



INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA
PARAÍBA – CAMPUS CAMPINA GRANDE
CURSO SUPERIOR DE TECNÓLOGO EM TELEMÁTICA

**ESTRATÉGIAS E BENEFÍCIOS NA MIGRAÇÃO DO SOFTWARE QUESTÕESPRO
WEB PARA O AMBIENTE MOBILE: UMA ANÁLISE FOCADA NAS VANTAGENS
PARA PROFESSORES E ALUNOS**

RODRIGO CUNHA FRANÇA

Campina Grande – PB

2023

Rodrigo Cunha França

**ESTRATÉGIAS E BENEFÍCIOS NA MIGRAÇÃO DO SOFTWARE QUESTÕES PRO
WEB PARA O AMBIENTE MOBILE: UMA ANÁLISE FOCADA NAS VANTAGENS
PARA PROFESSORES E ALUNOS**

Trabalho de Conclusão de Curso apresentado ao Curso Superior de Tecnologia em Telemática, do Instituto Federal de Educação, Ciência e Tecnologia da Paraíba – *Campus* Campina Grande, em cumprimento às exigências parciais para a obtenção do título de Tecnólogo em Telemática.

Orientador: Prof. Dr. Elmano Ramalho Cavalcanti

Campina Grande – PB

2023

F814e França, Rodrigo Cunha

Estratégias e benefícios na migração do software
QuestõesPro *web* para o ambiente *mobile*: uma análise
focada nas vantagens para professores e alunos / Rodrigo
Cunha França. - Campina Grande, 2023.
31f. : il.

Trabalho de Conclusão de Curso (Curso Superior de
Tecnologia de Telemática) - Instituto Federal da Paraíba,
2023.

Orientador: Prof. Dr. Elmano Ramalho Cavalcanti

1. Educação - tecnologia acadêmica 2. Tecnologia
educacional 3. Gestão da informação - tecnologia móvel -
QuestõesPró 4. Desenvolvimento de software -
documentação I. Cavalcanti, Elmano Ramalho II. Título.

CDU 004.7:37

Rodrigo Cunha França

**ESTRATÉGIAS E BENEFÍCIOS NA MIGRAÇÃO DO SOFTWARE QUESTÕES PRO
WEB PARA O AMBIENTE MOBILE: UMA ANÁLISE FOCADA NAS VANTAGENS
PARA PROFESSORES E ALUNOS**

Trabalho de Conclusão de Curso apresentado ao Curso Superior de Tecnologia em Telemática, do Instituto Federal de Educação, Ciência e Tecnologia da Paraíba – Campus Campina Grande, em cumprimento às exigências parciais para a obtenção do título de Tecnólogo em Telemática.

Trabalho aprovado. Campina Grande, 22 de dezembro de 2023:

Prof.Dr. Elmano Ramalho Cavalcanti
Orientador

Prof.Dr. Danyllo Wagner Albuquerque
Membro da Banca

Prof.Dr. Igor Barbosa da Costa Passos
Membro da Banca

Campina Grande – PB

2023

RESUMO

Atividades acadêmicas ainda demandam muito tempo do professor, tais como análise e gerenciamento de desempenho de turmas, alunos, criação de questões, processo de realização e correção de avaliações. Mesmo existindo aplicações na *internet* que fornecem algumas dessas funcionalidades, eles ainda não suprem as necessidades do docente. O software QuestõesPro foi desenvolvido com o intuito de além de possuir as demais funcionalidades de outras aplicações, visa aprimorar ainda mais o ecossistema acadêmico. Os smartphones estão cada vez mais presentes no dia a dia da sociedade, e cada vez mais se aproximam de um computador por executar funções semelhantes. Este trabalho visou o estudo da documentação da aplicação QuestõesPro, e com base nos requisitos foi desenvolvido a migração para o ambiente *mobile*, promovendo a integração do ecossistema acadêmico já fornecida pela atual aplicação, com o ambiente *mobile*.

Palavras-chave: migração de software, QuestõesPro, ambiente *mobile*, educação, dispositivos móveis.

ABSTRACT

Academic activities still demand a significant amount of time from teachers, such as the analysis and management of class and student performance, question creation, and the process of conducting and grading assessments. Despite the existence of internet applications that provide some of these functionalities, they still do not fully meet the needs of educators. The QuestõesPro software was developed not only to encompass the functionalities of other applications but also to further enhance the academic ecosystem. With smartphones becoming increasingly integrated into society's daily life, approaching the capabilities of computers, this study focused on examining the documentation of the QuestõesPro application. Based on the identified requirements, the migration to the mobile environment was implemented, aiming to integrate the academic ecosystem already provided by the current application with the mobile environment.

Keywords: software migration, QuestõesPro, mobile environment, education, mobile devices.

