cabeçalho é idêntico ao da Figura 16. No canto superior direito, o nome de usuário é 'Olá, User2'. À esquerda, há um menu lateral com as opções 'Conta' e 'Alterar Senha'. O formulário principal contém dois campos de texto: 'Nome' e 'Email', ambos com barras cinzas. À direita, há uma seção 'Foto de Perfil:' com uma imagem de perfil de um homem em um terno e um botão 'Enviar nova foto'. Abaixo dos campos de texto, há um botão verde com o texto 'Salvar'. Na barra de rodapé, há o endereço 'suporte@questionarium.com', o copyright 'Questionarium © 2024' e ícones para Instagram, Facebook e WhatsApp.

FONTE: O autor (2025).

A tela de Edição de Perfil (FIGURA 17) tem a finalidade de possibilitar o usuário editar seu nome no sistema mediante o preenchimento de campo de entrada.

FIGURA 18 - TELA EDIÇÃO DE SENHA

FONTE: O autor (2025).

Finalizando as telas de Edição, a tela de Edição de Senha (FIGURA 18), traz os campos para redefinição de senha do usuário.

FIGURA 19 - TELA VISUALIZAÇÃO DE QUESTÃO

Texto	Discursiva	Nível de Acesso	Nível	
Exemplo de cabeçalho questao 17	Não	Privado	ENSINO_FUNDAMENTAL	Visualizar
Exemplo de cabeçalho questao 17	Não	Privado	ENSINO_FUNDAMENTAL	Visualizar
Exemplo de cabeçalho questao 27	Não	Privado	ENSINO_FUNDAMENTAL	Visualizar
Exemplo de cabeçalho questao 37	Não	Privado	ENSINO_FUNDAMENTAL	Visualizar
Exemplo de cabeçalho questao 47	Não	Privado	ENSINO_FUNDAMENTAL	Visualizar
Com base nas diretrizes de referência de conteúdos dos curso...	Não	Público	ENSINO_FUNDAMENTAL	Visualizar
Um engenheiro (de produção, civil, mecânica etc.) é formado p...	Não	Público	ENSINO_FUNDAMENTAL	Visualizar

FONTE: O autor (2025).

A tela de visualização de questão (FIGURA 19) contém as questões do banco, com a possibilidade de filtros por categoria, se a questão é discursiva, nível de ensino, como ensino médio e fundamental por exemplo, se a questão é pública e um campo para pesquisa de texto. Além disso, o botão Nova Questão navega para tela de Cadastro, enquanto o botão Visualizar leva a tela de Edição.

FIGURA 20 - TELA CADASTRO E EDIÇÃO DE QUESTÃO

FONTE: O autor (2025).

A tela de Cadastro e Edição de Questão (FIGURA 20) é composta por diversos campos e funcionalidades para criação de questões. A seção de filtros traz diversas características importantes para a classificação da questão, como categoria, se é discursiva e pública, assim como nível de ensino. Ademais, apresenta os campos de enunciado, alternativas e explicação da alternativa, além de destacar a alternativa correta. Demais funcionalidades são a correção gramatical dos textos com IA, com a possibilidade de revertê-los para o estado original e Geração Automática apresentada na figura a seguir.

FIGURA 21 - TELA DE GERAÇÃO AUTOMÁTICA DE QUESTÃO

FONTE: O autor (2025).

A tela de Geração de Questão (FIGURA 21) é outra funcionalidade automatizada da aplicação, que por meio de campos como tema, categoria, nível de ensino e discursiva ou múltipla escolha gera uma questão automaticamente com IA. Nesta tela, o usuário pode aceitar a questão, levando-o para a tela de edição caso queira realizar alterações, gerar outra com os mesmos parâmetros ou cancelar a operação.

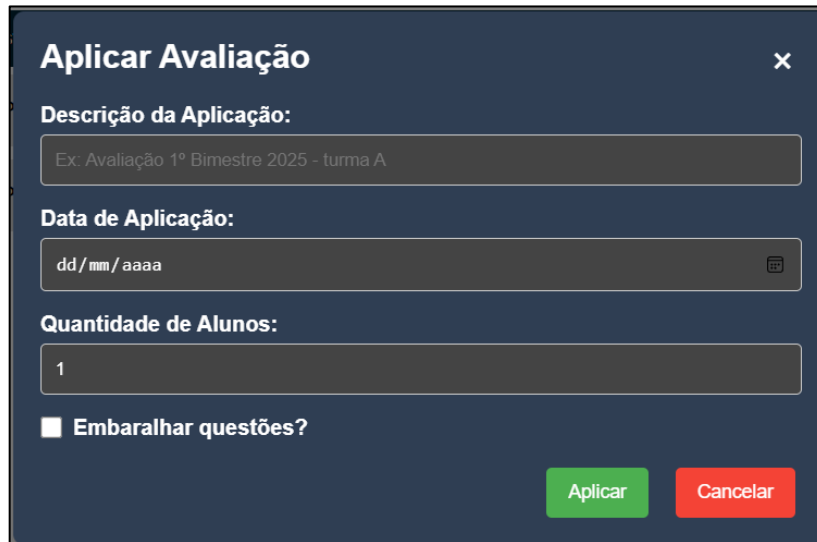
FIGURA 22 - TELA DE LISTAGEM DE AVALIAÇÕES

ID	Descrição	Instituição	Disciplina	Classe	Ações
1	Avaliação Inicial	Universidade Federal do Paraná	DAC	N1	Atualizar Aplicar Deletar
2	Avaliação Inicial	Universidade Federal do Paraná	DAC	N1	Atualizar Aplicar Deletar

FONTE: O autor (2025).

A FIGURA 22, ilustra a listagem de avaliações, onde o usuário pode criar um modelo de avaliação, editar um modelo existente, criar uma aplicação ou remover, com auxílio de uma barra de pesquisa para filtragem.

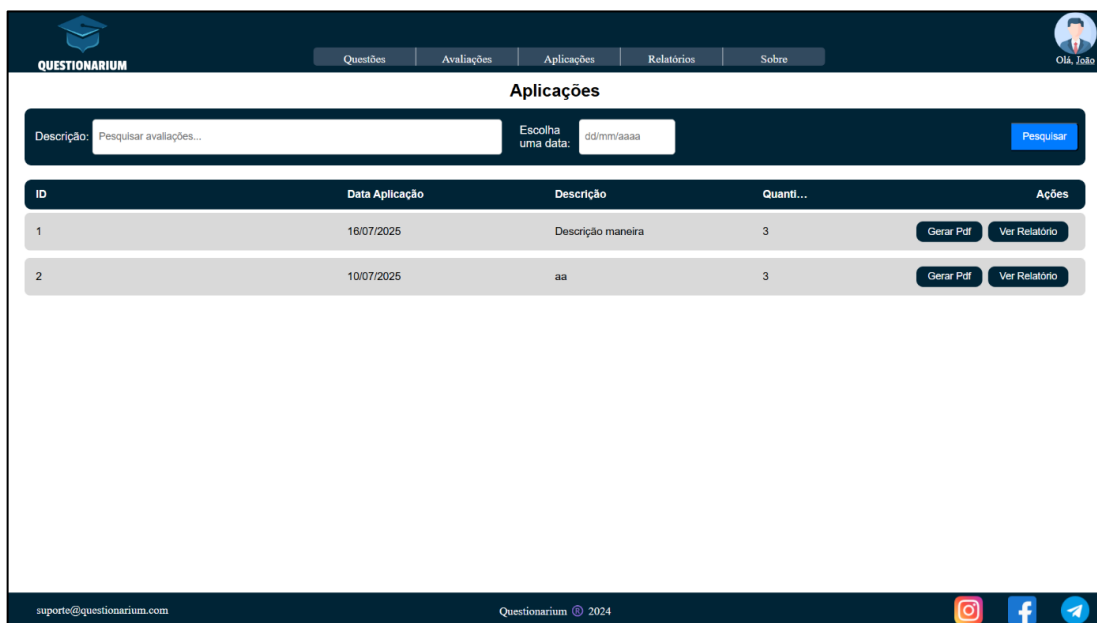
FIGURA 23 - TELA DE CRIAÇÃO DE APLICAÇÃO



FONTE: O autor (2025).

A FIGURA 23, ilustra a tela de criação de aplicação, onde a partir de um modelo o usuário determinará alguns parâmetros para geração de aplicação como embaralhamento de questões, data de aplicação e quantidade de aplicações.

FIGURA 24 - TELA DE LISTAGEM DE APLICAÇÃO



ID	Data Aplicação	Descrição	Quanti...	Ações
1	18/07/2025	Descrição maneira	3	Gerar Pdf Ver Relatório
2	10/07/2025	aa	3	Gerar Pdf Ver Relatório

FONTE: O autor (2025).

A tela Listagem de Aplicação, mostra todas as aplicações criadas pelo usuário, mostrando dados como identificador, data de aplicação, descrição e número de aplicações, ilustrada na FIGURA 24.

FIGURA 25 - TELA DE CRIAÇÃO/EDIÇÃO DE AVALIAÇÃO

FONTE: O autor (2025).

Ao selecionar uma avaliação ou clicando em Nova Avaliação, o usuário é levado para a tela de criação e edição de modelo de avaliação (FIGURA 25), onde pode selecionar ou criar um cabeçalho e selecionar ou remover questões.

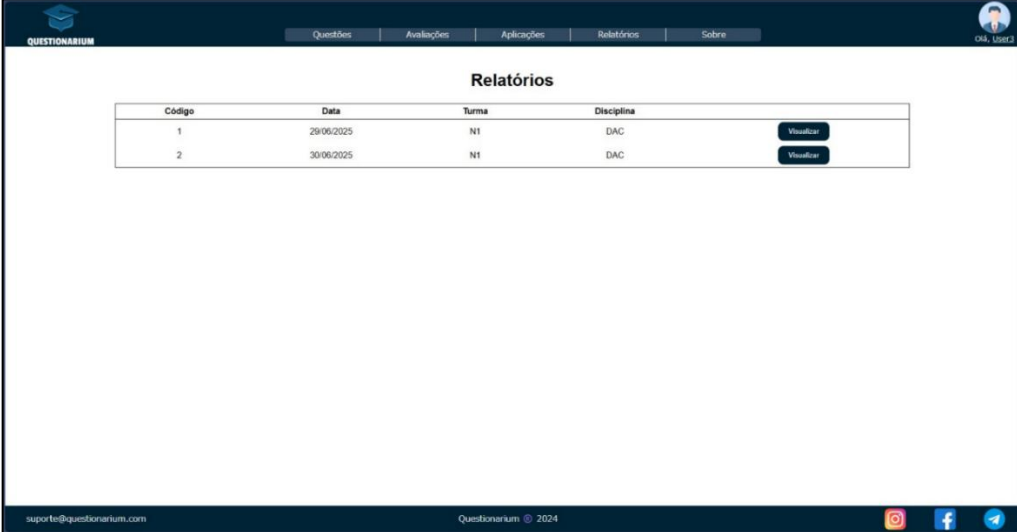
FIGURA 26 - TELA DE SELEÇÃO DE QUESTÕES

Nível	Acesso	Texto	Data criação
ENSINO_MEDIO	Público	[UFPR-2025] Uma fonte de ondas produz, em um	03/07/2025
ENSINO_MEDIO	Público	[UFPR - 2025] Uma dada lente delgada convergen	03/07/2025
ENSINO_MEDIO	Público	[UFPR - 2025] Em uma vasta região, há uma popu	03/07/2025
ENSINO_MEDIO	Público	[UFPR - 2025] Sobre a função e o transporte da se	03/07/2025
ENSINO_MEDIO	Público	[UFPR - 2025] Sobre as enzimas e a sua atuação	03/07/2025

FONTE: O autor (2025).

A partir da tela de criação de avaliação (FIGURA 26) é possível selecionar as questões previamente criadas pelo usuário, para compor o modelo de avaliação.

FIGURA 27 - TELA DE RELATÓRIOS DE AVALIAÇÃO

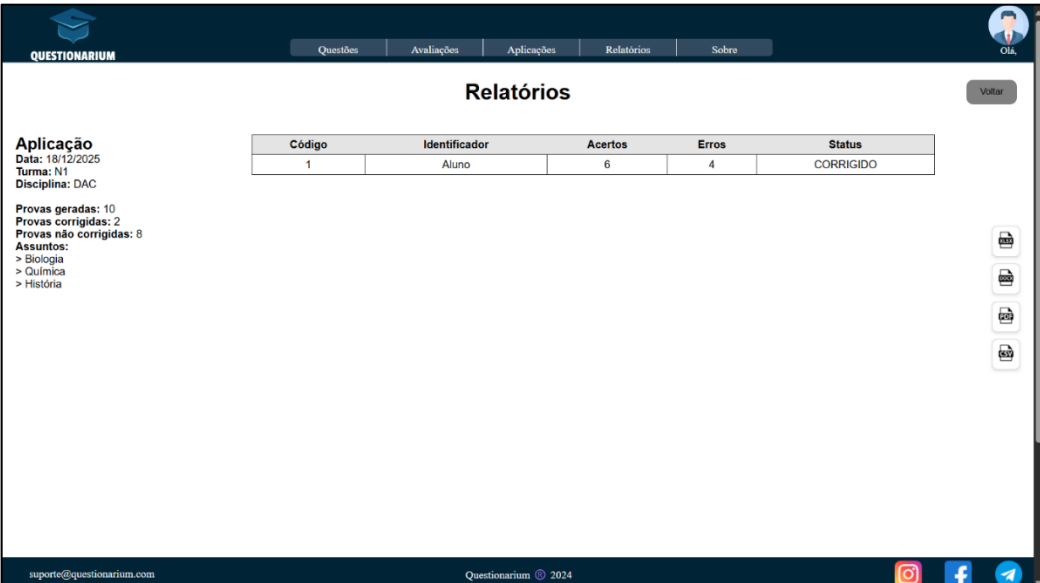


Código	Data	Turma	Disciplina
1	29/06/2025	N1	DAC
2	30/06/2025	N1	DAC

FONTE: O autor (2025).

A tela de Relatórios (FIGURA 27) apresenta uma tabela com as aplicações de avaliação para análise do usuário. O botão Visualizar redireciona para a aplicação específica selecionada, ilustrado pela figura abaixo.

FIGURA 28 - TELA DE RELATÓRIO DE APLICAÇÃO



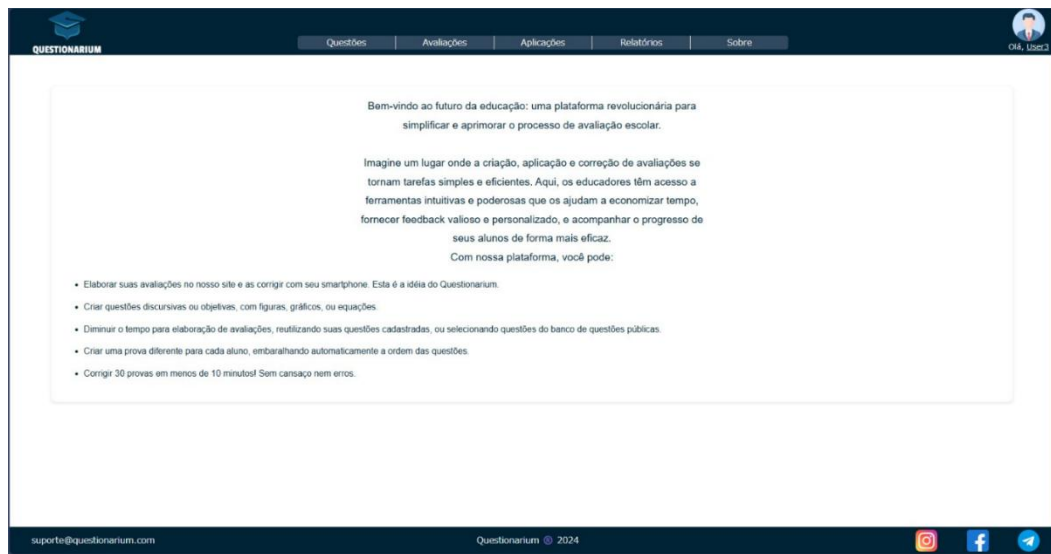
Código	Identificador	Acertos	Erros	Status
1	Aluno	6	4	CORRIGIDO

**Aplicação**  
 Data: 18/12/2025  
 Turma: N1  
 Disciplina: DAC  
 Provas geradas: 10  
 Provas corrigidas: 2  
 Provas não corrigidas: 8  
 Assuntos:  
 > Biologia  
 > Química  
 > História

FONTE: O autor (2025).

A tela de Relatório de Aplicação (FIGURA 28) contém os dados da aplicação selecionada, no lado esquerdo dados como data, turma, disciplina etc. Além de número de provas geradas, corrigidas e assuntos da avaliação. A tabela, por sua vez, apresenta cada resposta fornecida para a aplicação em questão, contando com código, identificação do aluno, acertos, erros e status.

FIGURA 29 - TELA SOBRE



FONTE: O autor (2025).

A tela “Sobre” (FIGURA 29) traz algumas informações e objetivos do Questionarium.

FIGURA 30 - TELA LISTAR CABEÇALHOS

Buscar Cabeçalho:

Instituição	Departamento	Curso	Turma	Professor	Ações
Universidade Federal do Paraná	TADS	DAC	N1	Razer	<span>Selecionar</span> <span>Excluir</span>
Universidade Federal do Rio Grande do Sul	BCC	SWE	N1	Anthon	<span>Selecionar</span> <span>Excluir</span>

Cancelar

FONTE: O autor (2025).

A Figura 30 apresenta a tela Listar Cabeçalhos, que mostra os cabeçalhos já cadastrados pelo usuário, com as opções de carregar o cabeçalho na avaliação e exclusão de um cabeçalho.

FIGURA 31 - TELA CRIAR/EDITAR CABEÇALHOS

A imagem mostra a interface de usuário para criar ou editar um cabeçalho. O título da tela é "Criar Cabeçalho". Abaixo do título, há um ícone de fechar (X). O formulário contém os seguintes campos:

- Instituição:** Campo de texto com o exemplo "Ex: UFPR".
- Departamento:** Campo de texto com o exemplo "Ex: DAC".
- Curso:** Campo de texto com o exemplo "Ex: TADS".
- Turma:** Campo de texto com o exemplo "Ex: B5".
- Professor:** Campo de texto com o exemplo "Nome do professor".
- Instruções:** Área de texto para "Instruções para os alunos".

Na parte inferior direita da tela, há dois botões: "Salvar" (verde) e "Cancelar" (vermelho).

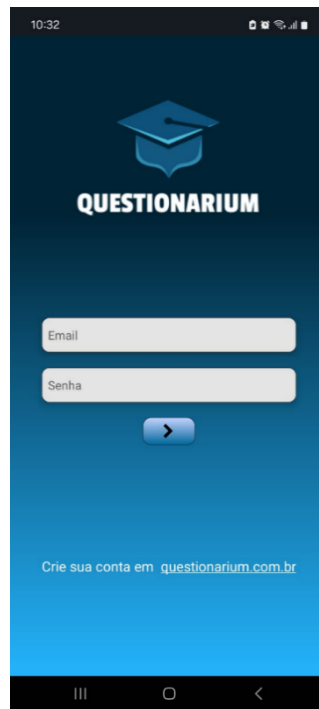
FONTE: O autor (2025).

A tela Criar/Editar cabeçalhos traz campos para preenchimento do usuário, como instituição, departamento etc, ilustrado na FIGURA 31. Além disso, o usuário pode salvar o cabeçalho ou cancelar a operação.

## 4.2 TELAS DA APLICAÇÃO MOBILE

A seção a seguir tem por objetivo mostrar as telas da aplicação e suas funcionalidades.

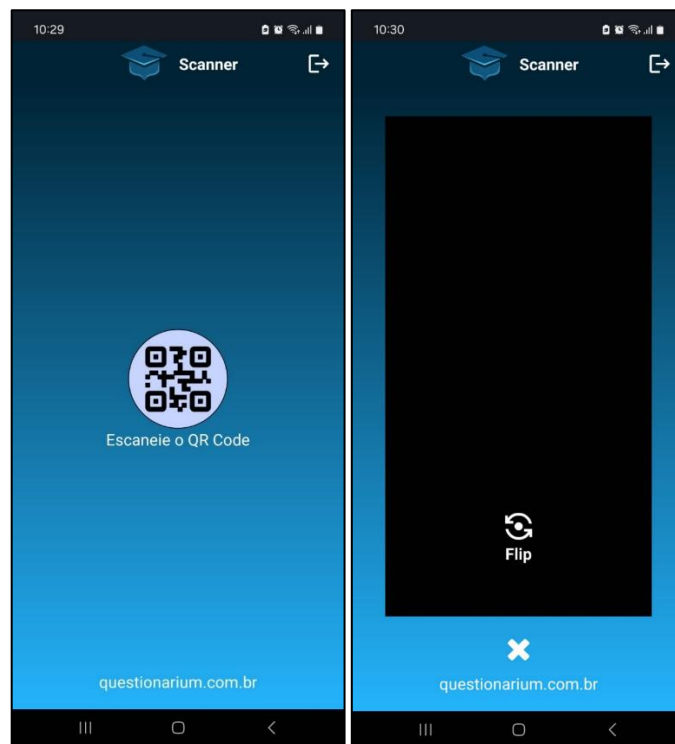
FIGURA 32 – TELA DE LOGIN



FONTE: O autor (2025).

A tela de login do aplicativo móvel (FIGURA 32) traz a funcionalidade de login, semelhante a aplicação web.