LISTA DE ILUSTRAÇÕES

Figura 1 - ExpoGo instalado no smartphone	28
Figura 2 - Divisão design atômico	29
Figura 3 - Representação de átomos da aplicação	30
Figura 4- Representação de moléculas da aplicação.....	31
Figura 5 - Representação de um organismo da aplicação	31
Figura 6 - Representação de um template	32
Figura 7 - Representação de uma page	32
Figura 8 – Esboço arquitetural da aplicação	35
Figura 9 - Arquitetura do Frontend	37
Figura 10 - Menu principal da aplicação.....	38
Figura 11 - Tela de Login da aplicação	39
Figura 12 - Menu de decisão de fluxo de cadastro.....	39
Figura 13 - Tela de registro da aplicação	40
Figura 14 - Navegação da aplicação	40
Figura 15 - Card representativo de uma turma.....	41
Figura 16 - Formulário de criação e entrada de turma	42
Figura 17 - Página de gerenciamento de turma	43
Figura 18 - Controle de agendamentos a alunos de uma turma	44
Figura 19 - Busca de agendamentos	44
Figura 20 - Botão para criar agendamento.....	45
Figura 21 - Formulário de criação de agendamento.....	46
Figura 22 - Cards de agendamentos	47
Figura 23 - Página de avaliações	47
Figura 24 - Modal de informações de avaliação.....	48
Figura 25 - Formulário de criação de avaliação	49
Figura 26- Formulário de edição de avaliação	50
Figura 27 - Página de gerenciamento de exame.....	50
Figura 28 - Gerenciamento da questão no exame	50
Figura 29 - Tabela, busca e filtros de questões na página de gerenciamento de exame	51
Figura 30 - Filtros da tabela de questões	52
Figura 31 - Modal para iniciar um exame	53
Figura 32 - página de realização do exame	53
Figura 33 - Enunciado e alternativas do exame	54
Figura 34 - Página de confirmação de envio do exame	54
Figura 35 - Página de busca de questões.....	55
Figura 36 - Formulário para criação de questão.....	55
Figura 37 - Modal de informações de questões	57
Figura 38 - Página de edição de informações do usuário	57
Figura 39 - Formulário de edição de informações do usuário	58
Figura 40 - Recurso para remoção de conta	59

LISTA DE TABELAS

Tabela 1 – Atribuições de usuários	26
Tabela 2 - Pastas do projeto	32
Tabela 3 - Tabela de Contextos da aplicação	37
Tabela 4 - Tabela de Fluxo de usuário	38
Tabela 5 - Tipos de usuário	39
Tabela 6 - Estrutura do Card de turmas	41
Tabela 7 - Estrutura header da página de gerenciamento de turma	43
Tabela 8 - Campos do formulário de criação de agendamento	45
Tabela 9 - Campos do Card de agendamento	46
Tabela 10 - Colunas da tabela de avaliações	47
Tabela 11 – Funcionalidade dos botões do modal de avaliação	48
Tabela 12 - Campos do formulário de criação de configuração de avaliação	49
Tabela 13 - Colunas da tabela de questões	51
Tabela 14 - Filtros de questões	52
Tabela 15 - Campos do formulário de criação de questão	56
Tabela 16 - Campos do formulário de edição de informações do usuário	57

SUMÁRIO

1. INTRODUÇÃO.....	9
1.1. Contextualização.....	10
1.2. Problemática	11
1.3. Justificativa.....	12
1.4. Objetivos	12
1.5. Estrutura do trabalho.....	13
2. REFERENCIAL TEÓRICO	15
2.1. Dos Celulares aos <i>Smartphones</i>	15
Trabalho, Educação e Tecnologia: Interfaces na Sociedade Contemporânea	16
2.2. Android e IOS.....	17
2.3. JavaScript e sua Evolução no Desenvolvimento de Aplicações	18
2.4. Node.js, V8, e TypeScript: Fundamentos Tecnológicos Relevantes	19
2.5. A biblioteca React e o <i>Framework</i> React Native	20
2.6. Material Design e React Native Paper	21
2.7. DESIGN SYSTEM E DESIGN ATÔMICO	22
3. METODOLOGIA	24
3.1. Processo de desenvolvimento do software	24
3.2. Levantamento de requisitos	25
3.3. Bases para a criação do sistema	27
3.4. Design System com o Design Atômico	29
3.5. Organização dos diretórios	32
3.6. Arquitetura do Sistema.....	34
3.7. Arquitetura do Frontend	35
4. RESULTADOS	38
4.1. Login e Criação de Conta.....	38
4.2. Navegação da aplicação	40
4.3. Processo de criação, visualização, exclusão, entrada e saída de uma turma. 41	
4.4. Gerenciamento de Turmas.....	43
4.5. Agendamentos	44
4.5.1. Visualização de agendamentos para usuário professor	44
4.5.2. Visualização de agendamentos para usuário aluno	46
4.6. Avaliações.....	47

4.6.1. Avaliações para o usuário professor.....	47
4.6.2. Avaliações para o usuário Aluno	52
4.7. Questões.....	54
4.8. Informações do usuário.....	57
5. CONSIDERAÇÕES FINAIS.....	60
REFERÊNCIAS.....	62

1. INTRODUÇÃO

No contexto contemporâneo do sistema capitalista, a dinâmica do ambiente de trabalho tem sido profundamente impactada, manifestando-se em um notável processo de precarização do trabalho. A atividade docente, imersa nesse cenário, destaca-se como uma das áreas mais suscetíveis aos efeitos das grandes crises econômicas, frequentemente desencadeadoras de reformas trabalhistas. Essas reformas, em muitos casos, resultam na supressão de direitos previamente conquistados por meio de árduas lutas classistas (Barros apud Piovezan, 2017).

Conforme observado por Perez (2014), a natureza flexível do trabalho docente na contemporaneidade é multifacetada, envolvendo demandas incessantes que persistem mesmo durante períodos de férias. A introdução de novas tecnologias tem contribuído para uma jornada de trabalho simultaneamente intensiva, caracterizada pela aceleração na produtividade, e extensiva, refletindo um aumento significativo no tempo dedicado às atividades laborais. É possível encontrar no mercado algumas plataformas que permitem a criação e aplicação *online* de avaliações e o gerenciamento de turmas e banco de questões. No entanto, tarefas que demandam um considerável período por parte do docente (professor), ainda persistem na agenda do professor.