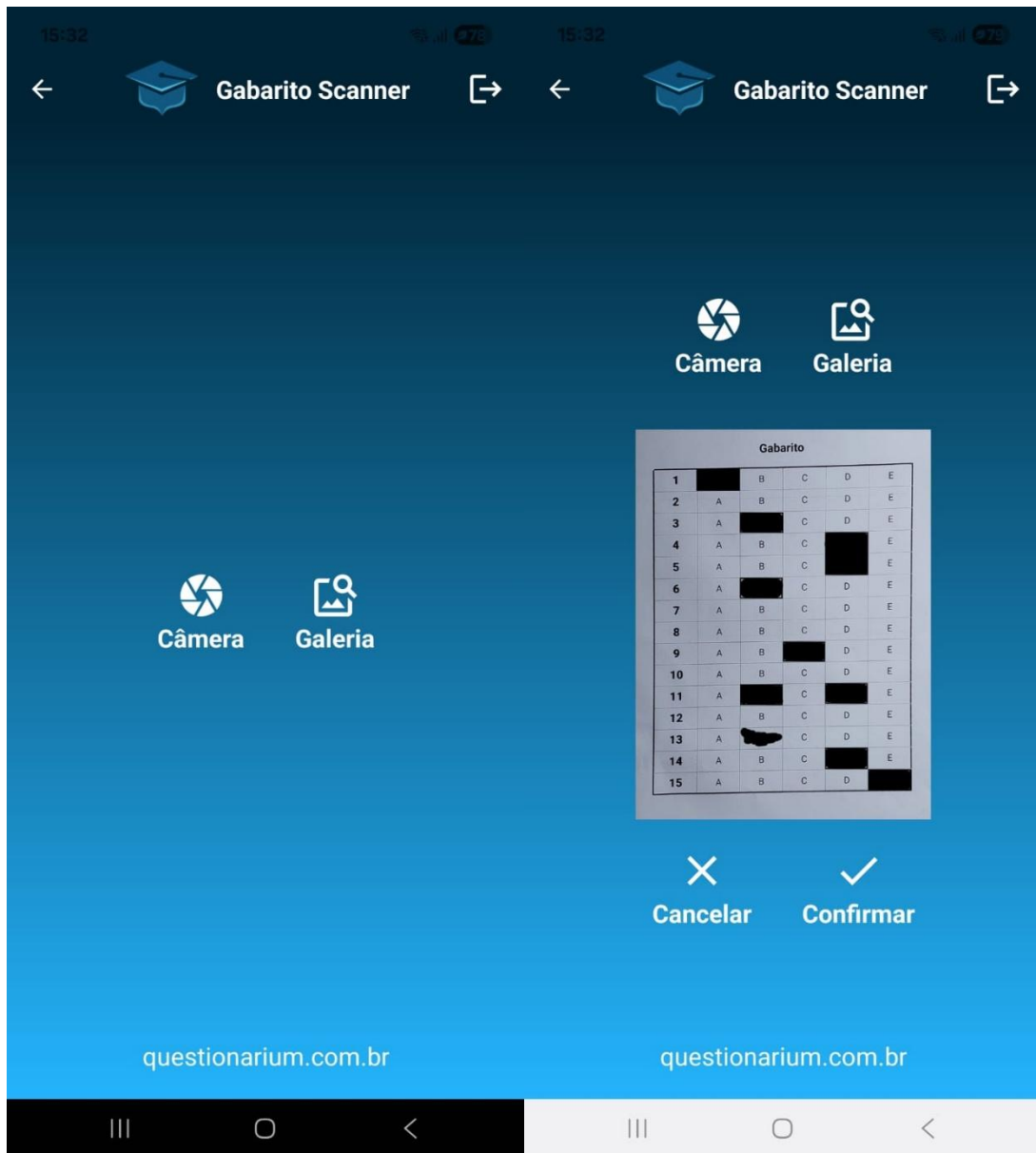
FIGURA 33 - TELA DE SCANNER



FONTE: O autor (2025).

A FIGURA 33 ilustra a tela responsável por escanear o QR Code a fim de identificar a aplicação de prova a ser corrigida. Ao pressionar o botão, a visualização de câmera é aberta.

FIGURA 34- TELA DE SELEÇÃO DE IMAGEM

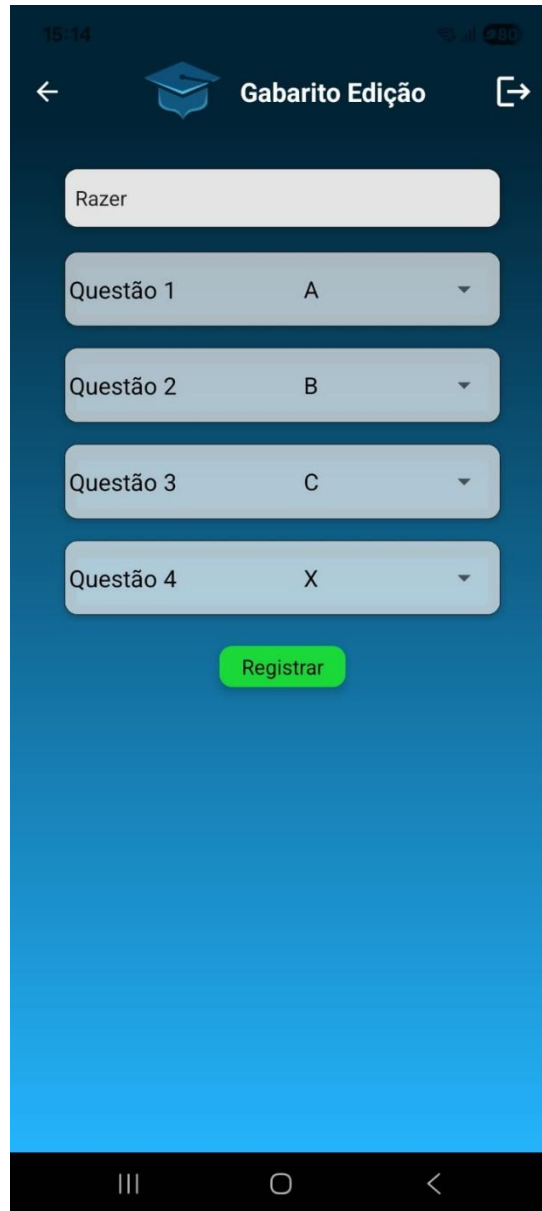


FONTE: O autor (2025).

Após a aplicação ser devidamente identificada, o aplicativo navega automaticamente para a tela de seleção de imagem, onde o usuário tem a sua disposição um botão para tirar uma foto ou selecionar uma imagem de sua galeria.

Após a seleção, é apresentado botão para cancelar ou confirmar a seleção (FIGURA 34).

FIGURA 35 - TELA DE EDIÇÃO DO GABARITO

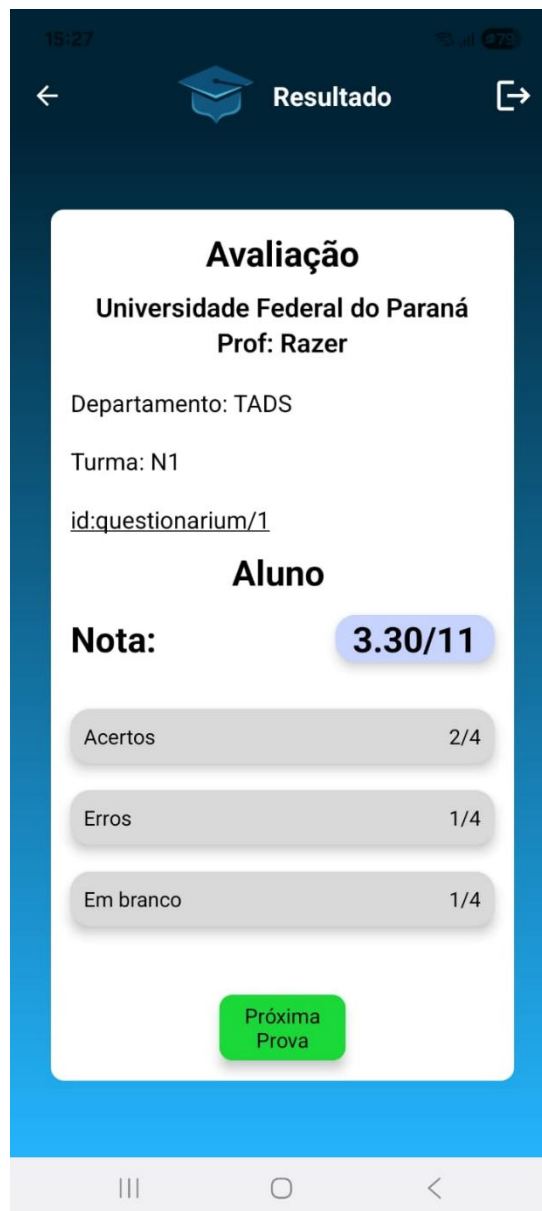


The screenshot displays a mobile application interface for editing an answer key. At the top, there is a navigation bar with a back arrow, a graduation cap icon, the title 'Gabarito Edição', and a share icon. Below the navigation bar, there is a text input field containing 'Razer'. Underneath, there are four rows, each representing a question and its selected answer: 'Questão 1' with 'A', 'Questão 2' with 'B', 'Questão 3' with 'C', and 'Questão 4' with 'X'. Each row has a dropdown arrow on the right. At the bottom of the form area, there is a green button labeled 'Registrar'. The background is a dark blue gradient. The bottom of the screen shows the Android navigation bar with three icons: a home button, a back button, and a recent apps button.

FONTE: O autor (2025).

O resultado da detecção do gabarito é apresentado na tela de edição, ilustrado pela FIGURA 35. Nesta tela, o usuário pode editar cada alternativa, selecionando uma das opções A, B, C, D, E ou X, além da identificação de prova, não sendo obrigatório. Para confirmar o resultado, o usuário pressiona o botão registrar que envia os dados ao servidor e navega para a próxima tela.

FIGURA 36 - TELA DE RESULTADO



FONTE: O autor (2025).

Finalizando o fluxo da aplicação móvel, a tela de resultado de prova (FIGURA 36), mostra o resultado obtido na avaliação escaneada, trazendo dados sobre a prova, nota, números de acertos, erros e questões em branco. O botão Próxima Prova navega para a tela de escanear o QR Code, reiniciando o fluxo.

## 5 CONSIDERAÇÕES FINAIS

As avaliações estão presentes em diferentes formas e cumprindo diferentes propósitos ao longo da vida. Especialmente no ensino, elas são ferramentas poderosas usadas por educadores para possibilitar que seus alunos atinjam os objetivos educacionais propostos. Todavia, ótimas avaliações são de difícil concepção, e todo o processo relacionado a elas é árduo e composto por vários obstáculos. Assim, um sistema de geração de provas que englobe as diversas atividades relacionadas a avaliações, desde criação à correção, pode ser de grande valia para auxiliar professores e profissionais de ensino em suas atividades avaliativas.

Para atingir o objetivo de criação de um software como o explicitado, se faz necessário o planejamento e modelagem, além de outras atividades de concepção de sistemas. Dessa forma, este documento evidencia os processos de fundamentação, planejamento, modelagem e arquitetura, através de revisão bibliográfica, aplicação de metodologia ágil, levantamento de requisitos (Apêndice A), diagramação (Apêndices A, C, D), prototipação de telas (Apêndice B) e demais tarefas. Essas atividades representam a primeira etapa deste trabalho, a qual estabelece a base para o desenvolvimento do software proposto e permitindo o alcance dos objetivos estabelecidos.

Ademais, uma das tarefas realizadas na primeira etapa deste trabalho foi o planejamento da segunda fase, que corresponde ao desenvolvimento do software. Para isso, foram planejadas Sprints com a alocação de histórias de usuário em cada uma delas, seguindo uma estrutura de entregas que priorizou, inicialmente, as funcionalidades mais importantes e complexas. Além disso, a arquitetura escolhida foi capaz de fornecer uma base sólida para o cumprimento dos requisitos funcionais e não funcionais, permitindo um desenvolvimento ágil e a constante evolução do sistema.

Na segunda fase do projeto, os objetivos inicialmente definidos foram revisados e ajustados, mas sem comprometer o escopo principal do trabalho. Esse aprofundamento permitiu as melhorias, tanto na parte teórica — envolvendo diagramas, casos de uso e modelagem — quanto na parte prática, com o desenvolvimento do sistema.

Apesar das alterações no escopo, os principais objetivos propostos foram alcançados. Ao final do projeto, o usuário é capaz de criar, armazenar e organizar suas questões e avaliações, sem se preocupar com a consistência entre as avaliações aplicadas. Além disso, o sistema possibilita a correção automatizada e a geração de relatórios das avaliações realizadas.

Dessa forma, espera-se que este sistema contribua efetivamente para a melhoria das condições de trabalho dos professores, reduzindo o tempo e a complexidade envolvidos na elaboração de avaliações, bem como a incidência de erros ao longo do processo.

## 5.1 ESTUDOS FUTUROS

Como continuidade deste trabalho, estudos futuros podem explorar a integração de técnicas de inteligência artificial, especialmente no campo da correção automática de questões discursivas e recomendação de conteúdo personalizado para questões. Além disso, propõe-se o desenvolvimento de uma nova aplicação voltada especificamente para provas online, com foco em responsividade e acessibilidade. Também é recomendada a ampliação das funcionalidades do sistema, incluindo maior diversidade de relatórios, melhorias nos parâmetros e/ou algoritmos de detecção de gabarito e novos cadastros de instituições, departamentos etc.

Outrossim, testes em ambientes reais de ensino, por exemplo a implantação em uma escola piloto, podem iluminar novas áreas de aprimoramento além de contribuir para a maturidade da aplicação e sua melhoria contínua.

## REFERÊNCIAS

ALVARENGA, Maria Isabel da Silva Azevedo; SOUZA, Maria Fátima de. **Professores, tarefas administrativas e seu impacto no trabalho docente**. Educação & Pesquisa, São Paulo, v. 38, n. 3, p. 663-678, 2012. Disponível em: <https://www.scielo.br/j/ep/a/CKkcb8c8Jt4wNwhvVsqSptm/>. Acesso em: 02 jun. 2024.

ANGULAR o framework da Google para quem procura eficiência. 2023. Disponível em: <https://www.dio.me/articles/angular-o-framework-da-google-para-quem-procura-eficiencia>. Acesso em: 10 maio 2024.

ASTAH. **Astah**. Disponível em: <https://astah.net/pt/>. Acesso em: 28 nov. 2024.

BARBOSA, Andreza; FERNANDES, Maria José da Silva; CUNHA, Renata Cristina Oliveira Barrichelo; AGUIAR, Thiago Borges de. **Tempo de trabalho e de ensino: composição da jornada de trabalho dos professores paulistas**. Educação & Sociedade, Campinas, v. 42, 2021. Disponível em: <<https://doi.org/10.1590/S1678-4634202147235807>>. Acesso em: 25 mar. 2024.

BEEDLE, Mike *et al.* **Manifesto para Desenvolvimento Ágil de Software**. Disponível em: Manifesto para Desenvolvimento Ágil de Software ([agilemanifesto.org](http://agilemanifesto.org)). Acesso em: 17 maio 2024.

BLACK, Paul; WILLIAM, Dylan. **Assessment and Classroom Learning**. Assessment in Education: Principles, Policy & Practice, v. 5, n. 1, p. 7-74, 1998. Disponível em: <http://dx.doi.org/10.1080/0969595980050102>. Acesso em: 30 abr. 2024. doi: 10.1080/0969595980050102

BOOCH G., RUMBAUGH J., JACOBSON I. **The unified modeling language user guide**. 2. ed. Massachusetts: Pearson Education, 2005.

BOURQUE, Pierre; FAIRLEY, Richard E. (Dick) (edit). **Guide to the Software Engineering Body of Knowledge**. New Jersey: IEE, 2014. Swebok.

BRASIL. **Decreto-Lei nº 5.452, de 1º de maio de 1943. Aprova a Consolidação das Leis do Trabalho (CLT)**. Diário Oficial da União: seção 1, Rio de Janeiro, p. 11937, 09 ago. 1943. Disponível em: <[http://www.planalto.gov.br/ccivil\\_03/decreto-lei/del5452.htm](http://www.planalto.gov.br/ccivil_03/decreto-lei/del5452.htm)>. Acesso em: 25 mar. 2024.

BRASIL. **Lei nº 9.394, de 20 de dezembro de 1996. Estabelece as diretrizes e bases da educação nacional**. Diário Oficial da União: seção 1, Brasília, DF, p. 27833, 23 dez. 1996. Disponível em: [http://www.planalto.gov.br/ccivil\\_03/leis/l9394.htm](http://www.planalto.gov.br/ccivil_03/leis/l9394.htm). Acesso em: 25 mar. 2024.

BRASIL. **Lei nº 11.738, de 16 de julho de 2008. Regulamenta o piso salarial profissional nacional para os profissionais do magistério público da educação básica**. Diário Oficial da União: seção 1, Brasília, DF, 17 jul. 2008. Disponível em: [https://www.planalto.gov.br/ccivil\\_03/\\_ato2007-2010/2008/lei/l11738.htm](https://www.planalto.gov.br/ccivil_03/_ato2007-2010/2008/lei/l11738.htm). Acesso em: 26 mar. 2024.

BRASIL. Câmara dos Deputados. **Proposição: PL 1135/2011**. Disponível em: <https://www.camara.leg.br/proposicoesWeb/fichadetramitacao?idProposicao=560047>. Acesso em: 02 jun. 2024.

BRASIL. Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira. **Resumo técnico: Censo Escolar da Educação Básica 2020**. Brasília, DF: INEP, 2021. Disponível em: [https://download.inep.gov.br/publicacoes/institucionais/estatisticas\\_e\\_indicadores/resumo\\_tecnico\\_censo\\_escolar\\_2020.pdf](https://download.inep.gov.br/publicacoes/institucionais/estatisticas_e_indicadores/resumo_tecnico_censo_escolar_2020.pdf). Acesso em: 02 jun. 2024.

BRITO, Filipe Silveira; SIMÕES, Silvia Regina Costa; LOPES, Anderson Ferreira. **Reestruturação do esporte universitário no Brasil: uma análise das políticas públicas no século XXI**. Movimento, Porto Alegre, v. 27, 2021. <Disponível em: <https://doi.org/10.22456/1982-8918.107646>>. Acesso em: 25 mar. 2024.

CASTRO, Maria Helena Guimarães de. **Sistemas de avaliação da educação no Brasil**. São Paulo em Perspectiva, São Paulo, v. 23, n. 1, p. 5-18, jan./jun. 2009.

CHAVES, Eduardo O. C. **A Tecnologia e a Educação**. Encyclopaedia of Philosophy of Education, Rio de Janeiro, ago. 1999. Disponível em: <https://smeduquedecaxias.rj.gov.br/nead/Biblioteca/Forma%C3%A7%C3%A3o%20Continuada/Tecnologia/chaves-tecnologia.pdf>. Acesso em: 6 maio 2024.

CHACON, Scott; STRAUB, Ben. **Pro Git**. 2. ed. Berkeley: Apress, 2014.

COHN, Mike. *User stories applied: for agile software development*. Boston: Addison-Wesley, 2004.

DRAWIO, 2024. Disponível em: <https://www.drawio.com/>. Acesso em: 10 maio 2024.

DOCKER INC. Compose Overview. [S.l.], 2024. Disponível em: <https://docs.docker.com/compose/>. Acesso em: 5 maio 2025.

DOCKER INC. *Docker Documentation*. [S.l.], 2024. Disponível em: <https://docs.docker.com/>. Acesso em: 5 maio 2025.

FERREIRA, Andrea de Lima; SANTOS, Eduardo. **A formação de professores na educação superior: desafios e perspectivas**. Ciência e Evidência, São Paulo, v. 10, n. 2, p. 123-137, 2020. Disponível em: <https://ojs.ifsp.edu.br/index.php/cienciaevidencia/article/view/2332/1437>. Acesso em: 06 jun. 2024.

FIELDING, Roy Thomas: **Architectural styles and the design of network-based software architectures**, 2000. Disponível em: [https://www.ics.uci.edu/~fielding/pubs/dissertation/rest\\_arch\\_style.htm](https://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm). Acesso em: 17 maio 2024

FIGMA, 2024. Disponível em: <https://www.figma.com/pt-br/>. Acesso em: 10 maio 2024.

FOWLER, Martin. ***Patterns of enterprise application architecture***. Boston: Addison-Wesley Professional, 2002. ISBN 978-0321127426.

FOWLER, Martin. **Patterns of Enterprise Application Architecture**. Bookman, 2009

FRANKL, Gabriele; NAPETSCHNIG; Sebastian, SCHARTNER Peter. **Pathways to Successful Online Testing: eExams with the “Secure Exam Environment” (SEE)**, p. 190-219. In: Technology Enhanced Assessment, 20th International Conference, TEA, Barcelona, Spain, October 5–6, 2017.

FREITAS, Sirley Leite, COSTA, Michele Gomes Noé da, MIRANDA, Flavine Assis de. **Avaliação Educacional: formas de uso na prática pedagógica**. Meta: Avaliação, Rio de Janeiro, v. 6, n. 16, p. 85-98, jan./abr. 2014. Disponível: <https://revistas.cesgranrio.org.br/index.php/metaavaliacao/article/view/217/pdf>. Acesso em: 29 abr. 2024

GAMMA, Erich; HELM, Richard; JOHNSON, Ralph; VLISSIDES, John. ***Design Patterns: Elements of Reusable Object-Oriented Software***. Addison-Wesley, 1994.

GATTI, B. A. **O professor e a avaliação em sala de aula: estudos em avaliação educacional**, 2003. Est. Aval. Educ. [online], n. 27, p. 97-114, 2003. Disponível em: [http://educa.fcc.org.br/scielo.php?pid=S0103-68312003000100004&script=sci\\_abstract](http://educa.fcc.org.br/scielo.php?pid=S0103-68312003000100004&script=sci_abstract). Acesso em: 29 abr. 2024

GITHUB, 2024. Disponível em: <https://github.com/>. Acesso em: 10 maio 2024.

HEIL, Joana; IFENTHALER, Dirk. Online Assessment in Higher Education: **A Systematic Review** March 2023, Online Learning 27(1). doi:10.24059/olj.v27i1.3398

IBM. *O que são contêineres?* [S.l.], 2024. Disponível em: <https://www.ibm.com/br-pt/topics/containers>. Acesso em: 5 maio 2025.

IEEE COMPUTER SOCIETY. **Guide to the Software Engineering Body of Knowledge (SWEBOK Guide)** Version 3.0. Los Alamitos, CA: IEEE Computer Society, 2014. Disponível em: <https://www.computer.org/education/bodies-of-knowledge/software-engineering>. Acesso em: 24 maio 2024.