Embora existam no mercado diversas plataformas que viabilizam a criação e aplicação *online* de avaliações, bem como o gerenciamento de turmas e banco de questões, permanecem desafios para os docentes em relação a tarefas que demandam uma parcela considerável de tempo e esforço. Nesse contexto, o projeto QuestõesPro teve origem na Universidade Federal de Campina Grande, sob a orientação da professora Francilene Procópio Garcia, durante a disciplina de Projeto 1 do curso de bacharelado em Ciência da Computação. O professor Elmano Ramalho Cavalcanti, orientador da presente pesquisa, desempenhou o papel de cliente do projeto. Atualmente, a versão *web* do projeto está em fase de testes e, no momento desta redação, está sendo desenvolvida por estudantes do curso de Engenharia de Computação do IFPB, além disso, pode-se encontrar uma versão do aplicativo web no disponível no endereço <http://questoespro.com.br>, e para o *backend* temos o endereço <https://qpro-back.onrender.com>. Esta pesquisa concentra-se no desenvolvimento e análise do processo de adaptação do QuestõesPro como uma aplicação nativa para um ambiente *mobile*.

1.1. CONTEXTUALIZAÇÃO

Segundo Batista (2013), a popularização dos celulares e o desenvolvimento tecnológico associado têm destacado esses equipamentos em ações relacionadas ao *m-learning* (Schmiedl et al., 2010; Robles et al., 2011; Xie et al., 2011). Pela sua popularidade, os celulares podem contribuir para aumentar o acesso a conteúdos educacionais digitais. Como dispositivos móveis, podem facilitar o aprendizado em contextos fora dos limites das instituições educacionais, bem como dentro das mesmas. Assim, esses dispositivos têm potencial para tornar o aprendizado mais acessível, colaborativo e relevante (Batista apud UNESCO, 2012).

Na contemporaneidade marcada pela era digital, os aplicativos móveis, comumente referidos como *apps*, desempenham um papel de destaque. Projetados meticulosamente para operar em dispositivos móveis, como *smartphones* e *tablets*, esses aplicativos abrangem uma vasta gama de funcionalidades, desde entretenimento, como jogos, até utilidades, como serviços de mensagens e assistência na navegação. Essa revolução tecnológica, rica em possibilidades, redefine a interação humana com a tecnologia e o ambiente circundante (Rodrigues, 2023).

No âmbito educacional, Da Silva Melo e Carvalho (2014) ressaltam as significativas potencialidades proporcionadas pelos dispositivos móveis. Destacam-se a ampliação do acesso a conteúdos pedagógicos e a criação de comunidades de aprendizagem ativa, interativa e colaborativa. Participar dessas comunidades propicia um intercâmbio multicultural, sendo a aposta na interconexão entre diferentes pessoas e culturas, capaz de potencializar a construção de conhecimento dentro e fora da sala de aula. Contudo, o *mobile learning* enfrenta desafios, especialmente devido à percepção frequente de educadores e gestores de que os dispositivos móveis, especialmente os telefones celulares, são prejudiciais ou representam um fator de distração em sala de aula, resultando na proibição de seu uso na maioria das instituições educacionais brasileiras.

Uma das vantagens notáveis das aplicações nativas de um sistema operacional reside em sua capacidade intrínseca de interagir diretamente com os dispositivos do aparelho, como câmeras, acelerômetros e outros sensores. Essa capacidade de interação proporciona um nível de integração que torna a aplicação mais completa e eficaz. Portanto, compreender minuciosamente os recursos que uma

aplicação móvel utilizará torna-se uma etapa crucial do processo de desenvolvimento (DEVMEDIA, 2023).

1.2. PROBLEMÁTICA

No contexto educacional contemporâneo, os docentes enfrentam uma série de desafios que impactam diretamente a eficiência e gestão das atividades acadêmicas. Dentre esses desafios, destacam-se situações práticas que revelam a necessidade premente de soluções inovadoras. A consulta de turmas, agendamento de avaliações e análise de desempenhos dos alunos representam áreas críticas que, muitas vezes, carecem de ferramentas tecnológicas específicas para otimização e facilitação.

A consulta de turmas emerge como um desafio recorrente, considerando a complexidade da gestão de informações sobre diferentes grupos de estudantes. Docentes frequentemente lidam com salas numerosas e variadas, tornando vital uma solução que ofereça uma visão consolidada e de fácil acesso às características de cada turma.

O agendamento de avaliações constitui outra área de desafio para os docentes. A logística envolvida na marcação de datas, disponibilidade de salas e comunicação eficaz com os alunos demanda uma ferramenta específica que possa simplificar e organizar esse processo, proporcionando maior eficiência e redução de possíveis conflitos de horários.

A análise de desempenhos dos alunos é um aspecto crucial para avaliação contínua e personalização do ensino. No entanto, a coleta, organização e interpretação desses dados muitas vezes se tornam morosas e complexas. Uma solução tecnológica que simplifique essa análise, oferecendo *insights* de maneira ágil e compreensível, representa uma resposta a esse desafio.

Diante desse panorama, a problemática que norteia o presente Trabalho de Conclusão de Curso (TCC) reside na carência de uma ferramenta que integre essas funcionalidades essenciais em um aplicativo dedicado aos docentes. Como desenvolver uma aplicação que permita aos docentes consultar turmas de forma intuitiva, agendar avaliações de maneira eficiente e analisar desempenhos dos alunos de forma simplificada? A resposta a essa problemática se torna essencial para aprimorar a eficiência e a gestão das atividades acadêmicas, alinhando-se às

demandas educacionais contemporâneas e proporcionando uma solução tecnológica adaptada às necessidades específicas da comunidade acadêmica.