LIBÂNEO, José Carlos. **Didática**. 2. ed. São Paulo: Cortez, 2013.

MICROSOFT CORPORATION. *Visual Studio Code Documentation*. [S.l.], 2024. Disponível em: <https://code.visualstudio.com/docs>. Acesso em: 5 maio 2025.

MONGODB INC. **MongoDB Manual**. 2024. Disponível em: <https://www.mongodb.com/docs/manual/>. Acesso em: 14 jun. 2025.

NÉSIO, Renata; FREITAS, Pedro F.; FREITAS, Gustavo. **Laboratório de Projeto - Concepção, Gestão e Implementação de Projetos Aplicados ao Ensino de**

**Engenharia de Sistemas.** 2020. Disponível em: [https://www.researchgate.net/figure/Figura-1-Ciclo-de-desenvolvimento-do-Scrum\\_fig1\\_347567574](https://www.researchgate.net/figure/Figura-1-Ciclo-de-desenvolvimento-do-Scrum_fig1_347567574). Acesso em: 30 maio 2025.

NICOL, David J., DICK, Debra Macfarlane. **Formative assessment and self-regulated learning: a model and seven principles of good feedback practice.** *Studies in Higher Education*, v. 31, n. 2, p. 199-218, 2007. Disponível em: <http://dx.doi.org/10.1080/03075070600572090>. Acesso em: 10 maio 2024.

ORACLE, 2024. Disponível em: <https://www.oracle.com/br/>. Acesso em: 10 maio 2024.

PACHECO, Roberto Carlos dos Santos; SANTOS, Neri dos; WAHRHAFTIG, Ramiro. **Transformação digital na Educação Superior: modos e impactos na universidade.** *Revista NUPEM, Campo Mourão*, v. 12, n. 27, p. 94-128, set./dez. 2020. Disponível em: <https://periodicos.unespar.edu.br/index.php/nupem/article/view/5631/3655>. Acesso em: 6 maio 2024.

PARANÁ. Secretaria da Educação. **Jornada de Trabalho com horas-atividade.** Secretaria da Educação do Paraná, [s.d.]. Disponível em: <https://www.educacao.pr.gov.br/Pagina/Jornada-de-Trabalho-com-horas-atividade>. Acesso em: 25 mar. 2024.

PEIXOTO, Joana; ARAÚJO, Cláudia Helena dos Santos. **Tecnologia e educação: algumas considerações sobre o discurso pedagógico contemporâneo.** Disponível em: <https://www.scielo.br/j/es/a/fKjYHb7qD8nK4MWQZFchr6K/?format=pdf&lang=pt>. Acesso em: 6 maio 2024.

PGADMIN. *pgAdmin 4 Documentation – Version 9.3.* [S.l.]: pgAdmin Development Team, 2024. Disponível em: <https://www.pgadmin.org/docs/pgadmin4/9.3/index.html>. Acesso em: 5 maio 2025.

POSTGRESQL, 2022. Disponível em: <https://www.postgresql.org/about/press/presskit15/pt/>. Acesso em: 10 maio 2024.

POSTMAN. *Postman API Platform Documentation.* [S.l.], 2024. Disponível em: <https://www.postman.com/docs/>. Acesso em: 5 maio 2025.

PRESSMAN, Roger S; MAXIM, Bruce R. **Engenharia de software: uma abordagem profissional [recurso eletrônico]**, tradução: Francisco Araújo da Costa Andrade, 9. ed. Porto Alegre: AMGH, 2021.E-pub.

PRUVO, 2024. Disponível em: [https://pruvo.ai/pt\\_br/](https://pruvo.ai/pt_br/). Acesso em: 12 maio 2024.

RABBITMQ. *RabbitMQ Documentation.* Disponível em: <https://www.rabbitmq.com/documentation.html>. Acesso em: 05 maio 2025.

REACT NATIVE, 2024. Disponível em: <https://reactnative.dev/>. Acesso em: 10 maio 2024.

RICHARDSON, Chris. *Microservices.io*. Disponível em: <https://microservices.io>. Acesso em: 28 maio 2025.

SIQUEIRA, Moisés Luiz Gomes, BITTENCOURT, Alexandre Horácio Couto, NOVAES, Ana Maria Pires, AVELAR, Kátia Eliane Santos. **Transformação digital e educação 4.0: cultura digital na educação básica**. Disponível em: <https://www.pensaracademico.unifacig.edu.br/index.php/pensaracademico/article/view/2754/2102>. Acesso em: 6 maio 2024.

SCHWAB, Klaus. **A quarta revolução industrial**. Trad. Daniel Moreira. São Paulo: Edipro, 2016. 159 p. ISBN 978-85-7283-978-0.

SCHWABER, Ken; SUTHERLAND, Jeff. **The Scrum Guide: the definitive guide to scrum: the rules of the game**, 2020. Disponível em: <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf>. Acesso em: 10 maio 2024.

SOMMERVILLE, Ian. **Engenharia de Software**. 9. ed. São Paulo: Pearson Prentice Hall, 2011.

SOMMERVILLE, Ian; SAWYER, Pete. **Requirements engineering: a good practice guide**. New York: Lancaster University, 1997.

SPRING AMQP. *Spring AMQP Reference Documentation*. Disponível em: <https://docs.spring.io/spring-amqp/reference/>. Acesso em: 05 maio 2025.

SPRING FRAMEWORK. Disponível em: <https://spring.io/projects/spring-framework>. Acesso em: 10 maio 2024.

STOLTERMAN, E.; FORS, A. C. (2004). **Information technology and the good life**. In: KAPLAN, Bonnie *et al.* (Eds.). *Information Systems Research: relevant theory and informed practice*. Boston: Springer.

UFPR. **Cipead**. Disponível em: <https://cipead.ufpr.br/index.php/saberesonline/ufpr-virtual>. Acesso em: 6 maio 2024.

VIEIRA, Jéssica Oliveira do Nascimento. **A prova como instrumento de avaliação da aprendizagem**. *Revista Científica Multidisciplinar Núcleo do Conhecimento*, ed. 10, v. 02, p. 112-125, out 2022. Disponível em: <https://www.nucleodoconhecimento.com.br/educacao/instrumento-de-avaliacao>. Acesso em: 29 abr. 2024. doi: 10.32749/nucleodoconhecimento.com.br/educacao/instrumento-de-avaliacao

VIEGAS, Moacir Fernando. **Trabalhando todo o tempo: sobrecarga e intensificação no trabalho de professoras da educação básica**. *Educação e Pesquisa*, v. 48, e244193, 2022. Disponível em: <https://doi.org/10.1590/S1678-4634202248244193>. Acesso em: 3 maio 2024.

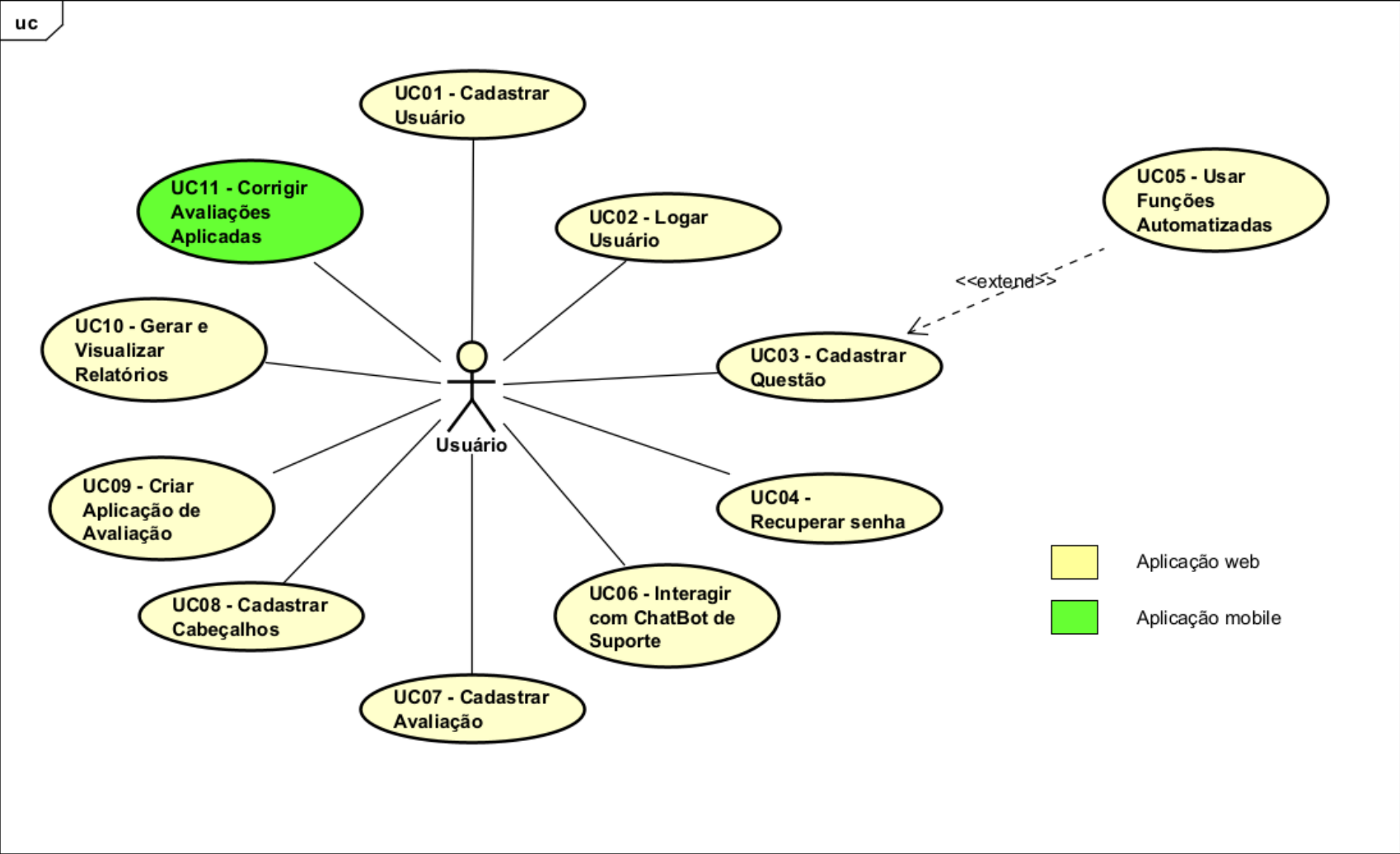
Washizaki, H. **Guide to the Software Engineering Body of Knowledge (SWEBOK Guide)**, Version 4.0, IEEE Computer Society, 2024; [www.swebok.org](http://www.swebok.org).

WILLIAMSON, D. M., Xi, N.; BREYER, F. J. **Assessing computer-mediated peer review in a college composition course**. *Computers & Composition*, v. 54,102516, 2019.

XP INC. **Microserviço: vantagens e desvantagens**. Medium, 24 mar. 2020. Disponível em: <https://medium.com/xp-inc/microservi%C3%A7o-vantagens-e-desvantagens-c89f227ef1f9>. Acesso em: 28 maio 2025.

## **APÊNDICES**

APÊNDICE A – DIAGRAMA DE CASOS DE USO



## APÊNDICE B – HISTÓRIAS DE USUÁRIO E PROTOTIPAÇÃO

### HU01 Cadastrar Usuário

Sendo: Usuário

Quero: Cadastrar e manter meu perfil

Para: Ter acesso às funcionalidades do sistema

Descrição:

Como usuário da aplicação, deseja-se cadastrar seu perfil no sistema para usar suas funcionalidades através do fornecimento de seus dados e confirmação do e-mail. Após realizar o cadastro, gostaria de ter a capacidade de atualizar suas informações e de excluir seu registro, se necessário.

Critérios de Aceitação:

- O sistema deve fornecer um formulário de cadastro onde o usuário possa inserir suas informações pessoais, como nome, e-mail e senha.
- O formulário de cadastro deve validar os campos obrigatórios e formatos de entrada adequados.
- Após o preenchimento correto do formulário, o sistema deve armazenar as informações do usuário no banco de dados de forma segura.
- O sistema deve enviar uma confirmação de cadastro para o e-mail fornecido pelo usuário.
- O usuário deve ser capaz de acessar as funcionalidades do sistema após o cadastro bem-sucedido.
- O sistema deve permitir que o usuário atualize suas informações de perfil, como nome e senha, posteriormente, se necessário.

### Cr terios de Aceita o Detalhamento:

1. O sistema deve fornecer formul rio de cadastro e verificar os dados fornecidos pelo usu rio
  - DADO QUE o usu rio acessa a tela de cadastro
  - QUANDO ele insere seu e-mail, nome e senha nos campos designados
  - ENT O o sistema deve verificar se o formato do e-mail e seu conte do s o v lidos fornecendo mensagens de erro apropriadas.
2. O sistema deve enviar uma confirma o de cadastro para o e-mail fornecido pelo usu rio
  - DADO QUE o usu rio inseriu corretamente seus dados no formul rio
  - QUANDO ele clica no bot o ‘Cadastrar’
  - ENT O o sistema deve enviar um e-mail de verifica o com o link de confirma o do cadastro.
3. Ap s o preenchimento correto do formul rio, o sistema deve armazenar as informa es do usu rio no banco de dados de forma segura
  - DADO QUE o usu rio confirma seu cadastro
  - QUANDO ele   redirecionado para tela de login
  - ENT O o sistema deve armazenar os dados e fornecer uma confirma o de sucesso ou falha do cadastro.

FIGURA – Tela de cadastro V1

QUESTIONARIUM

Cadastro

Nome\*

Email\*

Senha\*

Cadastrar

[Fa a login aqui](#)

suporte@questionarium.com Questionarium   2024

FONTE: O autor (2025).

FIGURA – Tela de cadastro V2

**QUESTIONARIUM** | Questões | Avaliações | Turmas | Sobre | Olá, Razer

Perfil Público  
Conta  
Alterar Senha

### Conta

Nome  Sobrenome  Email

Telefone

Trabalho

Instituição  Cargo  Tipo de conta  
Plano 1

Salvar

suporte@questionarium.com | Questionarium © 2024 | Instagram | Facebook | Telegram

FONTE: O autor (2025).

FIGURA – Tela de cadastro V3

**QUESTIONARIUM** | Questões | Avaliações | Turmas | Sobre | Olá, Razer

Perfil Público  
Conta  
Alterar Senha

### Conta

Nome  Sobrenome  Email

Telefone  CPF  RG

Endereço

CEP  Rua  Número

Estado  Bairro  Complemento

Trabalho

Instituição  Cargo  Tipo de conta  
Selecione uma opção ▼

Salvar

suporte@questionarium.com | Questionarium © 2024 | Instagram | Facebook | Telegram

FONTE: O autor (2025).

FIGURA – Tela de manutenção de cadastro

**QUESTIONARIUM**

Questões | Avaliações | Turmas | Sobre

Olá, Razer

Perfil Público  
Conta  
Alterar Senha

## Perfil

Nome

Email Público

Bio

Foto do perfil

Editar

Salvar

suporte@questionarium.com | Questionarium © 2024

FONTE: O autor (2025).

FIGURA – Tela de autenticação de cadastro

**QUESTIONARIUM**

Questões | Avaliações | Turmas | Sobre

Olá, Razer

Perfil Público  
Conta  
Alterar Senha

## Autenticação

Senha antiga

Nova senha

Confirma senha

Atualizar

Autenticação em dois fatores

**A autenticação de dois fatores ainda não está habilitada.**

A autenticação de dois fatores adiciona uma camada adicional de segurança à sua conta, exigindo mais do que apenas uma senha para fazer login.

Habilitar autenticação em dois fatores

Salvar

suporte@questionarium.com | Questionarium © 2024

FONTE: O autor (2025).

## HU02 Logar Usuário

Sendo: Sendo um usuário registrado no sistema.

Quero: Quero fazer login na minha conta.

Para: Para poder aproveitar as funcionalidades disponíveis no sistema e acessar meu perfil.

Descrição:

Os usuários da aplicação precisam ser capazes de fazer login em suas contas para acessar as funcionalidades e informações personalizadas oferecidas pelo sistema.

Critérios de Aceitação:

- O sistema deve fornecer um formulário de login onde o usuário possa inserir seu e-mail e senha.
- O formulário de login deve validar os campos obrigatórios e formatos de entrada adequados.
- Após o preenchimento correto do formulário, o sistema deve autenticar o usuário com base nas informações fornecidas.
- O sistema deve redirecionar o usuário para a página inicial.
- O sistema deve exibir uma mensagem de erro caso o e-mail ou senha fornecidos estejam incorretos.
- O sistema deve fornecer opções para recuperação de senha em caso de esquecimento.

Critérios de Aceitação Detalhamento:

1. O sistema deve fornecer um formulário de login que valide os campos obrigatórios e formatos dos dados de entrada
  - DADO QUE o usuário acessa a tela de login
  - QUANDO ele insere sua senha e e-mail no formulário de login
  - ENTÃO o sistema deve validar o formato e conteúdo do e-mail e senha fornecidos.
2. Após o preenchimento correto do formulário, o sistema deve autenticar o usuário com base nas informações fornecidas

- DADO QUE o usuário realiza o preenchimento correto dos dados no formulário de login
  - QUANDO ele pressiona o botão “Login”
  - ENTÃO o sistema deve autenticar o usuário validando-os em conjunto com os dados presentes no banco de dados.
3. O sistema deve exibir uma mensagem de erro caso o e-mail ou senha fornecidos estejam incorretos
- DADO QUE o usuário insere uma senha incorreta ou um e-mail não cadastrado
  - QUANDO ele realiza uma tentativa de autenticação
  - ENTÃO o sistema deve fornecer uma mensagem de erro apropriada indicando as alterações necessárias no formulário de login.
4. O sistema deve fornecer opções para recuperação de senha em caso de esquecimento
- DADO QUE o usuário deseje recuperar sua senha
  - QUANDO ele selecionar a opção “recuperar minha senha”
  - ENTÃO o sistema deve mostrar a tela de inserção de e-mail para recuperação, o qual irá direcioná-lo para a tela de recuperação de senha.

FIGURA – Tela de login

The image shows a login page for 'QUESTIONARIUM'. At the top, there is a dark blue header with a graduation cap icon and the word 'QUESTIONARIUM' in white. Below the header, the word 'Login' is centered. There are two input fields: 'Email:\*' and 'Senha\*'. Below the 'Senha\*' field is a blue button labeled 'Entrar'. Underneath the button is a link that says 'Crie sua conta aqui'. At the bottom of the page, there is a dark blue footer with the email 'suporte@questionarium.com', the text 'Questionarium © 2024', and three social media icons (Instagram, Facebook, and Telegram).

FONTE: O autor (2025).

### HU03 Cadastrar Questão

Sendo: Sendo um usuário registrado e logado no sistema.

Quero: Criar, ler, atualizar e excluir questões

Para: Preparar e gerenciar o banco de questões do sistema.

Descrição:

Os usuários registrados no sistema devem ter a capacidade de realizar operações básicas de CRUD (criar, ler, atualizar e excluir) sobre as questões. Isso permitirá que eles preparem e gerenciem o banco de questões.

Critérios de Aceitação:

- O sistema deve fornecer uma interface para criar questões, permitindo que o usuário insira todos os campos necessários, como enunciado, tipo de questão, alternativas múltiplas escolha, respostas corretas e erradas e categorias da questão.
- O usuário deve ser capaz de visualizar todas as questões existentes no banco de questões, com opções para filtrar por categorias e pesquisar questões específicas.
- O sistema deve permitir que o usuário atualize as informações de uma questão existente no seu banco pessoal, alterando seu enunciado, tipo de questão, alternativas, resposta correta e categorias.
- O usuário deve ter a capacidade de excluir uma questão do seu banco pessoal de questões, garantindo que a exclusão seja confirmada para evitar exclusões acidentais.
- Deve ser possível definir um limite de linhas para a resposta de questões discursivas.

Critérios de Aceitação Detalhamento:

1. O sistema deve fornecer uma interface para criar questões, permitindo que o usuário insira todos os campos necessários, como enunciado, tipo de questão, alternativas múltiplas escolha, respostas corretas e erradas, categorias da questão, dentre outras
- DADO QUE: o usuário acessa a tela de criação de questões

- QUANDO: ele criar uma questão e fornece os dados necessários
  - ENTÃO: o sistema deve armazenar de forma segura a questão no banco de dados e informar ao usuário o resultado da criação.
2. O usuário deve ser capaz de visualizar todas as questões existentes no banco de questões, com opções para filtrar por categorias e pesquisar questões específicas
- DADO QUE: o usuário acessa a tela de visualização de questões
  - QUANDO: ele escolhe os filtros e insere o texto de pesquisa
  - ENTÃO: o sistema deve atualizar as questões em visualização de acordo com os parâmetros fornecidos pelo usuário.
3. O sistema deve permitir que o usuário atualize as informações de uma questão existente no seu banco pessoal, alterando seu enunciado, tipo de questão, alternativas, resposta correta e categorias
- DADO QUE: o usuário escolhe uma questão específica
  - QUANDO: ele altera os dados da questão e pressiona o botão “Salvar”
  - ENTÃO: o sistema deve alterar os dados da questão e mostrar a questão atualizada para o usuário.
4. O usuário deve ter a capacidade de excluir uma questão do seu banco pessoal de questões, garantindo que a exclusão seja confirmada para evitar exclusões acidentais.
- DADO QUE: o usuário acessa a lista de questões
  - QUANDO: ele seleciona a opção de exclusão da questão criada e confirma a exclusão
  - ENTÃO: o sistema deve excluir a questão escolhida e garantir que não existam avaliações referenciando aquela questão.
5. Deve ser possível definir um limite de linhas para a resposta de questões discursivas
- DADO QUE: o usuário acessa a tela de criação de questões
  - QUANDO: ele define um limite de linhas para uma questão discursiva
  - ENTÃO: o sistema atualiza o limite de linhas para essa questão e utiliza o valor definido como padrão para as próximas questões criadas.