1.3. JUSTIFICATIVA

A pesquisa sobre a relevância do projeto QuestõesPro no contexto mais amplo da educação e tecnologia se mostra crucial diante das transformações significativas que a sociedade contemporânea vivência, especialmente no que tange à integração de tecnologias na prática educacional. A crescente ubiquidade dos dispositivos móveis e a evolução tecnológica proporcionam um cenário propício para explorar inovações que possam aprimorar o processo de ensino-aprendizagem.

No âmbito educacional, a utilização de aplicativos móveis tem se destacado como uma ferramenta potencialmente eficaz para engajar alunos, personalizar o aprendizado e facilitar o acesso a recursos educacionais (SOUZA et al., 2016). O projeto QuestõesPro apresenta-se como uma iniciativa relevante nesse contexto, oferecendo soluções que visam otimizar a criação, aplicação e gerenciamento de avaliações, contribuindo para a eficiência do processo educativo.

Além disso, a pesquisa sobre a relevância do QuestõesPro se alinha aos esforços para compreender como a tecnologia pode ser incorporada de maneira efetiva no ambiente educacional, proporcionando subsídios para práticas pedagógicas mais inovadoras e alinhadas às demandas contemporâneas. A investigação se propõe a avaliar não apenas o impacto imediato do projeto nas práticas de avaliação, mas também seu potencial para influenciar positivamente a dinâmica da educação, promovendo uma abordagem mais dinâmica e adaptativa ao ensino.

Dessa forma, a presente pesquisa busca justificar a relevância do projeto QuestõesPro não apenas como uma ferramenta específica, mas como parte integrante de um movimento mais amplo em direção à incorporação efetiva da tecnologia no contexto educacional, visando melhorias substanciais na qualidade e eficácia do processo educativo.

1.4. OBJETIVOS

Este trabalho tem por objetivo geral adaptar o QuestõesPro *web* para um ambiente mobile, contribuindo assim para a eficiência das tarefas docentes e para uma experiência de ensino e aprendizado mais completa e integrada.

Os objetivos específicos da presente pesquisa estão descritos de forma detalhada a seguir:

- Analisar a documentação existente do projeto QuestõesPro *web*;
- Identificar quais as melhores tecnologias para desenvolvimento e migração do QuestõesPro *web* para o ambiente *mobile*;
- Realizar a adaptação do projeto Web para ambientes *mobile*, abrangendo os sistemas operacionais iOS e Android.

1.5. ESTRUTURA DO TRABALHO

Os capítulos restantes que compõem este documento estão organizados da seguinte forma:

Capítulo 2: Referencial Teórico:

- **2.1 Dos Celulares aos *Smartphones*.** Discute o impacto transformador dos *smartphones* na sociedade;
- **2.2 Trabalho, Educação e Tecnologia: Interfaces na Sociedade Contemporânea.** Discute a interconexão entre trabalho, educação e tecnologia da informação na sociedade contemporânea;
- **2.3 Android vs iOS.** Discute a evolução dos dispositivos móveis, desde os celulares convencionais até a ascensão dos *smartphones*;
- **2.4 Javascript.** Aborda o JavaScript e apresenta uma breve análise sobre esse assunto.
- **2.5 Node e Typescript.** Discute o Node destacando sua importância como um ambiente de tempo de execução JavaScript, junto com a tipagem estática aplicada com o Typescript que são amplamente utilizados para o desenvolvimento de aplicativos e servidores web eficientes.
- **2.6. A biblioteca *React* e o *Framework React Native*.** Enfatiza a importância do *React Native* como um *framework* JavaScript usado para desenvolver aplicativos móveis nativos para iOS e Android, oferecendo muita reutilização de código e proporcionando uma experiência de usuário rica e ágil.
- **2.7. *Material Design* e *React Native Paper*.** Discute sobre bibliotecas para agilizar o desenvolvimento de componentes.

- **2.8. Design System e o Design Atômico.** Enfatiza a importância de escolha de um *design system* para o desenvolvimento de uma aplicação.

Capítulo 3: Metodologia – Apresenta-se, nesta seção, a configuração metodológica adotada, delineando os passos efetuados para a construção deste estudo.

Capítulo 4: Demonstração – Apresenta-se os resultados obtidos ao final do processo metodológico.

Capítulo 5: Considerações finais e trabalhos futuros – Nesta seção, são apresentadas as principais conclusões e desdobramentos decorrentes da realização deste estudo.

2. REFERENCIAL TEÓRICO

Nesta seção, são apresentadas as bases teóricas utilizadas no desenvolvimento deste trabalho, juntamente com a explicação de algumas tecnologias que são relevantes para a discussão proposta.

São explorados os componentes essenciais ao desenvolvimento de uma aplicação *mobile*, bem como são fornecidas referências de trabalhos relacionados. Ao longo deste capítulo, são fornecidas as informações necessárias para compreender e contextualizar a discussão sobre desenvolvimento *mobile*.

2.1. DOS CELULARES AOS SMARTPHONES

A compreensão do impacto dos dispositivos móveis, especialmente *smartphones*, na sociedade contemporânea requer uma análise abrangente das transformações ocasionadas por esses aparelhos. Segundo Sato (2016), em um contexto atual, computadores e celulares são considerados janelas pelas quais os usuários se conectam ao mundo, gerando novas dimensões de espaço e tempo e contribuindo para a formação de uma identidade na denominada "Sociedade em Rede" (CASTELLS, 1999).

Os telefones celulares, disponíveis desde a década de 1980, tiveram seu papel expandido com os avanços tecnológicos em capacidade de processamento, tamanho, design e velocidade (BORTOLAZZO, 2020). Eles passaram de meros aparelhos de telefonia para ferramentas multifuncionais, permitindo atividades como ouvir música, jogar, assistir filmes e efetuar pagamentos de contas.

O conceito de *smartphone*, conforme definido por Bortolazzo (2020), diferencia-se por sua capacidade de executar tarefas semelhantes às de um computador, enfatizando seu caráter inteligente. Verkasalo (2008) destaca que esses dispositivos amalgamam serviços de interação social com aplicativos similares aos de computadores, enquanto Zheng e Ni (2006) destacam sua posição como uma convergência entre telefone celular e assistente digital, o PDA (*Personal Digital Assistant*).

Esses dispositivos, considerados agentes de transformação e revolução, desempenham um papel crucial na sociedade contemporânea, oferecendo singularidade aos usuários, facilitando o acesso à informação e proporcionando benefícios como portabilidade. Os *smartphones* influenciam significativamente as

capacidades cognitivas, com impactos notáveis nas esferas acadêmica, profissional e cotidiana (REINALDO et al., 2016).

2.2. TRABALHO, EDUCAÇÃO E TECNOLOGIA: INTERFACES NA SOCIEDADE CONTEMPORÂNEA

O projeto QuestõesPro encontra fundamentos e justificativas sólidas nos contextos sociológicos e educacionais delineados pelo referencial teórico apresentado. A análise do ato de trabalhar, à luz das observações de Marx (1982), não se restringe a meras atividades laborais, mas se estende a um processo no qual os indivíduos interagem com a natureza, manifestando um movimento simultaneamente externo e interno. Esse entendimento é crucial ao considerarmos a integração da tecnologia da informação no ambiente educacional, pois ressalta a natureza dinâmica das relações entre seres humanos e suas ferramentas de trabalho.

A atividade educativa, como destacado por Arroyo (apud PEREZ), é percebida como um encontro entre os mestres do viver e do ser, delineando a importância do processo educacional na evolução da humanidade. O projeto QuestõesPro alinha-se a essa perspectiva ao buscar proporcionar ferramentas tecnológicas que auxiliam tanto educadores quanto aprendizes, promovendo um ambiente educacional enriquecedor.

No cenário contemporâneo, a ascensão dos dispositivos móveis e a disseminação das Tecnologias de Informação e Comunicação (TICs) transformaram significativamente a dinâmica da sociedade, conforme observado por Perez (2013). O conceito de *mobile learning*, definido por Mülbert e Pereira (2011), destaca a entrega de aprendizado por meio de dispositivos móveis, abrindo novas possibilidades no campo educacional. O QuestõesPro incorpora essa tendência, reconhecendo a natureza convergente, portátil e multimídia dos dispositivos móveis como facilitadores da aprendizagem.

No ambiente educacional, a automação de atividades, especialmente a avaliação, é reconhecida como uma prática valiosa (SILVA, 2019). O QuestõesPro se alinha a essa perspectiva, proporcionando uma ferramenta que otimiza processos manuais, permitindo maior agilidade e eficiência na avaliação, abrangendo desde o ensino fundamental até o ensino superior.

A inserção das TICs como recursos pedagógicos oferece uma nova perspectiva para o ensino, possibilitando a transição do novo contexto social para o

contexto social tecnológico. O potencial transformador da tecnologia móvel no contexto educacional, ilustrado por projetos estrangeiros como Apple© e Google©, ressalta a necessidade de adaptação e inovação no cenário brasileiro. O QuestõesPro emerge como uma resposta a essas demandas, apresentando-se como uma solução alinhada às pressões para adoção da informática, influenciadas pela opinião pública, comércio e propaganda.

2.3. ANDROID E IOS

Ao mergulharmos na análise dos sistemas operacionais *mobile*, Android e iOS, é crucial contextualizar sua relevância no panorama dos dispositivos móveis contemporâneos. Esses sistemas, concebidos pela Google e Apple, respectivamente, desempenham um papel central na experiência dos usuários de *smartphones* e *tablets*, oferecendo abordagens distintas, funcionalidades específicas e ecossistemas próprios. Para uma compreensão mais profunda, é imperativo iniciar nossa investigação com uma abordagem sobre o papel dos sistemas operacionais em dispositivos computacionais como um todo.

No cerne de qualquer sistema computacional moderno, encontramos uma intrincada composição de componentes que incluem processadores, disco rígido, impressoras, teclado e monitor. Os sistemas operacionais, por sua vez, assumem uma função vital nesse ecossistema, gerenciando esses componentes e recursos essenciais para o funcionamento do computador. Além disso, fornecem uma interface gráfica que facilita a interação do usuário (TANENBAUM, 2008). Esta compreensão inicial é fundamental para estabelecer as bases necessárias para a análise mais específica dos sistemas operacionais voltados para dispositivos móveis.

O iOS, elaborado pela Apple, se configura como um Sistema Operacional Móvel específico para os dispositivos da marca. Derivado do macOS X, o iOS foi projetado para atender às demandas dos dispositivos móveis da Apple. A constante evolução desses sistemas é evidenciada pelas melhorias introduzidas em suas versões mais recentes, como o iOS 17, lançado em setembro de 2023, que trouxe aprimoramentos na comunicação, compartilhamento de conteúdo e experiências inovadoras (LOPES, 2023). Essas características fundamentais delineiam a importância dos sistemas operacionais móveis na dinâmica dos dispositivos móveis na sociedade contemporânea.