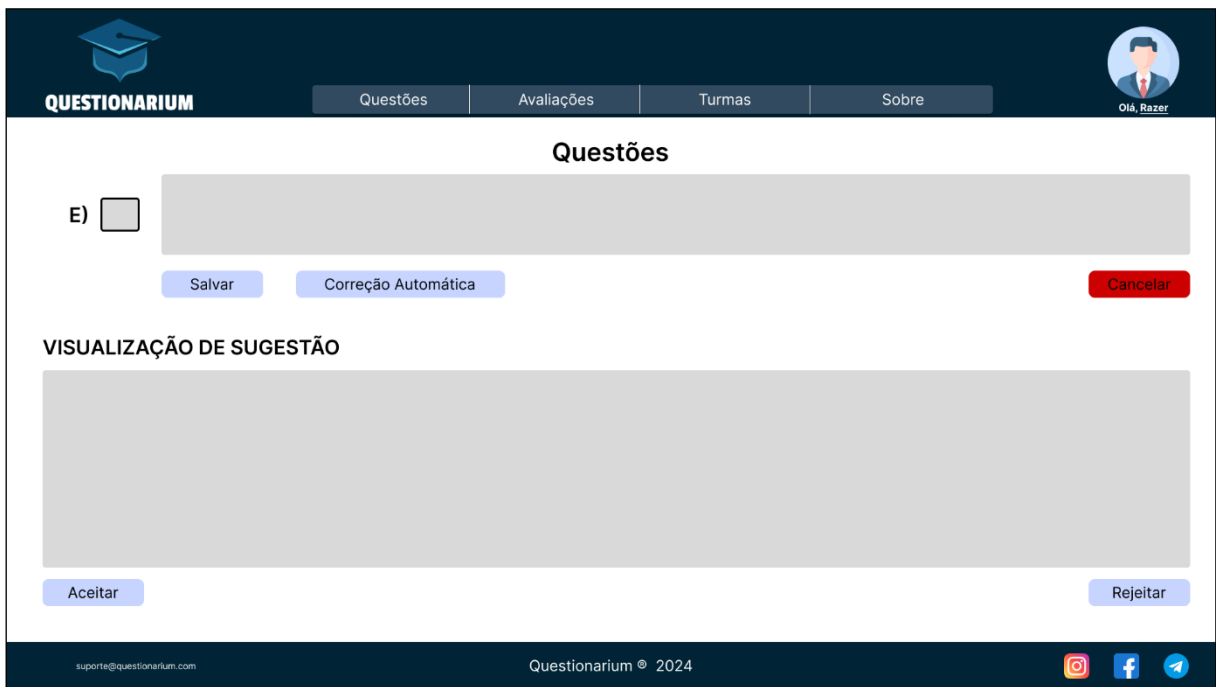
FIGURA – Tela 1 de cadastro de questão

FONTE: O autor (2025).

FIGURA – Tela 2 de cadastro de questão

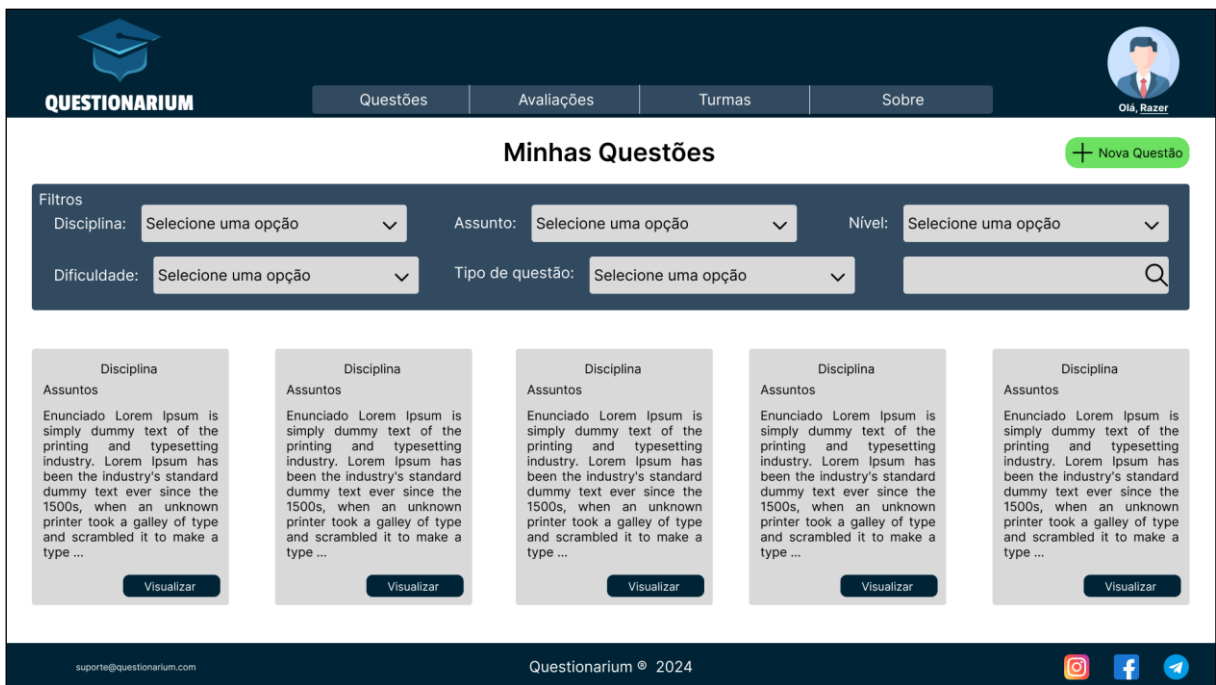
FONTE: O autor (2025).

FIGURA – Tela 3 de cadastro de questão



FONTE: O autor (2025).

FIGURA – Tela de manutenção de questão



FONTE: O autor (2025).

## HU04 Recuperar senha

Sendo: Sendo um usuário logado.

Quero: Recuperar minha senha.

Para: Garantir o meu acesso ao sistema.

Descrição:

O usuário deve ser capaz de recuperar sua senha em caso de esquecimento.

Critérios de Aceitação:

- O sistema deve enviar um e-mail com link para recuperação de senha
- Os campos preenchidos devem ser validados pelo sistema
- O sistema deve informar ao usuário sobre a atualização por e-mail.

Critérios de Aceitação Detalhamento:

1. O sistema deve enviar um e-mail com link para recuperação de senha
  - DADO QUE o usuário pressiona o botão de Esqueci a senha e informa o seu e-mail para recuperação de senha
  - QUANDO ele pressiona o botão “Enviar”
  - ENTÃO o sistema envia um e-mail para o endereço inserido que redirecionará o usuário para tela de recuperação de senha
2. Os campos preenchidos devem ser validados pelo sistema
  - DADO QUE o usuário acessa a tela de recuperação de senha
  - QUANDO ele insere sua nova senha e confirmação de nova senha
  - ENTÃO o sistema deve validar o formato e conteúdo da nova senha e confirmação de nova senha.
3. O sistema deve informar ao usuário o resultado da alteração.
  - DADO QUE o usuário insere seus dados no formulário de atualização de senha
  - QUANDO ele pressiona o botão “Atualizar”
  - ENTÃO o sistema deve enviar uma mensagem para o e-mail cadastrado alertando sobre a alteração.

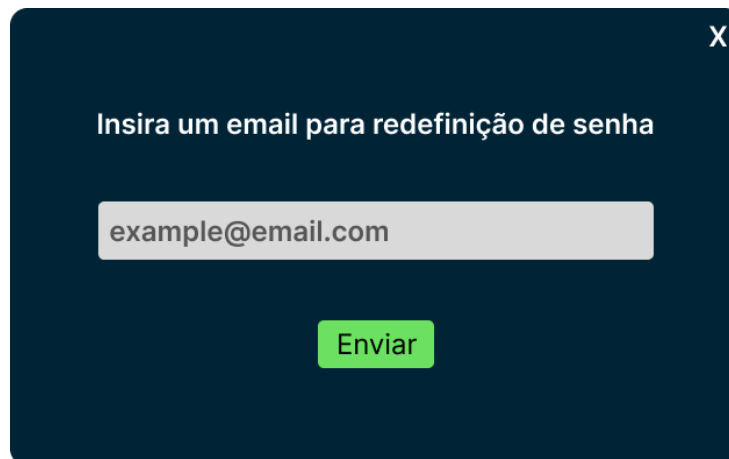
FIGURA – Tela de redefinição de senha



The screenshot shows the 'QUESTIONARIUM' interface. At the top, there is a navigation bar with a logo on the left and a user profile on the right. The main content area is titled 'Redefinição de senha'. It contains two input fields: 'Nova Senha\*' and 'Confirmar a senha\*'. Below these fields is a blue 'Salvar' button. The footer includes the email 'suporte@questionarium.com', the text 'Questionarium © 2024', and social media icons for Instagram, Facebook, and Telegram.

FONTE: O autor (2025).

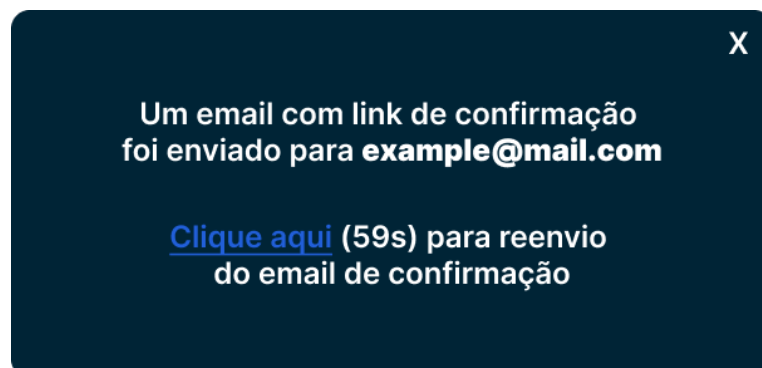
FIGURA – Popup 1 de redefinição de senha



The popup has a dark blue background and a close button (X) in the top right corner. The text reads 'Insira um email para redefinição de senha'. Below this is an input field containing the email address 'example@email.com'. At the bottom center is a green 'Enviar' button.

FONTE: O autor (2025).

FIGURA – Popup 2 de redefinição de senha



The popup has a dark blue background and a close button (X) in the top right corner. The text reads 'Um email com link de confirmação foi enviado para **example@mail.com**'. Below this is a blue link: 'Clique aqui (59s) para reenvio do email de confirmação'.

FONTE: O autor (2025).

## HU05 Usar Funções Automatizadas

Sendo: Sendo um usuário registrado e logado no sistema.

Quero: Utilizar das ferramentas de automatização, como correção gramatical, sugestão de criação de questão.

Para: Agilizar o processo de criação de questões.

Descrição:

Como usuário registrado e logado no sistema, deseja-se aproveitar as ferramentas de automatização disponíveis para agilizar tarefas relacionadas à criação de questões. Isso inclui a correção gramatical automática de questões e sugestão de criação de questões com base em determinados critérios ou temas, automatizada por IA.

Critérios de Aceitação:

- Durante a criação ou manutenção de uma questão, o sistema deve oferecer uma funcionalidade de sugestão de correções gramaticais nas questões de forma automatizada.
- O sistema deve ter uma opção para sugerir uma criação de questão a partir de parâmetros estabelecidos pelo usuário.
- Deve haver uma maneira de os usuários revisarem e confirmarem as correções sugeridas pelo sistema antes de aplicá-las.

Critérios de Aceitação Detalhamento:

1. Durante a criação ou manutenção de uma questão, o sistema deve oferecer uma funcionalidade de sugestão de correções gramaticais nas questões de forma automatizada.
  - DADO QUE o usuário esteja criando ou editando uma questão
  - QUANDO ele clica o botão ou link de Correção Gramatical
  - ENTÃO o sistema corrige automaticamente os textos da questão
2. O sistema deve ter uma opção para sugerir uma criação de questão a partir de parâmetros estabelecidos pelo usuário.
  - DADO QUE o usuário esteja criando ou editando uma questão

- QUANDO ele clica no botão ou link de Gerar Automaticamente
  - ENTÃO o sistema deve apresentar uma tela de criação automática de questão com parâmetros definido pelo usuário
3. Deve haver uma maneira de os usuários revisarem e confirmarem as correções sugeridas pelo sistema antes de aplicá-las.
- DADO QUE o usuário corrigiu os textos de sua questão automaticamente
  - QUANDO ele pressiona o botão “Reverter”
  - ENTÃO o sistema deve reverter todas as alterações para o texto original

FIGURA – Tela com exemplo de botão com função automatizada

The screenshot displays the 'Questionnaire' system interface. At the top, there is a dark blue header with the 'QUESTIONARIUM' logo on the left and a user profile icon on the right labeled 'Olá, Razer'. Below the header is a navigation menu with tabs for 'Questões', 'Avaliações', 'Turmas', and 'Sobre'. The main content area is titled 'Questões' and contains a form with the following fields:

- Tipo de questão:** Seleccione uma opção (dropdown menu)
- Nível de Ensino:** Seleccione uma opção (dropdown menu)
- Dificuldade da Questão:** Radio buttons for 1 (Fácil), 2, 3 (Médio), 4, and 5 (Difícil)
- Disciplina(s):** Text input field
- Assunto(s):** Text input field

A 'Gerar Automaticamente' button is located to the right of the form fields. Below the form is a large grey rectangular area labeled 'ENUNCIADO:'. At the bottom of the interface, there is a footer with the email 'suporte@questionarium.com', the text 'Questionarium © 2024', and social media icons for Instagram, Facebook, and Telegram. The main question area shows five multiple-choice options (A) through (E), each with a radio button and a greyed-out text input field. At the bottom of this area are three buttons: 'Salvar' (blue), 'Correção Automática' (blue), and 'Cancelar' (red).

FONTE: O autor (2025).

## HU06 Interagir com ChatBot de Suporte

Sendo: Sendo um usuário registrado e logado no sistema.

Quero: Utilizar o ChatBot para receber auxílio na utilização do sistema.

Para: Facilitar o uso das funcionalidades da aplicação.

Descrição:

O ChatBot servirá como ferramenta de suporte fornecendo dados e informações sobre como utilizar o sistema. Fornecerá dados sobre criação de questões, provas, como manter e atualizar informações, dentre outras informações. A interação com o ChatBot se dará através de uma “conversa” fornecendo perguntas em forma de texto. Para cada pergunta o ChatBot, que utiliza de IA, irá retornar uma resposta apropriada e irá guiar o usuário.

Critérios de Aceitação:

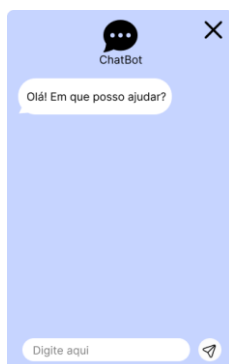
- O sistema deve integrar o ChatBot com uma interface para interação.
- O ChatBot deve oferecer suporte técnico para resolver problemas ou explicar funções da plataforma.
- Em casos em que o ChatBot não saiba responder uma pergunta, ele fornecerá uma forma de contato com o suporte.

Critérios de Aceitação Detalhamento:

1. O sistema deve integrar o ChatBot com uma interface para interação.
  - DADO QUE: o usuário está na tela inicial
  - QUANDO: ele pressiona o ícone do ChatBot
  - ENTÃO: o sistema deve acionar o ChatBot e permitir a interação
2. O ChatBot deve oferecer suporte técnico para resolver problemas ou explicar funções da plataforma.
  - DADO QUE: o usuário acessa o ChatBot
  - QUANDO: ele realiza uma pergunta
  - ENTÃO: o sistema deve acessar o banco de perguntas e soluções disponíveis e fornecer uma resposta apropriada ao usuário

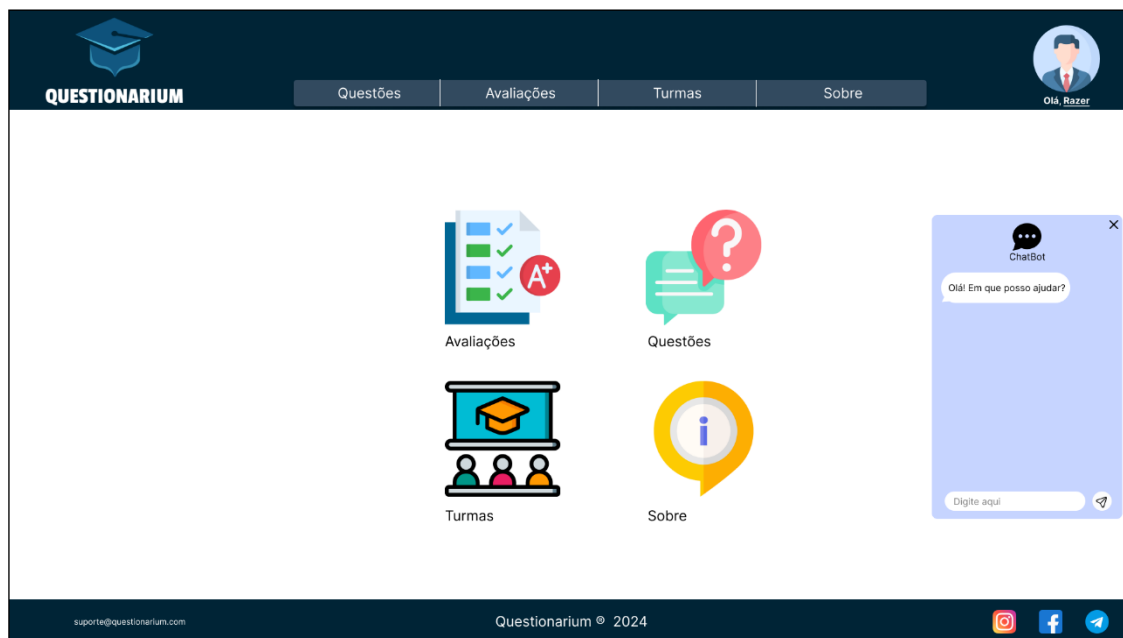
3. Em casos em que o ChatBot não saiba responder uma pergunta, ele fornecerá uma forma de contato com o suporte.
- DADO QUE: o usuário realiza uma pergunta ao ChatBot
  - QUANDO: ele não é capaz de encontrar uma resposta adequada
  - ENTÃO: ele deve enviar o contato de suporte da plataforma

FIGURA 50 – Modal do ChatBot



FONTE: O autor (2025).

FIGURA – Tela com exemplo de ChatBot



FONTE: O autor (2025).

## HU07 Cadastrar Avaliação

Sendo: Sendo um usuário registrado e logado no sistema.

Quero: Criar avaliações personalizadas.

Para: Usá-la em uma ou mais aplicações de avaliações.

Descrição:

O professor poderá cadastrar, alterar e excluir uma avaliação no sistema. Ao criá-la, o usuário irá escolher as questões com base no banco de dados de questões privadas e públicas e irá selecionar um modelo de cabeçalho previamente cadastrado.

Critérios de Aceitação:

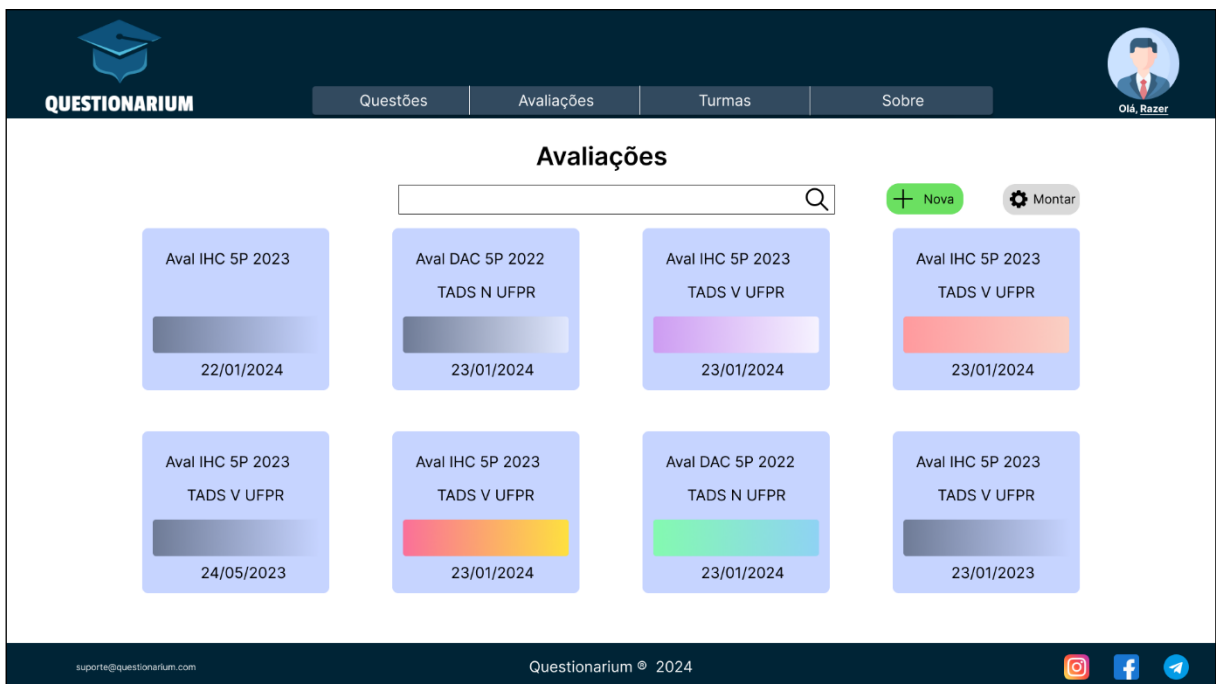
- O sistema deve fornecer uma interface para o professor criar uma avaliação.
- O professor deve ser capaz de selecionar as questões para a avaliação a partir do banco de dados de questões privadas e públicas.
- Deve haver a opção de escolher um modelo de cabeçalho previamente cadastrado para a avaliação.
- Após a criação, o professor deve ser capaz de revisar e, se necessário, editar a avaliação antes de finalizá-la.
- Deve ser possível editar e excluir avaliações já criadas.