Os aplicativos iOS são desenvolvidos na linguagem de programação *Objective-C*, a qual incorpora características da linguagem *Smalltalk* e da linguagem C. O programador iOS deve estar atento ao gerenciamento de memória, uma vez que essa linguagem de programação está intimamente relacionada à família de linguagens C. Vale ressaltar que o desenvolvimento para iOS é restrito, uma vez que o ambiente de desenvolvimento é acessível somente em computadores Mac executando o sistema operacional Mac OS X. É importante observar que a quantidade de computadores Mac disponíveis é inferior em comparação com o número de dispositivos que utilizam outros sistemas operacionais (MENDONÇA, BITTAR, DIAS apud GOADRICH, 2011).

No contexto dos sistemas operacionais móveis, o Android se destaca como um sistema de código aberto desenvolvido pela Google. Baseado na versão 2.6 do *kernel* Linux, o Android utiliza essa estrutura para controlar funções cruciais do sistema, como segurança, gerenciamento de memória, processos, pilha de rede e modelo de driver (GOOGLE INC., 2011). A gestão do Android foi transferida para a Open Handset Alliance (OHA), fortalecendo sua posição no cenário dos sistemas operacionais móveis (MENDONÇA, BITTAR e DIAS, 2011).

Ao contrário do iOS, o desenvolvimento para Android é multiplataforma, podendo ser realizado em qualquer um dos principais sistemas operacionais, como Windows (XP ou superior), Mac OS X (10.5.8 ou superior) e sistemas Linux (executando com *kernel* 2.6 ou superior). Essa abordagem proporciona uma maior abrangência para os programadores e a viabilidade de desenvolvimento independente do sistema operacional. Além da flexibilidade em relação ao sistema operacional, o programador tem a liberdade de escolher o Ambiente Integrado de Desenvolvimento (IDE) de sua preferência para o processo de desenvolvimento.

2.4. JAVASCRIPT E SUA EVOLUÇÃO NO DESENVOLVIMENTO DE APLICAÇÕES

O JavaScript, linguagem de *script* interpretada pelos navegadores, evoluiu consideravelmente ao longo do tempo, expandindo seu escopo de atuação. Originalmente, essa linguagem desempenhava funções específicas nos navegadores, como requisições assíncronas para obtenção de dados e alterações dinâmicas de CSS. Contudo, sua utilização era restrita ao ambiente web. Com o advento do Node.js,

essa restrição foi superada, e o JavaScript passou a ser empregado em diversas aplicações, transcendendo o âmbito da web.

O JavaScript, como linguagem de alto nível, desempenha agora papéis diversos, desde a criação de microsserviços até ferramentas de automação de tarefas, exemplificadas por Grunt, Gulp e Webpack. Além disso, integra ambientes de desenvolvimento como o Visual Studio Code da Microsoft. Esse cenário promoveu um significativo crescimento da comunidade JavaScript, notavelmente caracterizada pelo modelo *open-source*, fundamentado em código aberto e licenciamento livre (ANJOS, 2017).

Essa evolução ressalta a versatilidade do JavaScript, evidenciando sua capacidade de adaptação a diferentes contextos e necessidades no universo do desenvolvimento de software. A transição do JavaScript de uma linguagem restrita nos navegadores para uma ferramenta essencial em variados cenários ilustra seu papel fundamental na atualidade.

2.5. NODE.JS, V8, E TYPESCRIPT: FUNDAMENTOS TECNOLÓGICOS RELEVANTES

O projeto QuestõesPro encontra sustentação em um referencial teórico consistente, refletindo a integração complexa entre tecnologias de desenvolvimento. O Node.js, conforme analisado por Tilkov e Vinoski (2010), é apresentado como um ambiente de desenvolvimento JavaScript notável por sua ênfase em desempenho e baixo consumo de memória. Esta característica, compartilhada com a implementação V8 da Google, destaca o Node.js como um ambiente eficiente, especialmente em cenários de servidores. Este conceito, alinha-se com a proposta do QuestõesPro ao oferecer uma plataforma eficaz para a execução de processos em ambientes estendidos.

A *engine* JavaScript V8, originária do navegador Chrome, desempenha um papel crucial no funcionamento do JavaScript no lado do cliente. Sua incorporação pelo Node.js, conforme observado por Paulino (2018), amplia sua aplicabilidade para contextos de servidores. A abordagem adotada pelo Node.js em relação ao V8 destaca a versatilidade dessa engine, flexível o suficiente para se adaptar a diferentes contextos de aplicação, característica essencial para um ambiente de desenvolvimento ágil e adaptável.

Além da relação com o V8, o Node.js se conecta ao TypeScript, conforme mencionado por Heller (2017). O TypeScript, por sua vez, apresenta a capacidade de ser transpilado para JavaScript, tornando-se uma escolha prática no ambiente Node.js. A incorporação do TypeScript como um superconjunto do JavaScript, com recursos avançados de tipagem, evidencia uma abordagem proativa para a detecção de erros, conforme ressaltado por Dreimanis (2020). Esta funcionalidade, fundamental para garantir a qualidade do código, alinha-se com os propósitos do QuestõesPro ao buscar aprimorar a eficiência e confiabilidade das soluções oferecidas.

Segundo Cruz (2017), as tecnologias Node.js, V8 e TypeScript são importantes para o desenvolvimento de aplicações. Além da eficiência técnica proporcionada por essas tecnologias, destaca-se a integração flexível de diferentes elementos no universo do JavaScript. Essa característica reflete diretamente a abordagem do QuestõesPro, que busca oferecer soluções adaptáveis e eficazes no contexto de desenvolvimento de aplicações web.

2.6. A BIBLIOTECA REACT E O *FRAMEWORK* REACT NATIVE

O projeto QuestõesPro se fundamenta em uma compreensão aprofundada das tecnologias envolvidas no desenvolvimento de aplicações web, destacando o React e o React Native como elementos essenciais para a consecução dos objetivos propostos.

Desenvolvido por Jordan Walke, engenheiro de software do Facebook, o React, conforme observado por Ferreira e Zuchi (2018), destaca-se por ser mantido pela própria empresa, em colaboração com o Instagram e uma comunidade de desenvolvedores individuais (Mariano, 2017). Conforme consta na documentação oficial do React (2018), é uma biblioteca de Interface de Usuário (UI), o React concentra sua atuação na camada de visualização do Modelo-Visão-Controlador (MVC), apresentando alto desempenho ao atualizar o *Document Object Model* (DOM) de maneira eficiente. Sua natureza declarativa incentiva os desenvolvedores a concentrarem-se no resultado, não na abordagem, tornando-o ideal para ser utilizado em conjunto com outras bibliotecas ou *frameworks*. No contexto do QuestõesPro, o React assume um papel central na criação de interfaces de usuário eficazes e dinâmicas.

Ferreira e Zuchi (2018) também ressaltam a complexidade das decisões diárias enfrentadas por desenvolvedores de software, destacando a multiplicidade de

opções de linguagens de programação, *frameworks* e ferramentas disponíveis. A escolha de um *framework*, conforme Mariano (2017), vai além da adequação às necessidades de um projeto, envolvendo a entrega de um código de alta qualidade e desempenho otimizado. Nesse contexto, o React se destaca como uma escolha relevante, proporcionando uma abordagem eficiente e flexível para o desenvolvimento de interfaces de usuário.

Eisenman (2015) introduz o React Native como um *framework* JavaScript específico para a criação de aplicações móveis nativas para iOS e Android. Baseado no React, essa ferramenta permite aos desenvolvedores criar aplicativos móveis com uma aparência genuinamente nativa, aproveitando os mesmos componentes fundamentais de Interface de Usuário (UI) utilizados em aplicativos móveis convencionais para iOS. Silva (2018) destaca que o React Native representa uma abordagem única, diferenciando-se de aplicativos web móveis, aplicativos HTML5 ou aplicativos híbridos, proporcionando uma experiência autêntica para os usuários.

Assim, o React e do React Native, são elementos centrais no contexto do projeto QuestõesPro. Essas tecnologias emergem como pilares fundamentais para o desenvolvimento de aplicações web eficientes e inovadoras.

2.7. MATERIAL DESIGN E REACT NATIVE PAPER

O *Material Design*, é um sistema de design abrangente concebido e mantido pelo Google, como um elemento fundamental para a criação de interfaces de usuário inovadoras e visualmente atraentes.

Desenvolvido por designers e desenvolvedores do Google, o *Material Design* oferece um conjunto abrangente de diretrizes de experiência do usuário (UX) e implementações de componentes de interface do usuário (UI) para Android, Flutter e a Web (Google, 2023). A versão mais recente, Material 3, introduz experiências personalizadas, adaptativas e expressivas, incorporando elementos como cores dinâmicas, acessibilidade aprimorada e fundamentos para layouts em telas grandes, além de *tokens* de design (Google, 2023). No contexto do projeto QuestõesPro, o *Material Design* emerge como uma base sólida para a criação de interfaces intuitivas e esteticamente agradáveis.

O *Material Design*, como sistema adaptável, é composto por diretrizes, componentes e ferramentas que respaldam as melhores práticas no design de interfaces de usuário. Sua natureza de código aberto promove a colaboração eficiente

entre designers e desenvolvedores, possibilitando a criação ágil de produtos visualmente atrativos (Google, 2023). No contexto do projeto QuestõesPro, a utilização do Material Design se alinha à busca por interfaces consistentes, intuitivas e modernas.

No âmbito do React Native, o React Native Paper se destaca como uma coleção de componentes prontos para produção e personalizáveis, alinhados às diretrizes de *Material Design* do Google. Esses componentes oferecem uma interface responsiva e ágil em ambas as plataformas, contribuindo para a criação de aplicativos visualmente coesos. No entanto, a estilização de cada componente em um ambiente React Native pode ser desafiadora e complexa, exigindo conformidade rigorosa com as diretrizes da plataforma de destino (React Native Paper, 2023). Assim, o *Material Design* e suas implementações, como o React Native Paper, representam elementos essenciais para a concepção e desenvolvimento de interfaces consistentes e visualmente atraentes no escopo do projeto QuestõesPro.

2.8. DESIGN SYSTEM E DESIGN ATÔMICO

O projeto QuestõesPro incorpora conceitos fundamentais, como o *design system* e o Atomic Design, que desempenham um papel crucial na concepção e desenvolvimento de interfaces coesas e eficientes.

O *design system*, conforme definido por Quevedo et al. (2023), representa um conjunto de diretrizes, recursos e padrões que orientam a criação de interfaces. Essa abordagem abrange diversos elementos, como visuais, tipografia, cores, espaçamentos e interações, proporcionando uma base para o desenvolvimento consistente. Criado com a intenção de ser reutilizável e escalável, o *design system* é essencial para auxiliar as equipes de design na criação eficiente de interfaces, mantendo a qualidade e a consistência. Além disso, contribui para a padronização de elementos em uma marca ou aplicação, resultando em melhorias significativas na usabilidade e na experiência do usuário. No contexto do projeto QuestõesPro, a implementação de um *design system* emerge como uma prática essencial para garantir interfaces coesas e eficientes em todas as interações.