Critérios de Aceitação Detalhamento:

1. O sistema deve fornecer uma interface para o professor criar uma avaliação.
  - DADO QUE: o usuário acessa a tela de listagem de avaliações
  - QUANDO: ele pressiona o botão “Novo”
  - ENTÃO: o sistema dele redirecionar o usuário para a tela de criação de avaliações.
2. O professor deve ser capaz de selecionar as questões para a avaliação a partir do banco de dados de questões privadas e públicas
  - DADO QUE: o usuário está na tela de criação de avaliações
  - QUANDO: ele escolhe adicionar uma nova questão a avaliação
  - ENTÃO: o sistema deve abrir uma tela com a lista de questões criadas pelo usuário e questões públicas disponíveis.

3. Deve haver a opção de escolher um modelo de cabeçalho previamente cadastrado para a avaliação
  - DADO QUE: o usuário está na tela de criação de avaliações
  - QUANDO: ele seleciona a opção “Carregar Cabeçalho”
  - ENTÃO: o sistema deve abrir uma tela com a lista de cabeçalhos previamente cadastrados para seleção.
4. Após a criação, o professor deve ser capaz de revisar e, se necessário, editar a avaliação antes de finalizá-la
  - DADO QUE: o usuário finalizou a inserção de questões
  - QUANDO: ele seleciona a opção “Salvar”
  - ENTÃO: o sistema deve colocar o usuário em modo de visualização e pedir a confirmação do usuário.

FIGURA – Tela de manutenção de avaliações



FONTE: O autor (2025).

FIGURA – Tela de cadastro de avaliação

**QUESTIONARIUM** Questões Avaliações Turmas Relatórios Sobre Olá, Razer

## Avaliação

**Cabeçalho**

Instituição  Curso  Instruções

Disciplina  Professor

Salvar Cabeçalho Carregar Cabeçalho

**Questão 01**  
 Instituição: Instituição xx Enunciado: "Lorem ipsum dolor sit amet, consectetur a... Pública

**Questão 02**  
 Instituição: Instituição xx Enunciado: "Lorem ipsum dolor sit amet, consectetur a... Pública

suporte@questionarium.com Questionarium © 2024

**Questão 03**  
 Instituição: Instituição xx Enunciado: "Lorem ipsum dolor sit amet, consectetur a... Privado

+ Nova Questão

[Crie sua questão aqui](#)

Turma  Cancelar Salvar

FONTE: O autor (2025).

FIGURA – Modal de criação rápida de cabeçalho

### Criar Cabeçalho

Instituição:

Disciplina:

Curso:

Professor:

+ Novo Campo

Instruções:

Salvar Cancelar

FONTE: O autor (2025).

FIGURA – Modal pesquisa de cabeçalhos prontos

## Cabeçalhos

Campo: Número ▾

Filtrar 

Número	Disciplina	Curso	Observação	Data Criação	
00001	5P2023P1-0001	Curso1	Sed ut perspica...	17/06/2024	
00002	5P2023P1-0002	Curso2	Sed ut perspica...	17/06/2024	
00003	5P2023P1-0004	Curso3	Sed ut perspica...	17/06/2024	
00004	5P2023P1-0006	Curso4	Sed ut perspica...	17/06/2024	
00005	5P2023P1-0007	Curso5	Sed ut perspica...	17/06/2024	
00006	5P2023P1-0008	Curso6	Sed ut perspica...	17/06/2024	

**Cancelar**

FONTE: O autor (2025).

FIGURA – Modal de filtro dos cabeçalhos

✕

Instituição:

UFPR

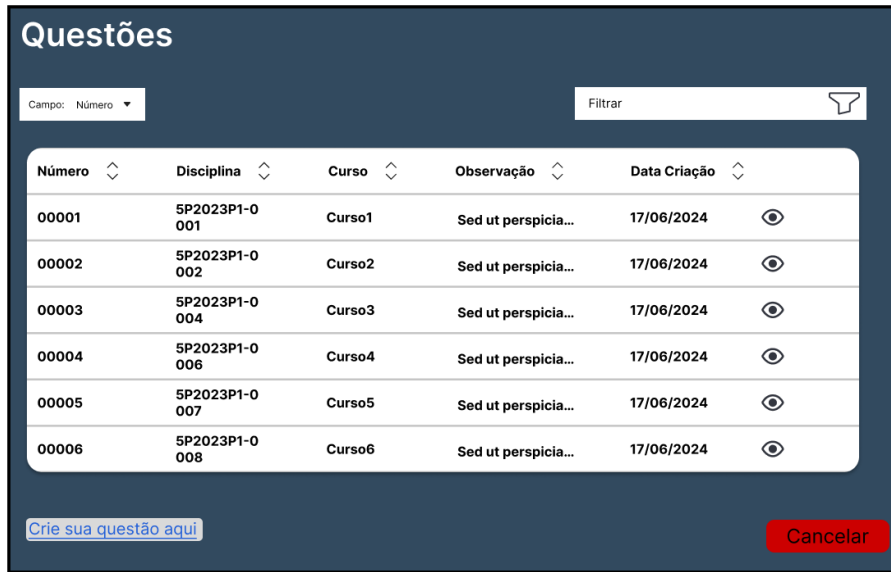
Curso:

Disciplina:

**Filtrar**

FONTE: O autor (2025).

FIGURA - Modal de pesquisa de questões



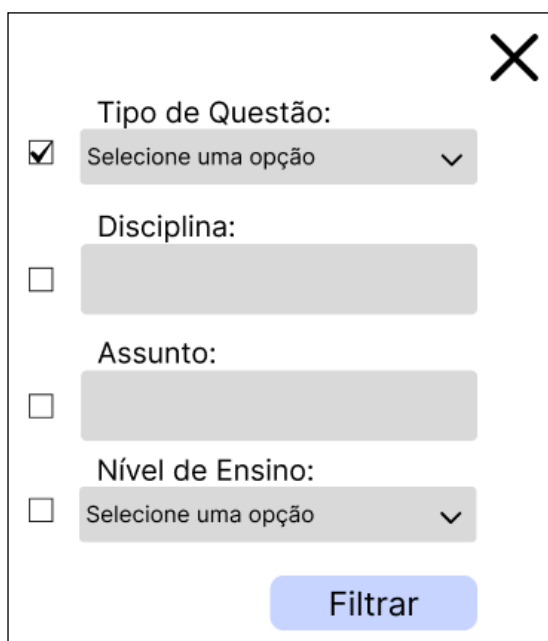
FONTE: O autor (2025).

FIGURA – Modal de visualização de questão



FONTE: O autor (2025).

FIGURA – Modal de filtro de questões



Modal de filtro de questões com um ícone de fechar (X) no canto superior direito. O formulário contém:

- Tipo de Questão:** Um campo de seleção com uma caixa de seleção marcada e o texto "Selecione uma opção" e uma seta para baixo.
- Disciplina:** Um campo de texto com uma caixa de seleção desmarcada.
- Assunto:** Um campo de texto com uma caixa de seleção desmarcada.
- Nível de Ensino:** Um campo de seleção com uma caixa de seleção desmarcada e o texto "Selecione uma opção" e uma seta para baixo.

Um botão azul com o texto "Filtrar" está localizado na parte inferior direita do modal.

FONTE: O autor (2025).

## HU08 Cadastrar Cabeçalhos

**Sendo:** Sendo um usuário registrado e logado no sistema, durante a criação de avaliação.

**Quero:** Definir os parâmetros do cabeçalho para avaliação.

**Para:** Personalizar o que aparecerá no modelo final da avaliação

**Descrição:**

Durante o processo de criação de uma avaliação, o usuário deseja definir os parâmetros do cabeçalho, incluindo informações como nome da instituição, disciplina, curso ou nível, nome do professor, nome do aluno, identificador da avaliação e instruções. Isso permitirá que o professor personalize o cabeçalho da avaliação de acordo com as necessidades específicas da turma ou da instituição.

**Critérios de Aceitação:**

- O sistema deve fornecer uma interface para o professor definir os parâmetros do cabeçalho durante a criação da avaliação.
- Deve ser possível incluir informações como nome da instituição, disciplina, curso ou nível, nome do professor, nome do aluno, identificador da avaliação e instruções.
- O professor deve ter a capacidade de revisar e editar os parâmetros do cabeçalho antes de finalizar a criação da avaliação.
- O sistema deve garantir que os parâmetros do cabeçalho sejam exibidos corretamente no modelo final da avaliação.
- Deve ser possível salvar modelos de cabeçalho previamente configurados para reutilização em avaliações futuras.

**Critérios de Aceitação Detalhamento:**

1. O sistema deve fornecer uma interface para o professor definir os parâmetros do cabeçalho durante a criação da avaliação
  - DADO QUE: o usuário acessa a tela de criação de avaliações
  - QUANDO: ele insere os dados do cabeçalho
  - ENTÃO: o sistema dele associar os dados fornecidos a avaliação e permitir salvar o cabeçalho.

2. Deve ser possível incluir informações como nome da instituição, disciplina, curso ou nível, nome do professor, nome do aluno, identificador da avaliação e instruções.
  - DADO QUE: o usuário está na tela de criação de cabeçalho
  - QUANDO: preenche os dados necessários
  - ENTÃO: o sistema deve permitir remover ou adicionar elementos no cabeçalho de acordo com as escolhas do usuário.
3. O professor deve ter a capacidade de revisar e editar os parâmetros do cabeçalho antes de finalizar a criação da avaliação
  - DADO QUE: o usuário está na tela de criação de cabeçalhos
  - QUANDO: ele seleciona a opção “Salvar”
  - ENTÃO: o sistema colocar o usuário em modo de visualização e solicitar a confirmação final.
4. O sistema deve garantir que os parâmetros do cabeçalho sejam exibidos corretamente no modelo final da avaliação
  - DADO QUE: o usuário finaliza a criação de uma avaliação
  - QUANDO: ele salva e visualiza a avaliação impressa
  - ENTÃO: o sistema deve redimensionar os dados do cabeçalho para acomodar o modelo de impressão escolhido e permitir alterações no seu formato.
5. Deve ser possível salvar modelos de cabeçalho previamente configurados para reutilização em avaliações futuras.
  - DADO QUE: o usuário está na tela de criação de avaliações
  - QUANDO: ele seleciona a opção “Carregar Cabeçalho”
  - ENTÃO: o sistema deve mostrar uma lista com cabeçalhos salvos e trazer os dados do cabeçalho escolhido.

FIGURA – Tela de cadastro de cabeçalho

**QUESTIONARIUM** | Questões | Avaliações | Turmas | Sobre | Olá, Razer

### Novo/Edição de Cabeçalho

Instituição:

Curso:

Disciplina:

Professor:

[+ Novo Campo](#)

Instruções:

[← Voltar](#) [Cancelar](#) [Salvar](#)

suporte@questionarium.com | Questionarium © 2024

FONTE: O autor (2025).

FIGURA – Tela de manutenção de cabeçalho

### Cabeçalhos

[+ Novo](#)

Instituição	Disciplina	Curso		
UFPR	Matemática Financeira	Administração Financeira		
UFMG	Biologia Celular	Biomedicina		
UFPA	História do Brasil	História		
UFPR	Literatura Mundial	Letras		
UFMG	Física Moderna	Física		
UFSC	Geometria Analítica	Engenharia Civil		
UFPR	Química Orgânica	Química		
UFMG	Economia Global	Economia Internacional		
UFRS	Psicologia do Desenvolvimento	Psicologia		
	Direito Constitucional	Direito		
	Tecnologia da Informação e Comunicação	Ciência da Computação		

[← Voltar](#)

suporte@questionarium.com | Questionarium © 2024

FONTE: O autor (2025).

## HU09 Criar Aplicação de Avaliação

Sendo: Sendo um usuário registrado e logado no sistema.

Quero: Criar uma aplicação de uma avaliação específica.

Para: Gerar a avaliação em seu formato final e poder aplicá-la.

Descrição:

O sistema deve permitir a criação de aplicação de avaliações. Após configurar os parâmetros, o professor pode gerar a prova física, gerando um QR Code automático de acordo com o número de alunos e turma, ou apenas salvar a aplicação para uso futuro em uma aplicação física.

Critérios de Aceitação:

- O usuário poderá escolher uma avaliação, definir número de alunos, possibilidade de embaralhar questões e data de aplicação.

Critérios de Aceitação Detalhamento:

1. O usuário poderá escolher uma avaliação, definir número de alunos, possibilidade de embaralhar questões.
  - DADO QUE: o usuário acessa a tela de aplicação de avaliações
  - QUANDO: ele seleciona um modelo de avaliação
  - ENTÃO: o sistema mostra um formulário para definição de número de avaliação, a data de aplicação e opção para embaralhamento de questões

FIGURA – Tela de seleção de avaliação para aplicação



FONTE: O autor (2025).

FIGURA – Tela de cadastro de modelo de avaliação

**QUESTIONARIUM** Questões Avaliações Turmas Relatórios Sobre Olá, Razer

## Avaliação

**Cabeçalho**

Instituição  Curso  Instruções

Disciplina  Professor

Salvar Cabeçalho Carregar Cabeçalho

**Questão 01**  
 Instituição: Instituição xx Enunciado: "Lorem ipsum dolor sit amet, consectetur a... Pública

**Questão 02**  
 Instituição: Instituição xx Enunciado: "Lorem ipsum dolor sit amet, consectetur a... Pública

suporte@questionarium.com Questionarium © 2024

**Questão 03**  
 Instituição: Instituição xx Enunciado: "Lorem ipsum dolor sit amet, consectetur a... Privado

+ Nova Questão

[Crie sua questão aqui](#)


Turma  Cancelar Salvar

FONTE: O autor (2025).

FIGURA – Tela de pré-visualização de avaliação

**QUESTIONARIUM** | Questões | Avaliações | Turmas | Relatórios | Sobre | Olá, Razer

### Formatação

Visualização 

#### Avaliação

UFPR | TADS | Engenharia de Software 2 | Razer A N R Montaña

Instruções:

- Não colar
- Usar apenas caneta preta
- Proibido consulta
- Saída permitida após 15 min.

Nome: \_\_\_\_\_ Data: 23/06/2024

1) (xx) Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat:

A) Nam ac diam lobortis, posuere augue vitae, laoreet tortor. Quisque non neque felis. Duis quis ipsum non urna ullamcorper ullamcorper. Curabitur cursus at leo ac posuere. Ut consequat tincidunt risus non faucibus. Donec at volutpat nulla. Mauris in leo in arcu malesuada mollis.

B) Nam ac diam lobortis, posuere augue vitae, laoreet tortor. Quisque non neque felis. Duis quis ipsum non urna ullamcorper ullamcorper. Curabitur cursus at leo ac posuere. Ut consequat tincidunt risus non faucibus. Donec at volutpat nulla. Mauris in leo in arcu malesuada mollis.

C) Nam ac diam lobortis, posuere augue vitae, laoreet tortor. Quisque non neque felis. Duis quis ipsum non urna ullamcorper ullamcorper. Curabitur cursus at leo ac posuere. Ut consequat tincidunt risus non faucibus. Donec at volutpat nulla. Mauris in leo in arcu malesuada mollis.

D) Nam ac diam lobortis, posuere augue vitae, laoreet tortor. Quisque non neque felis. Duis quis ipsum non urna ullamcorper ullamcorper. Curabitur cursus at leo ac posuere. Ut consequat tincidunt risus non faucibus. Donec at volutpat nulla. Mauris in leo in arcu malesuada mollis.

E) Nam ac diam lobortis, posuere augue vitae, laoreet tortor. Quisque non neque felis. Duis quis ipsum non urna ullamcorper ullamcorper. Curabitur cursus at leo ac posuere. Ut consequat tincidunt risus non faucibus. Donec at volutpat nulla. Mauris in leo in arcu malesuada mollis.

2) (xx) Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat:

A) Nam ac diam lobortis, posuere augue vitae, laoreet tortor. Quisque non neque felis. Duis quis ipsum non urna ullamcorper ullamcorper. Curabitur cursus at leo ac posuere. Ut consequat tincidunt risus non faucibus. Donec at volutpat nulla. Mauris in leo in arcu malesuada mollis.

B) Nam ac diam lobortis, posuere augue vitae, laoreet tortor. Quisque non neque felis. Duis quis ipsum non urna ullamcorper ullamcorper. Curabitur cursus at leo ac posuere. Ut consequat tincidunt risus non faucibus. Donec at volutpat nulla. Mauris in leo in arcu malesuada mollis.

C) Nam ac diam lobortis, posuere augue vitae, laoreet tortor. Quisque non neque felis. Duis quis ipsum non urna ullamcorper ullamcorper. Curabitur cursus at leo ac posuere. Ut consequat tincidunt risus non faucibus. Donec at volutpat nulla. Mauris in leo in arcu malesuada mollis.

D) Nam ac diam lobortis, posuere augue vitae, laoreet tortor. Quisque non neque felis. Duis quis ipsum non urna ullamcorper ullamcorper. Curabitur cursus at leo ac posuere. Ut consequat tincidunt risus non faucibus. Donec at volutpat nulla. Mauris in leo in arcu malesuada mollis.

E) Nam ac diam lobortis, posuere augue vitae, laoreet tortor. Quisque non neque felis. Duis quis ipsum non urna ullamcorper ullamcorper. Curabitur cursus at leo ac posuere. Ut consequat tincidunt risus non faucibus. Donec at volutpat nulla. Mauris in leo in arcu malesuada mollis.

3) (xx) Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat:

A) Nam ac diam lobortis, posuere augue vitae, laoreet tortor. Quisque non neque felis. Duis quis ipsum non urna ullamcorper ullamcorper. Curabitur cursus at leo ac posuere. Ut consequat tincidunt risus non faucibus. Donec at volutpat nulla. Mauris in leo in arcu malesuada mollis.

B) Nam ac diam lobortis, posuere augue vitae, laoreet tortor. Quisque non neque felis. Duis quis ipsum non urna ullamcorper ullamcorper. Curabitur cursus at leo ac posuere. Ut consequat tincidunt risus non faucibus. Donec at volutpat nulla. Mauris in leo in arcu malesuada mollis.

C) Nam ac diam lobortis, posuere augue vitae, laoreet tortor. Quisque non neque felis. Duis quis ipsum non urna ullamcorper ullamcorper. Curabitur cursus at leo ac posuere. Ut consequat tincidunt risus non faucibus. Donec at volutpat nulla. Mauris in leo in arcu malesuada mollis.

D) Nam ac diam lobortis, posuere augue vitae, laoreet tortor. Quisque non neque felis. Duis quis ipsum non urna ullamcorper ullamcorper. Curabitur cursus at leo ac posuere. Ut consequat tincidunt risus non faucibus. Donec at volutpat nulla. Mauris in leo in arcu malesuada mollis.


E) Nam ac diam lobortis, posuere augue vitae, laoreet tortor. Quisque non neque felis. Duis quis ipsum non urna ullamcorper ullamcorper. Curabitur cursus at leo ac posuere. Ut consequat tincidunt risus non faucibus. Donec at volutpat nulla. Mauris in leo in arcu malesuada mollis.

Página 2

Página 3

Pinte todo o quadro, como no exemplo:

1		B	C	D	E
---	--	---	---	---	---



ID: 0039  
[Verifique seu resultado pelo app!](#)

#### Gabarito

1	A	B	C	D	E
2	A	B	C	D	E
3	A	B	C	D	E
4	A	B	C	D	E
5	A	B	C	D	E
6	A	B	C	D	E
7	A	B	C	D	E

Cancelar Salvar

FONTE: O autor (2025).

## HU10 Gerar e Visualizar Relatórios

Sendo: Sendo um usuário registrado e logado no sistema.

Quero: Gerar relatórios estatísticos.

Para: Analisar indicadores relacionados às avaliações aplicadas.

Descrição:

O professor deve ter a capacidade de gerar relatórios estatísticos sobre as avaliações aplicadas. Esses relatórios devem incluir informações como notas, acertos, erros e outros indicadores fornecidos pelo sistema.

Critérios de Aceitação:

- O sistema deve mostrar as avaliações disponíveis para geração de relatórios.
- O sistema deve apresentar opção de exportação do relatório para formatos comuns como Excel (xls), arquivos Comma-separated values (csv), PDF, DOC.

Critérios de Aceitação Detalhamento:

1. O sistema deve mostrar as avaliações disponíveis para geração de relatórios
  - DADO QUE: o usuário está na tela de avaliações
  - QUANDO: ele seleciona o ícone de relatórios
  - ENTÃO: o sistema deve redirecionar o usuário para a tela de relatórios e permitir que o usuário escolha o tipo de relatório a ser gerado.
2. O sistema deve apresentar opção de exportação do relatório para formatos comuns como Excel (xls), arquivos Comma-separated values (csv), PDF, DOC e outros
  - DADO QUE: o usuário acessa a tela de relatórios
  - QUANDO: ele seleciona a opção de impressão
  - ENTÃO: o sistema deve abrir uma tela com as opções de impressão disponíveis.

FIGURA – Tela de relatório de avaliação

**QUESTIONARIUM**

Questões Avaliações Turmas Sobre

Olá, Razer

### Relatório de Avaliação

**Aplicação A00001**

Data: 17/03/2024  
 Hora: 19:00 - 20:00  
 Turma: DAC 5P 2022  
 Disciplina: DAC

Provas geradas: 35  
 Provas corrigidas: 20  
 Provas não corrigidas: 15

Número de questões: 20

Assuntos:

- Tag 1
- Tag 2
- Tag 3
- Tag 4
- Tag 5
- Tag 6
- Tag 7
- Tag 8

**Alunos:**

Código de Avaliação	Identificador	Acertos	Erros	Status
A00001.1	Aluno 1	17	3	CORRIGIDO
A00001.2	Aluno 2	-	-	CORRIGIDO
A00001.3	Aluno 3	-	-	CORRIGIDO
A00001.4	Aluno 4	-	-	-
A00001.5	Aluno 5	-	-	-
-	-	-	-	-
-	-	-	-	-
-	-	-	-	-
-	-	-	-	-
-	-	-	-	-
-	-	-	-	-
-	-	-	-	-

← Voltar

suporte@questionarium.com Questionarium © 2024

FONTE: O autor (2025).

## HU11 Corrigir Avaliações Aplicadas

Sendo: Sendo um usuário registrado e logado na aplicação móvel.

Quero: Corrigir as provas dos alunos através da câmera do aplicativo.

Para: Facilitar o processo de correção e acessar os dados de correção no sistema.

Descrição:

O professor deve ter a capacidade de corrigir as provas dos alunos escaneando o QR Code e gabarito do aluno presente nas provas físicas utilizando um aplicativo móvel. O sistema deve ser capaz de extrair os dados do QR Code e gabarito, compará-los com o gabarito correto e realizar a correção automaticamente. Os resultados da correção devem ser armazenados e disponibilizados para o professor no sistema.

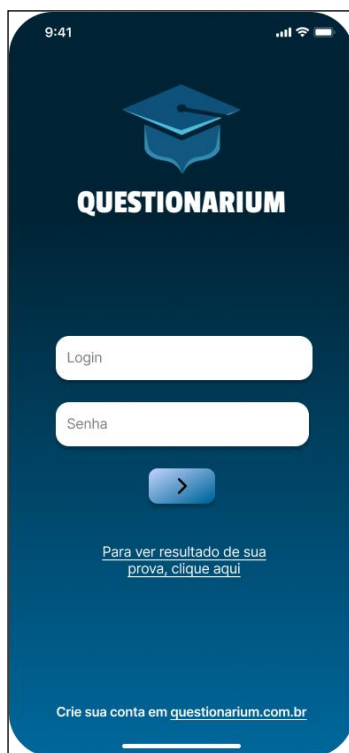
Critérios de aceitação:

- O sistema deve ser capaz de enviar os dados para o sistema de forma automática

Critérios de Aceitação Detalhamento:

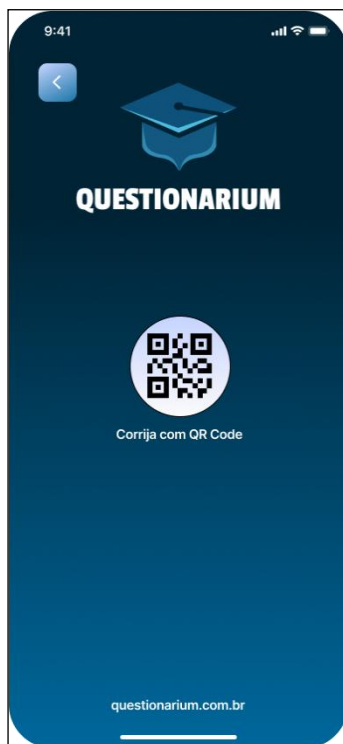
1. O sistema deve ser capaz de enviar os dados para o sistema de forma automática
  - DADO QUE: o usuário está realizando a correção das avaliações
  - QUANDO: ele escanear o QR Code e fornece os dados necessários
  - ENTÃO: o sistema atualiza o banco de dados com os dados da correção.

FIGURA – Tela mobile de login professor



FONTE: O autor (2025).

FIGURA – Tela mobile home professor



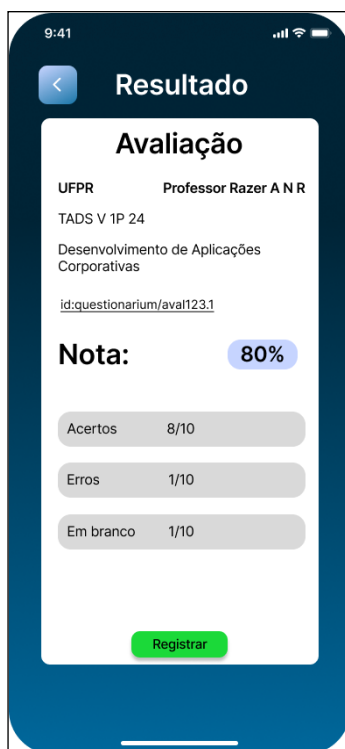
FONTE: O autor (2025).

FIGURA – Tela mobile de correção



FONTE: O autor (2025).

FIGURA – Tela mobile de resultado de correção

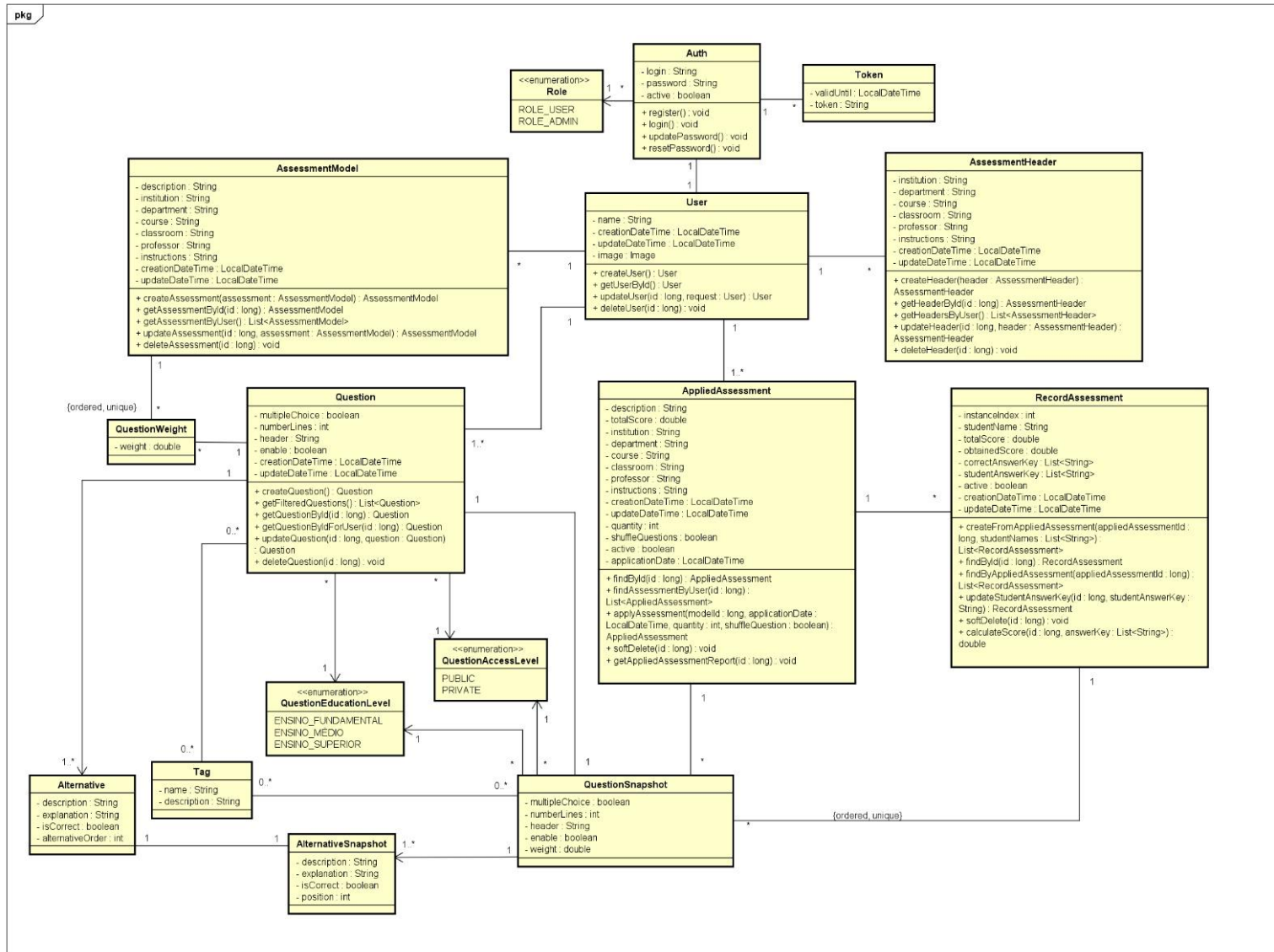


FONTE: O autor (2025).

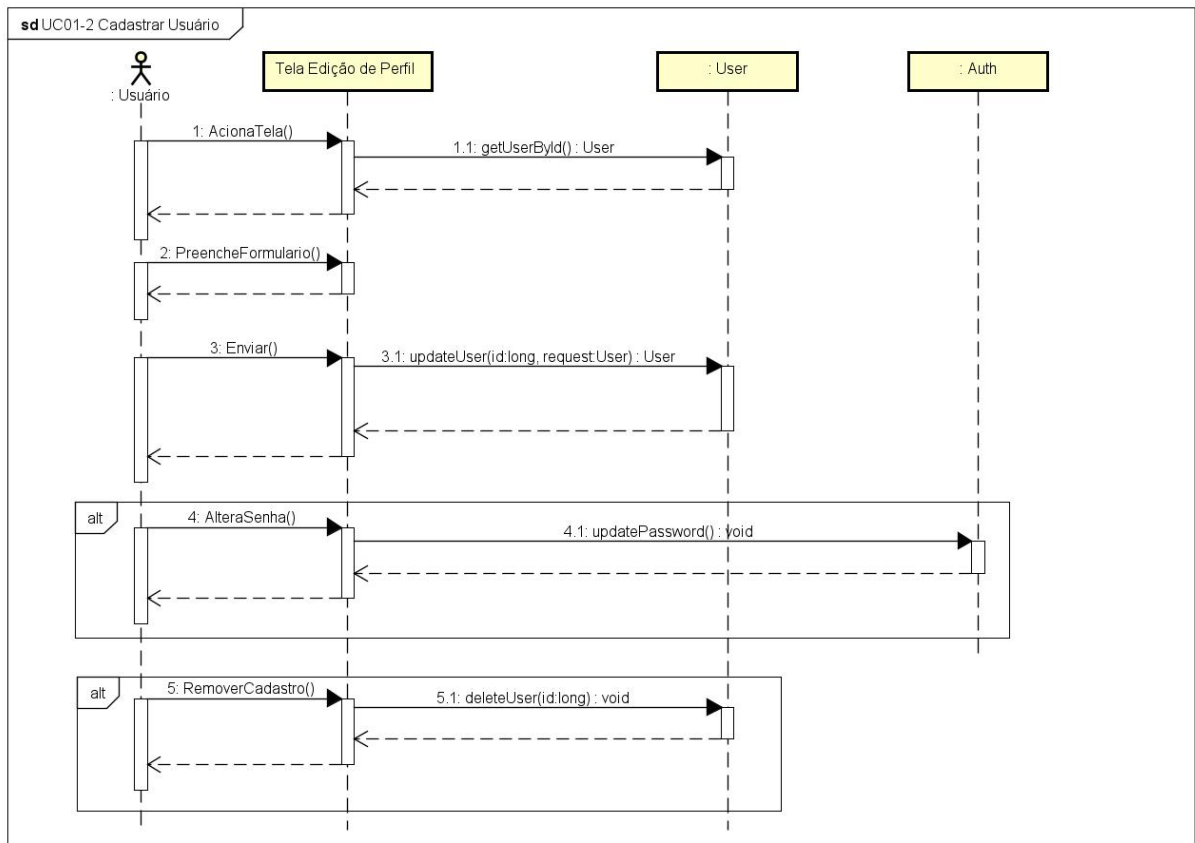
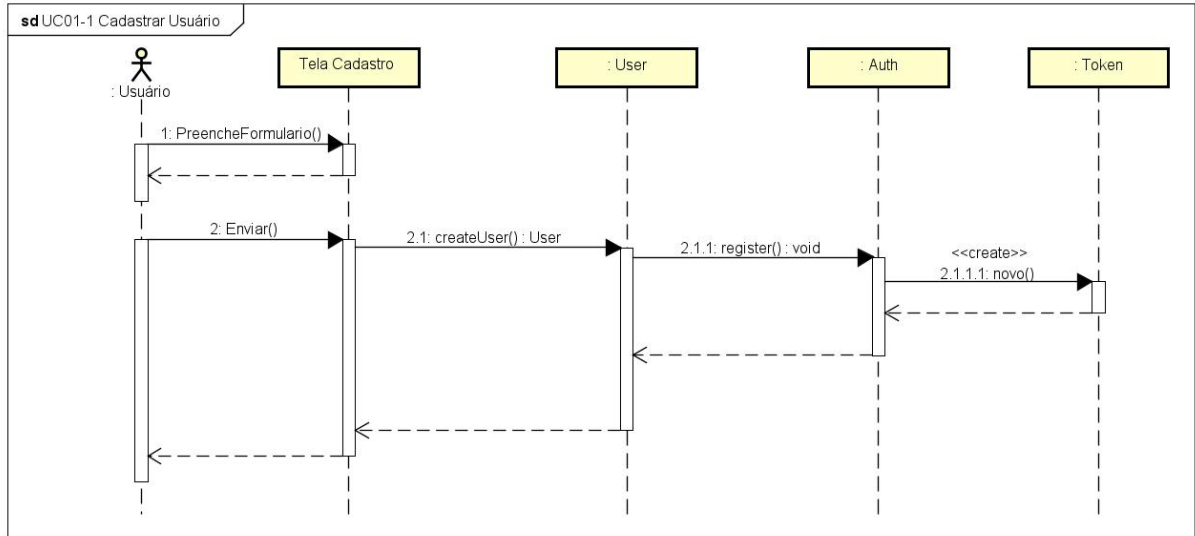
**Nota adicionais:**

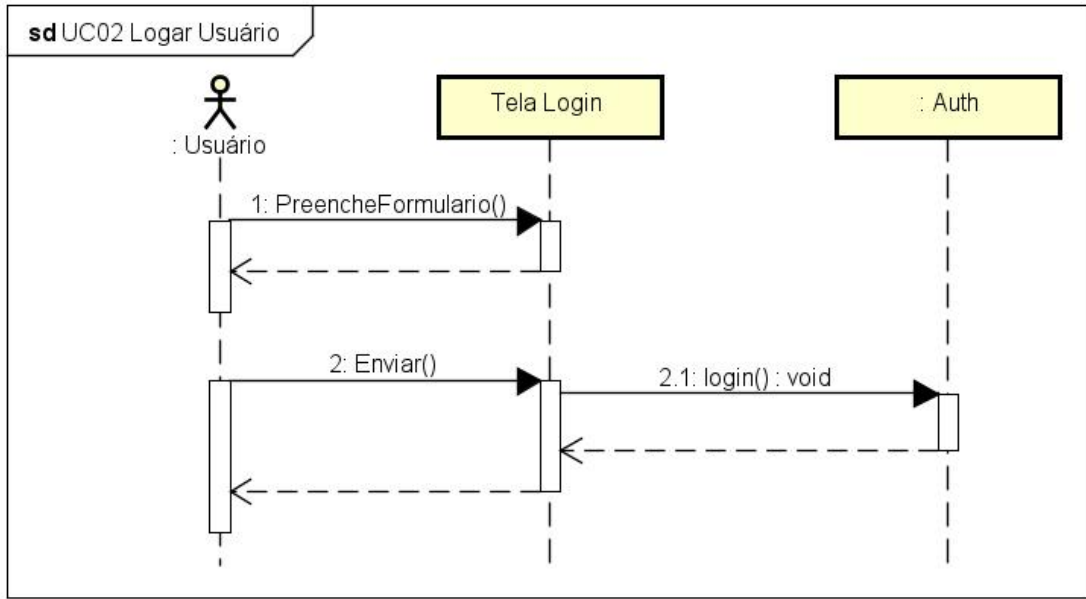
- O sistema deve garantir a segurança dos dados dos usuários, utilizando práticas de criptografia e proteção contra acesso não autorizado.
- O formulário de cadastro (HU01) deve ser intuitivo e fácil de usar, para proporcionar uma experiência agradável ao usuário.
- As mensagens de erro devem ser claras e informativas, caso haja problemas durante o processo de cadastro (HU01).
- O sistema deve se proteger de ataques de força bruta ou outros tipos de ataques de segurança durante o processo de login (HU02).
- As informações de login dos usuários devem ser mantidas de forma segura e confidencial no banco de dados (HU02).
- O sistema deve garantir a consistência e integridade dos dados do banco de questões durante todas as operações de CRUD (HU03).
- As permissões de acesso devem ser aplicadas adequadamente, garantindo que apenas usuários autorizados possam criar, ler, atualizar e excluir questões (HU03).
- Avaliações que não foram aplicadas não podem ter relatórios gerados (HU10).

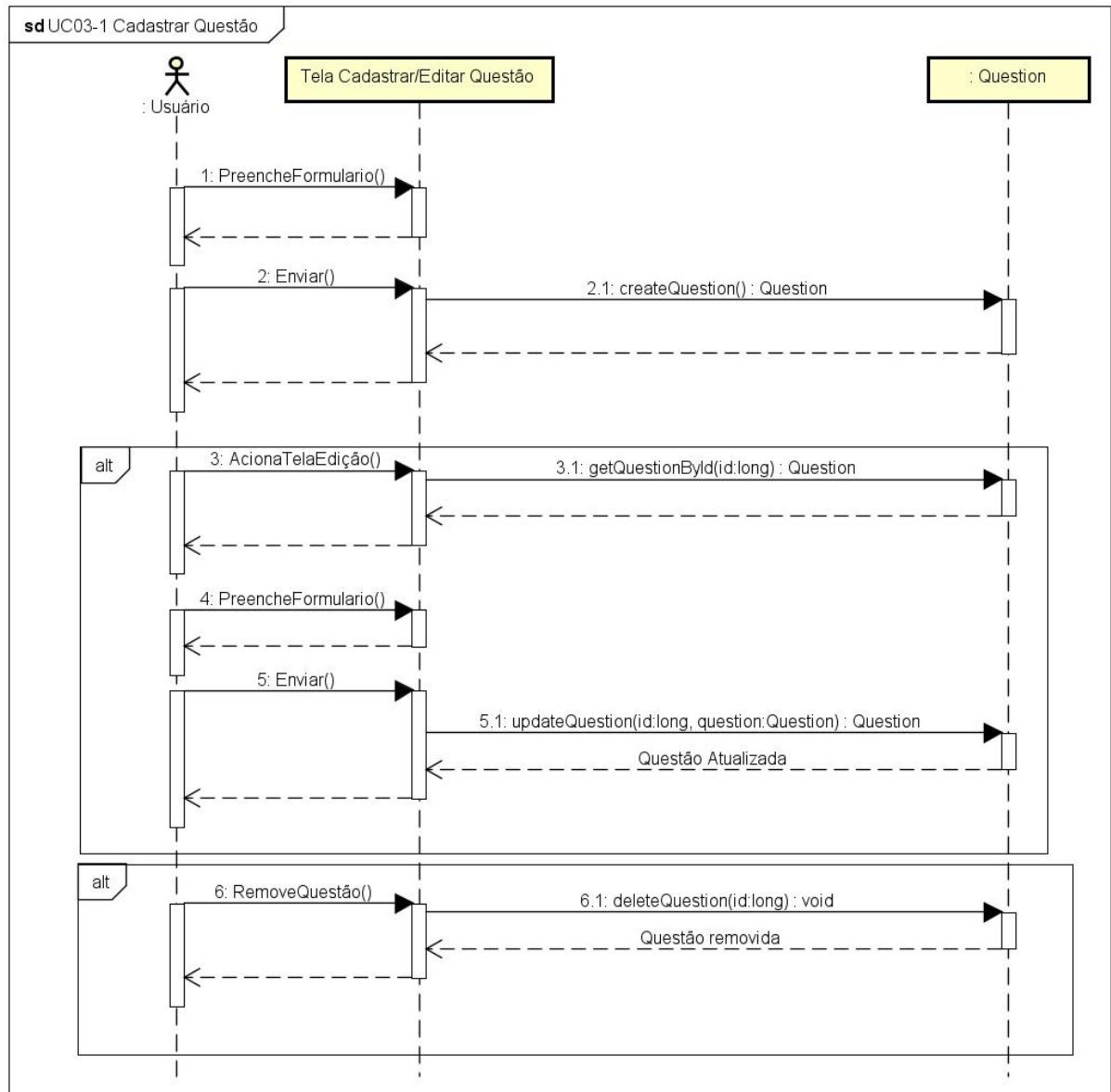
## APÊNDICE C – DIAGRAMAS DE CLASSES

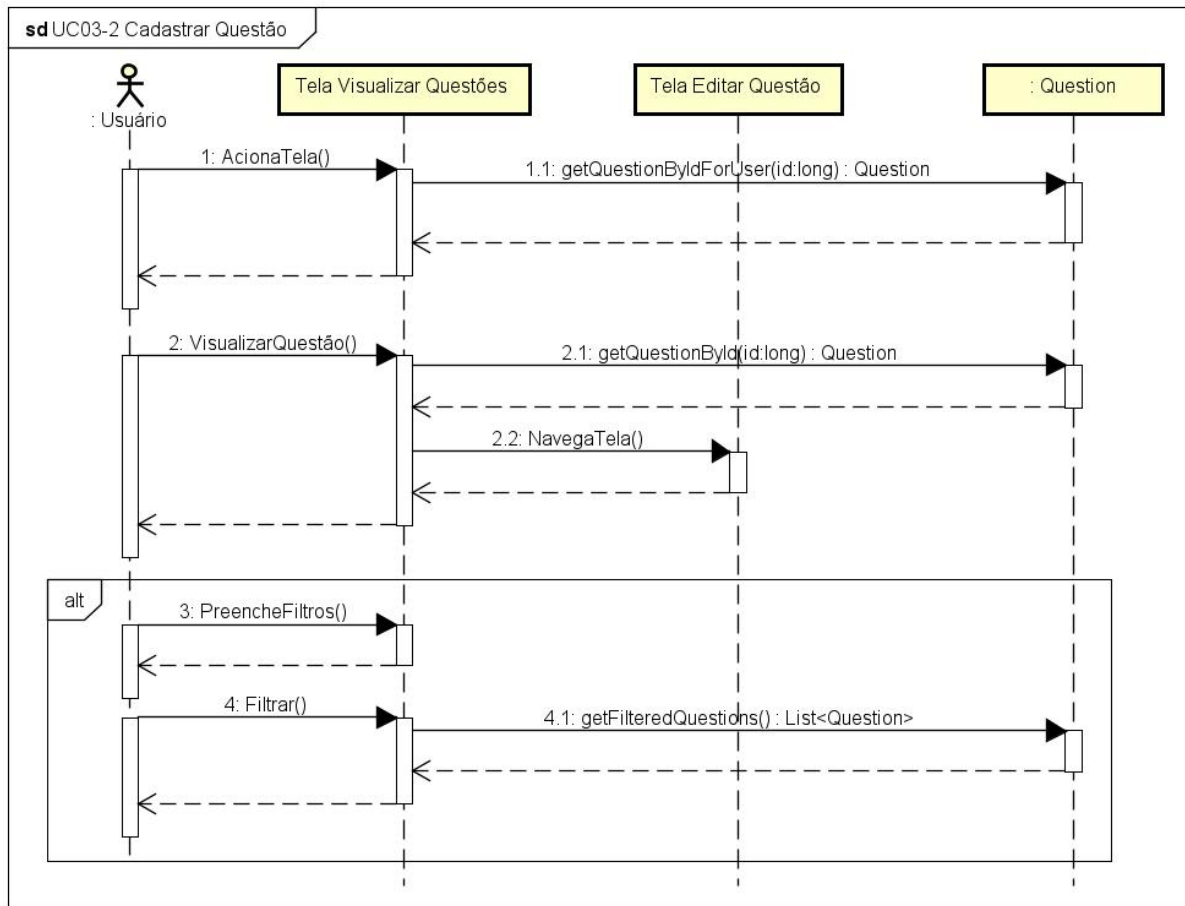


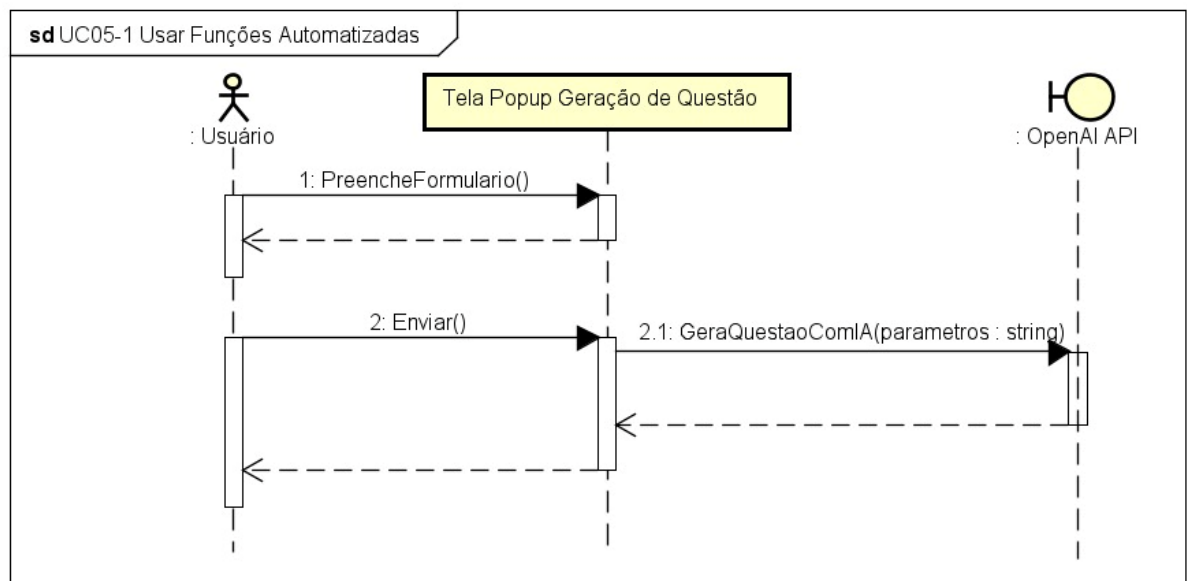
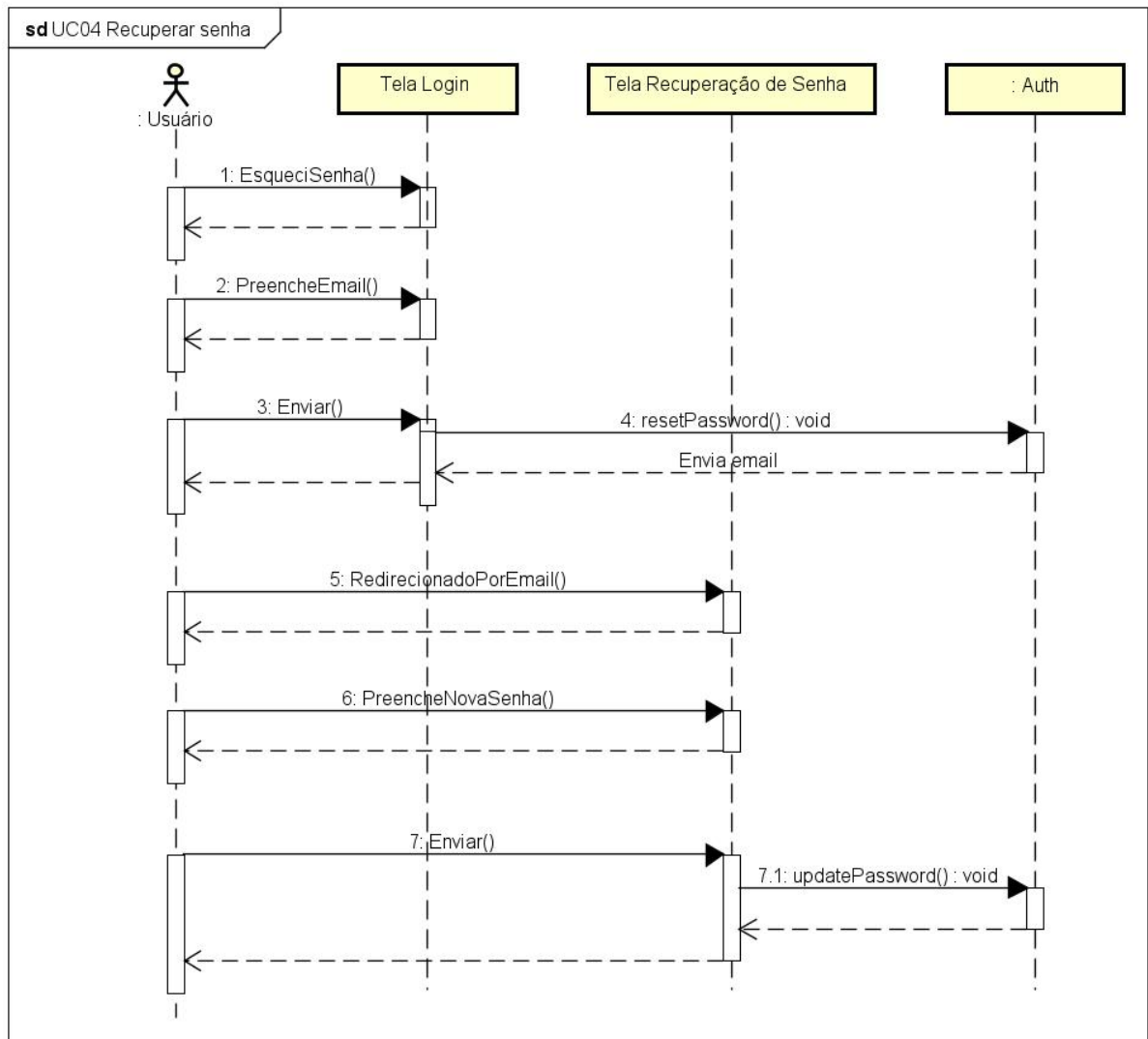
## APÊNDICE D – DIAGRAMAS DE SEQUÊNCIA

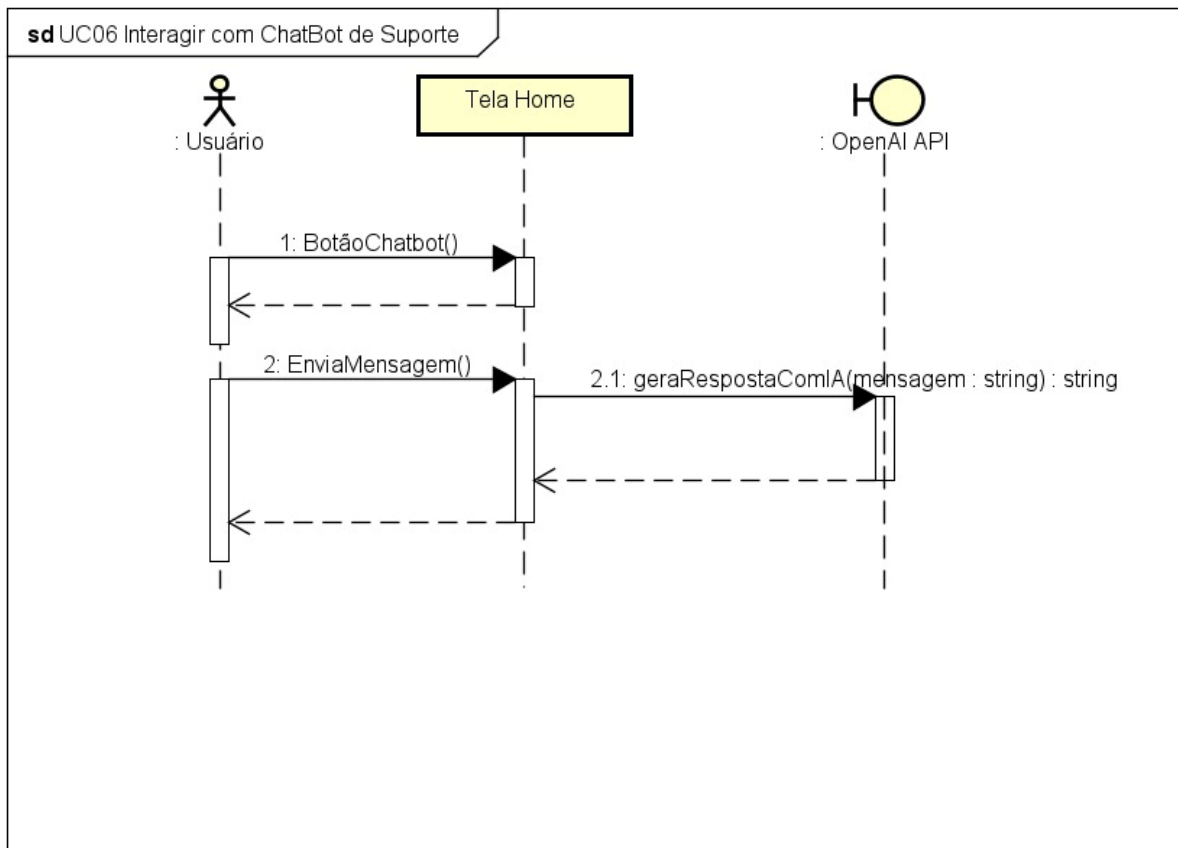
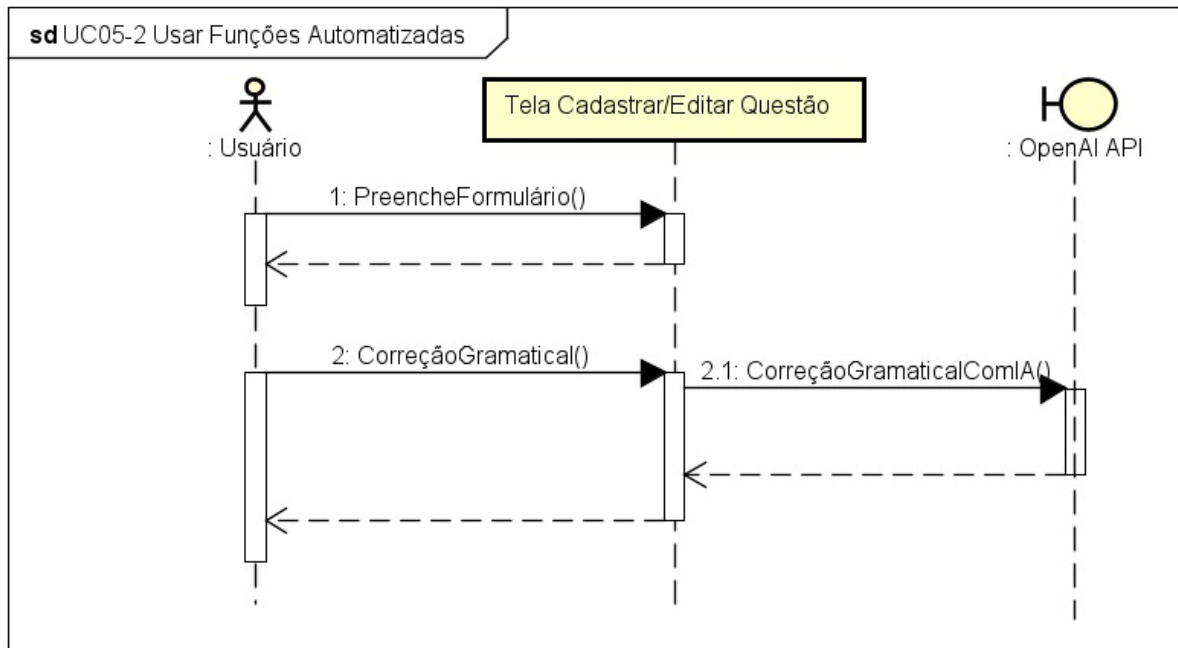


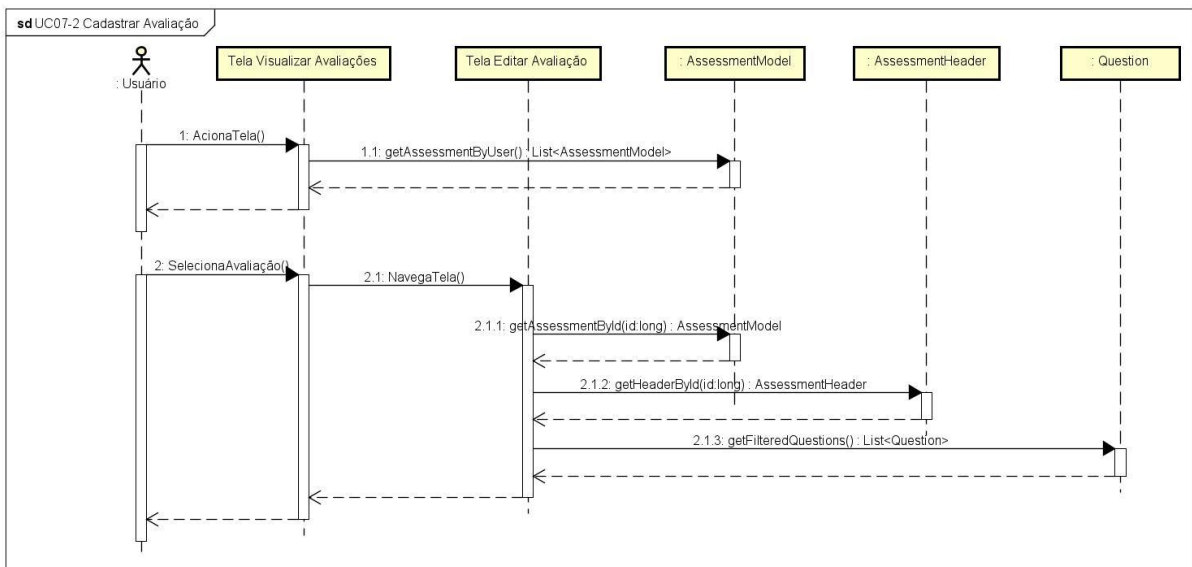
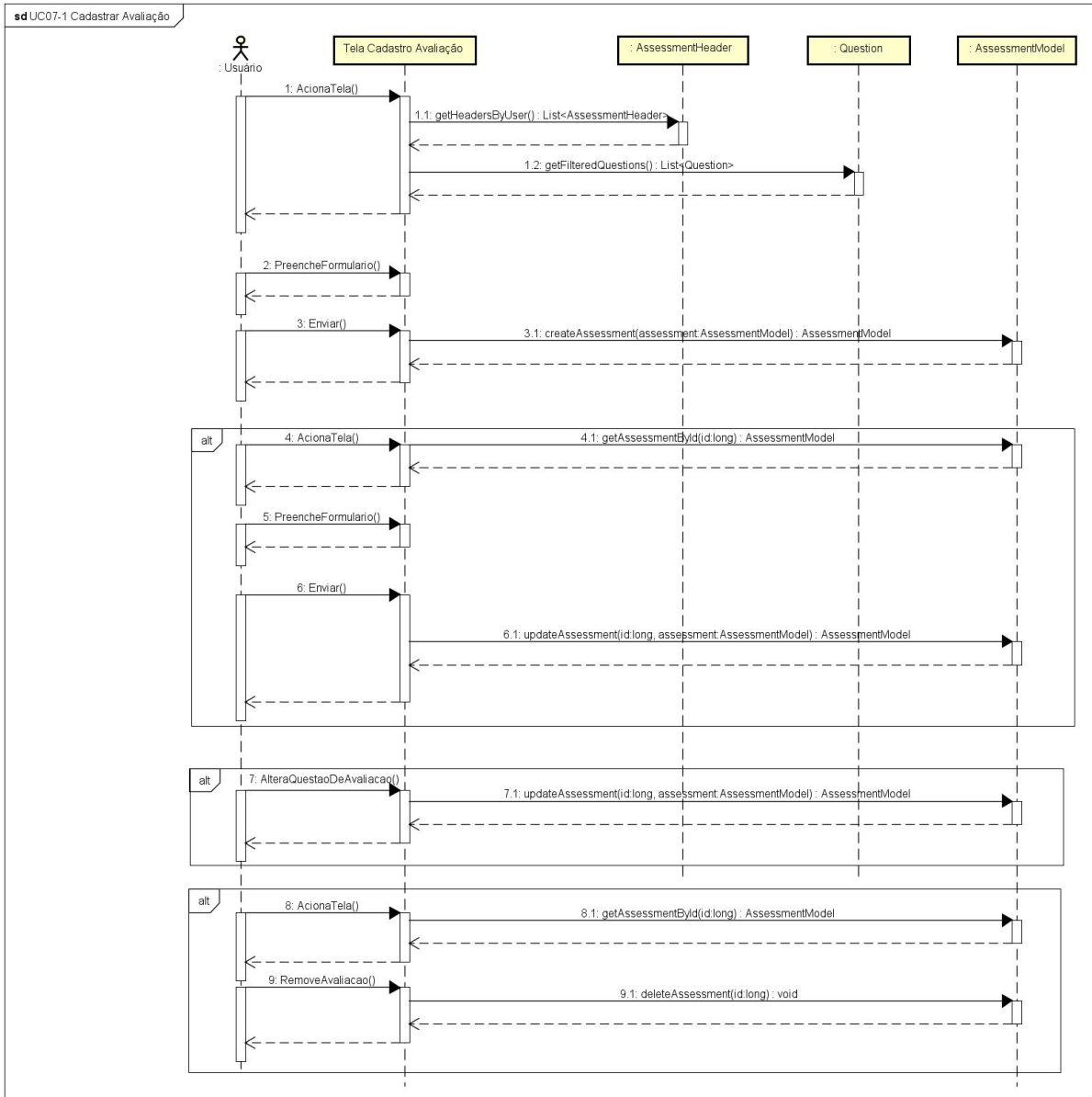


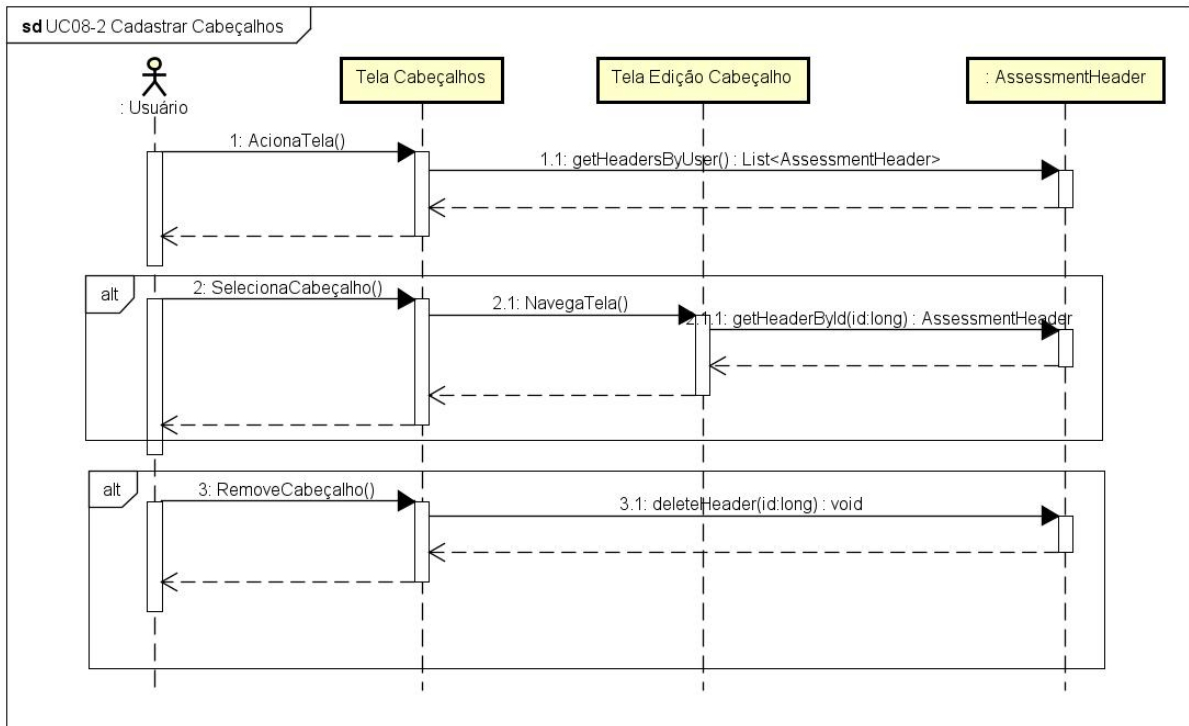
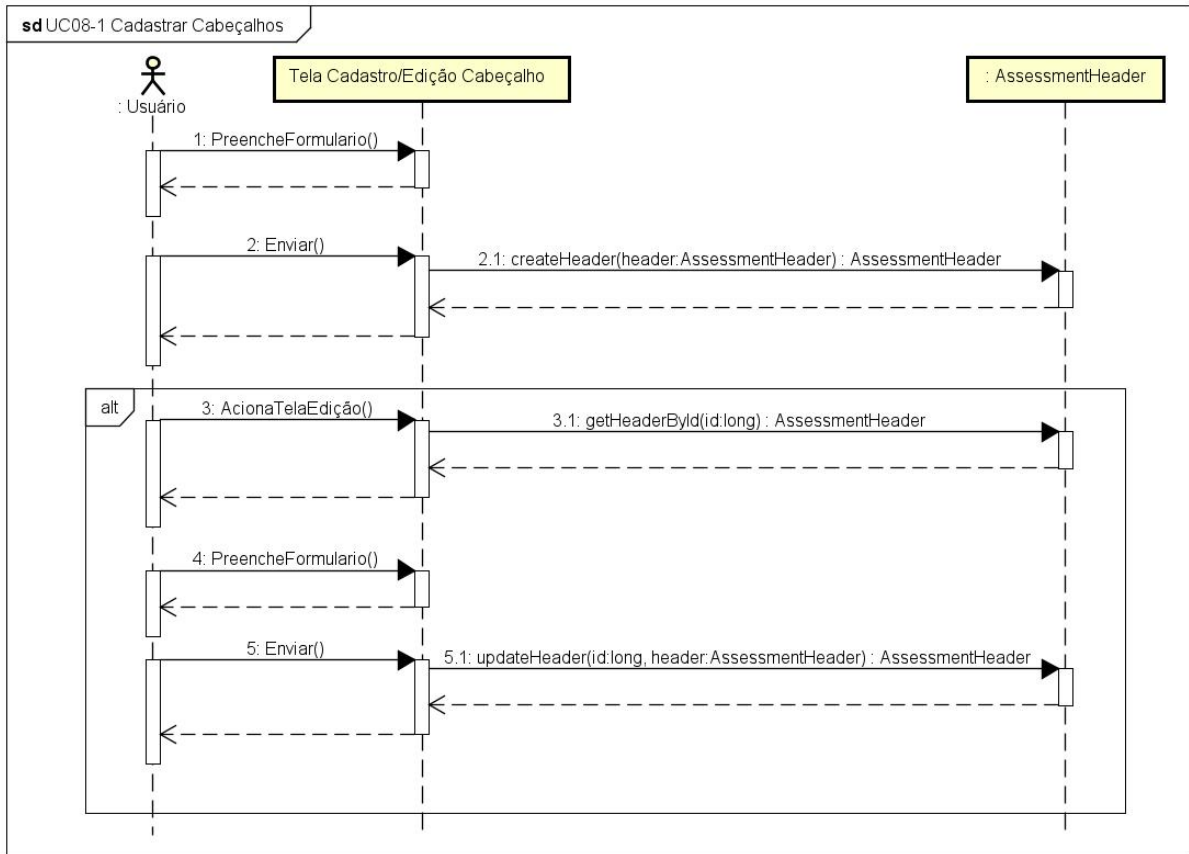


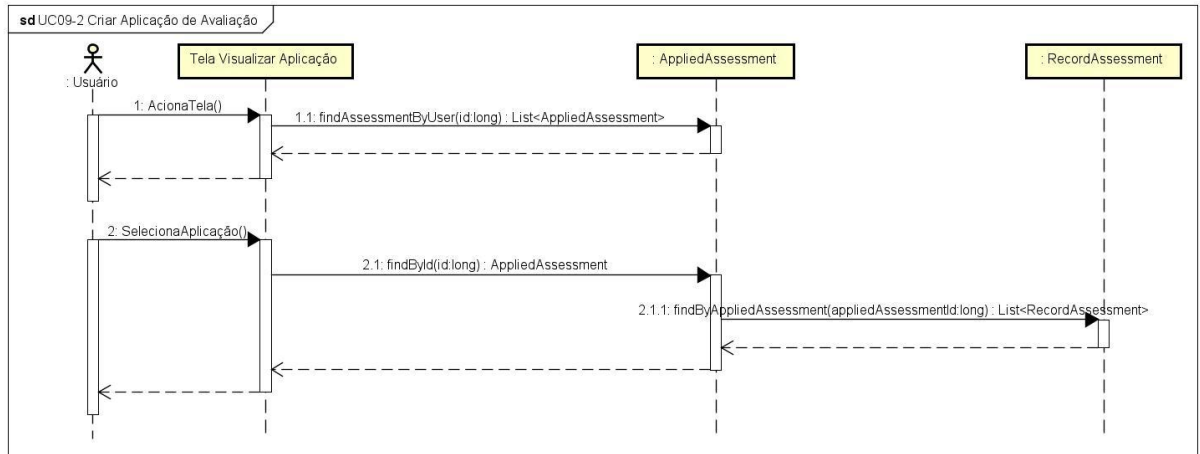
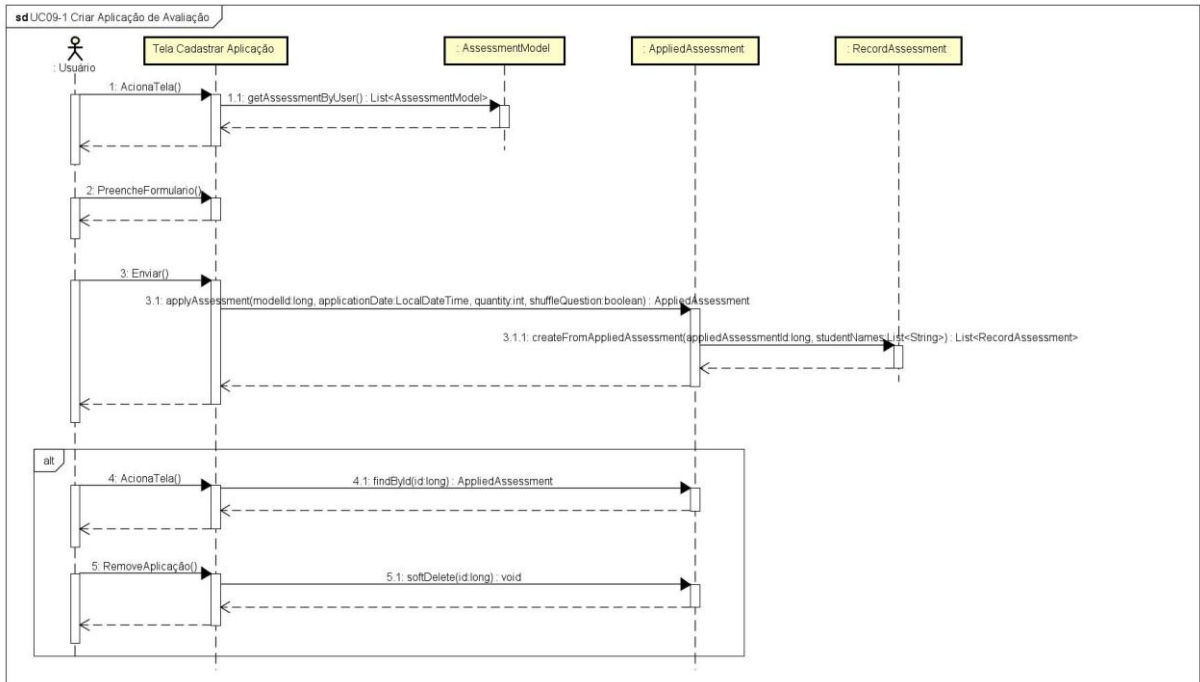


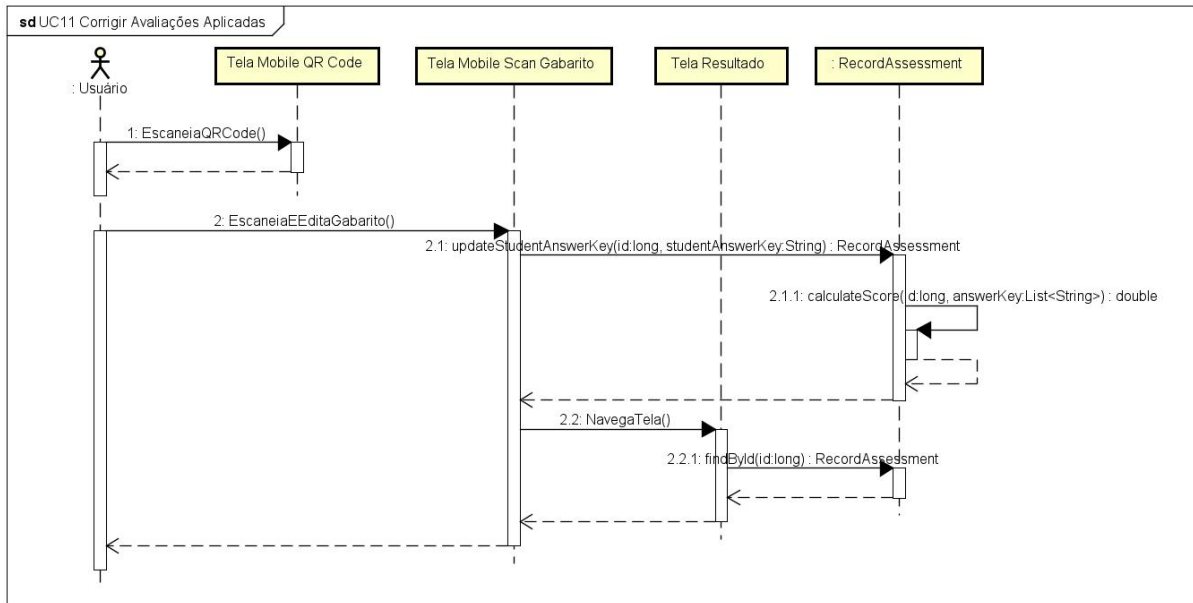
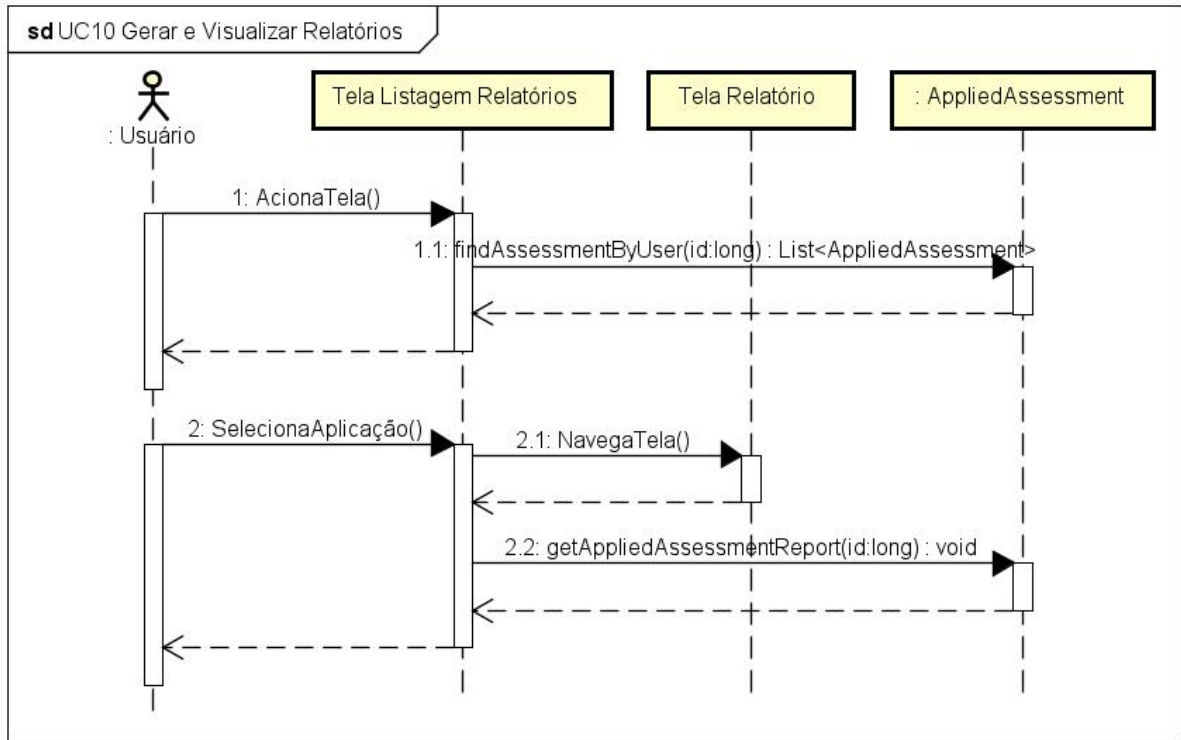




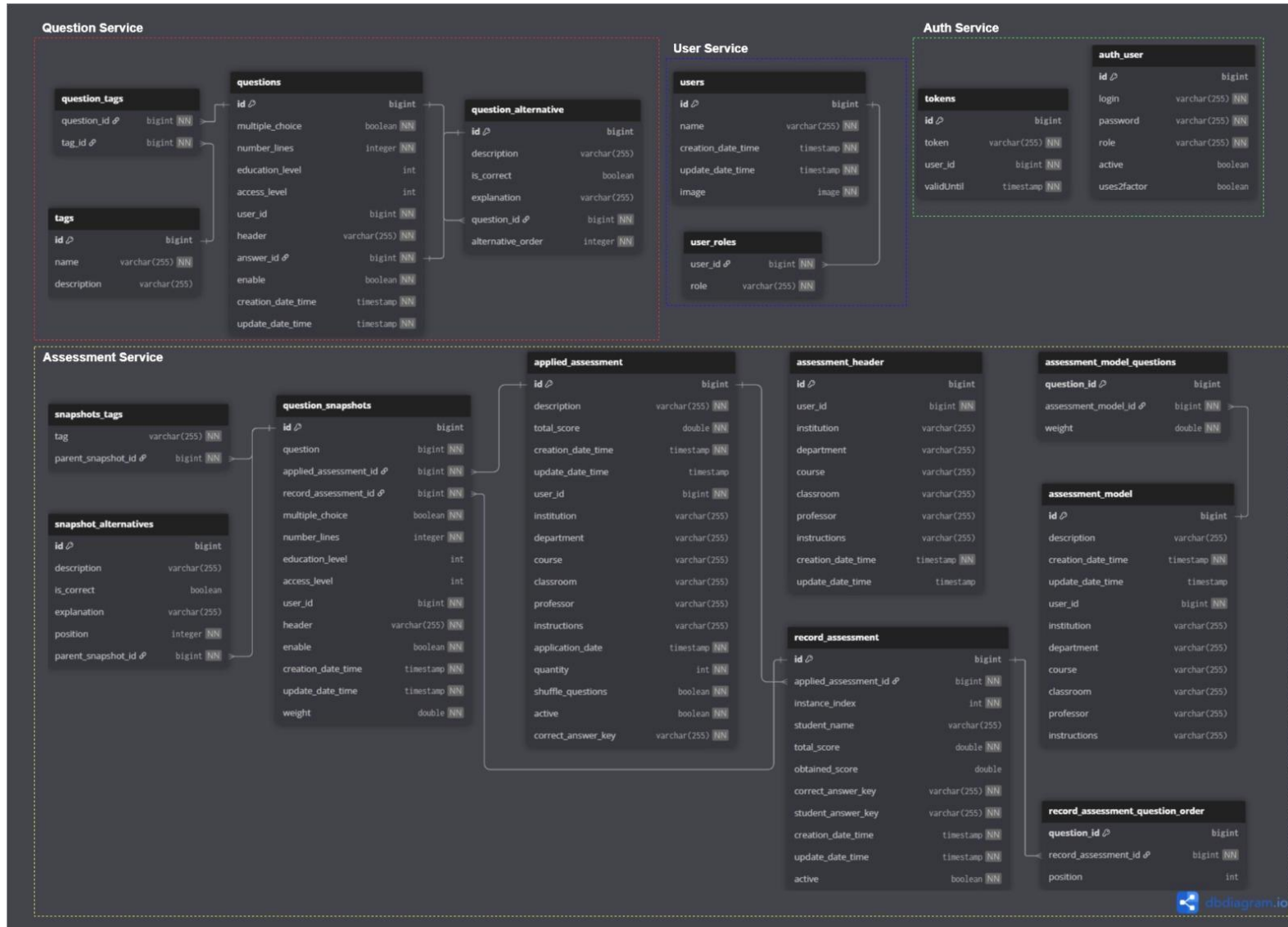








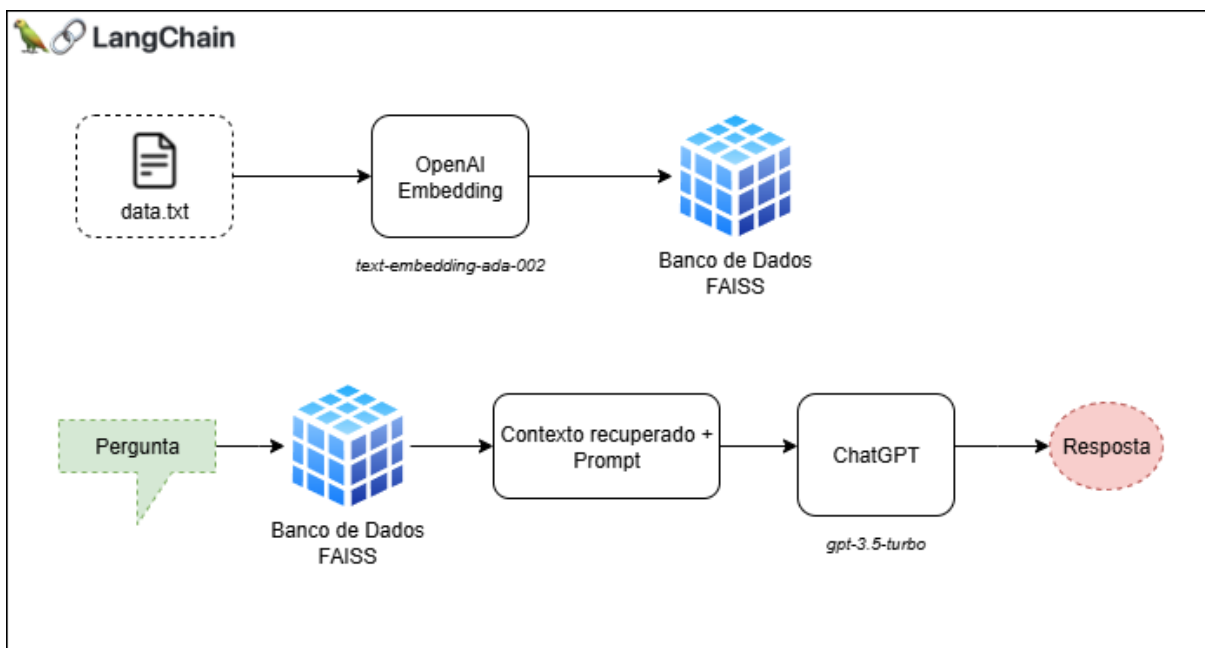
## APÊNDICE E – DIAGRAMA FÍSICO DO BANCO DE DADOS



## APÊNDICE F – IMPLEMENTAÇÃO DA INTELIGÊNCIA ARTIFICIAL NO SOFTWARE

A inteligência artificial foi usada no software em duas principais funcionalidades: Chatbot de Suporte e Criação de Questão e Correção Gramatical. Este Apêndice apresenta a implementação da tecnologia utilizada.

### F.1 ChatBot de Suporte



FONTE: O autor (2025).

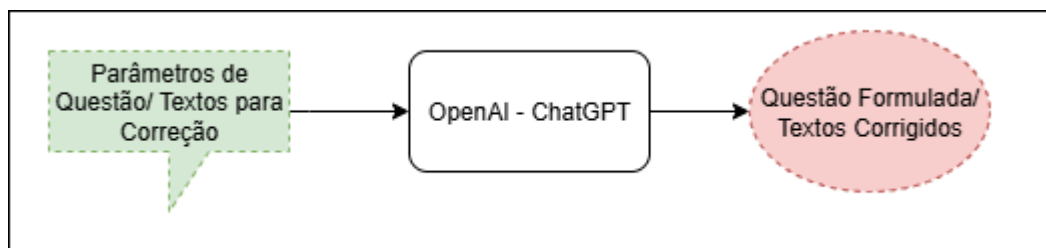
A figura acima é uma representação de alto nível de dois fluxos de trabalho: configuração do banco de dados e perguntas e respostas. Para a orquestração dos fluxos foi utilizado o framework LangChain, que oferece ferramentas para todos os passos do desenvolvimento de agentes de IA.

O primeiro fluxo, configuração do banco de dados, consiste em dividir o documento data.txt, que contém dados, características e manuais de uso do software, em pequenos pedaços. Após isso, o modelo da OpenAI text-embedding-ada-002 é responsável pelo embedding, ou seja, transformar os pedaços em vetores semânticos, que por sua vez são armazenados no banco vetorial FAISS (Facebook AI Similarity Search).

O segundo fluxo, perguntas e respostas, é a interação com o ChatBot de fato. O sistema recebe a pergunta, faz uma pesquisa de contexto por similaridade no banco vetorial, passa o contexto e um prompt personalizado para o ChatGPT que monta a resposta.

## F.2 Criação de Questões e Correção Gramatical

A segunda funcionalidade que utiliza IA é a criação de questões e correção gramatical, tendo implementação mais simples que o Chatbot de Suporte.



FONTE: O autor (2025).

Para cada funcionalidade, o fluxo de trabalho é o seguinte:

### Criação de Questão

1. Usuário insere parâmetros como tema, categorias, nível de ensino etc.
2. O sistema monta um prompt personalizado com os parâmetros e envia para a OpenAI, utilizando ChatGPT para montagem de respostas.
3. A OpenAI responde e a nova questão é mostrada para o usuário.

### Correção Gramatical

1. Usuário insere textos como enunciado, alternativas e explicações.
2. O sistema monta um prompt personalizado com os textos e envia para a OpenAI, utilizando ChatGPT para montagem de respostas.
3. A OpenAI responde e os textos corrigidos substituem os antigos.