O Atomic Design, proposto por Brad Frost e fundamentado em analogias com a química, introduz uma metodologia flexível e escalável para o desenvolvimento de sistemas de componentes de interface. Essa abordagem estabelece uma hierarquia entre elementos, classificando-os como átomos, moléculas, organismos,

templates e páginas em ordem crescente (Quevedo et al., 2023). O conceito de Atomic Design proporciona uma visão sistêmica e estruturada para o desenvolvimento de interfaces, permitindo que as equipes de design construam sistemas coesos e modularmente escaláveis. No âmbito do projeto QuestõesPro, a aplicação do Atomic Design oferece uma metodologia eficaz para a criação e manutenção de sistemas de design que evoluem de maneira organizada e eficiente.

Ambos os conceitos, *design system* e Atomic Design, desempenham papéis sinérgicos, sendo o *design system* uma implementação prática e abrangente, enquanto o Atomic Design fornece uma metodologia estruturada para a organização e desenvolvimento de componentes de interface. Essa combinação estratégica contribui diretamente para os objetivos do projeto QuestõesPro, garantindo consistência, escalabilidade e eficiência no desenvolvimento de interfaces de usuário inovadoras e de alta qualidade.

3. METODOLOGIA

Este capítulo descreve a metodologia empregada na migração e desenvolvimento do software QuestõesPro para o ambiente *mobile*. A metodologia está estruturada em três etapas distintas, a saber: Levantamento de Requisitos, Definição da Arquitetura do Sistema e Implementação do Desenvolvimento.

No primeiro capítulo, é enfatizado o desenvolvimento do projeto. São detalhados os processos, métodos e abordagens utilizados para coletar informações relevantes que direcionam o desenvolvimento do aplicativo QuestõesPro para o ambiente *mobile*. Teremos ainda uma subseção que aborda o processo de estudo detalhado da documentação do projeto QuestõesPro na versão *web*. O objetivo é obter uma compreensão profunda das funcionalidades, fluxos de trabalho e requisitos da versão existente, a fim de atender às necessidades do usuário final no ambiente *mobile*.

A segunda parte deste capítulo concentra-se na arquitetura final do sistema *frontend* para o ambiente *mobile*. Aqui, são estabelecidas as bases para a criação do sistema, incluindo a seleção de elementos-chave e o planejamento da estrutura do projeto. Será definido qual *design system* é mais apropriado para garantir a consistência da interface do usuário no ambiente *mobile*. A escolha se baseia em critérios de adaptação e reutilização de componentes do sistema.

Na terceira e última parte, a estrutura organizacional do projeto é delineada. Isso inclui a definição da organização de diretórios e a implementação de padrões de codificação para garantir uma arquitetura robusta, de fácil manutenção e escalabilidade.

3.1. PROCESSO DE DESENVOLVIMENTO DO SOFTWARE

Na primeira etapa do processo de desenvolvimento, realizamos uma análise abrangente do projeto *frontend* QuestõesPro *web*, considerando tanto o público-alvo de alunos quanto de professores. O objetivo principal dessa análise foi garantir que a migração do projeto ocorresse sem impactos negativos na experiência do usuário. Para isso, foram realizadas as seguintes atividades:

- Avaliação de todos os recursos existentes no projeto QuestõesPro *web*, identificando elementos que seriam migrados para a versão *mobile*;

- Análise da documentação já existente do projeto, com foco na identificação de requisitos e informações relevantes para aprimoramentos do sistema;
- Levantamento de requisitos específicos para a versão móvel do QuestõesPro, considerando as expectativas e necessidades dos usuários.

Dada a natureza de migração do projeto, não houve a necessidade de desenvolver um novo *backend*. Optou-se por reutilizar o *backend* já existente do QuestõesPro que se mostrou uma base sólida para o projeto móvel. Nesta etapa, conduzimos as seguintes atividades:

- Identificação e mapeamento de todos os *endpoints* existentes no projeto QuestõesPro Web;
- Definição dos parâmetros de rota necessários para o funcionamento do aplicativo móvel;
- Análise dos tipos de requisições utilizadas pelo projeto, a fim de manter a consistência com o *backend* existente.

Na segunda fase do desenvolvimento, concentramo-nos no planejamento e codificação dos recursos do projeto. Priorizamos a definição e o reaproveitamento de recursos que fossem semelhantes para ambos os tipos de usuários (alunos e professores). Esta etapa incluiu as seguintes atividades:

- Identificação de recursos compartilhados entre alunos e professores para otimização do desenvolvimento;
- Planejamento detalhado de como cada recurso seria implementado e adaptado ao ambiente *mobile*;
- Codificação dos recursos, com ênfase na manutenção da experiência do usuário e na eficiência das funcionalidades.

3.2. LEVANTAMENTO DE REQUISITOS

A fase inicial deste processo consistiu na compreensão dos objetivos subjacentes à modelagem e desenvolvimento da migração do projeto QuestõesPro para o ambiente *mobile*. Isso envolveu a análise de documentos existentes e o levantamento de requisitos, a fim de estabelecer clareza sobre o propósito e as metas do projeto. Após a definição dos objetivos, procedemos com a análise detalhada das

histórias de usuário que orientariam o desenvolvimento. Cada história representava um elemento essencial do projeto, conforme Tabela 1.

Tabela 1 – Atribuições de usuários

Entidade	Atribuições
Aluno	Para realizar um cadastro devo fornecer CPF, nome completo, data de nascimento, e senha;
	Devo possuir uma tela para editar minhas informações;
	Devo possuir uma tela para responder uma prova agendada em um determinado tempo permitido;
	Durante a prova devo ter a opção de pular questões, e acompanhar o tempo restante de prova;
	Devo fornecer um código para entrar em uma sala e uma tela apresentando as turmas que eu estou cadastrado;
	Ao acessar uma turma devo ter uma tela apresentando as provas agendadas relacionadas a turma;
	Devo ter a opção de sair de uma turma;
	Devo fornecer CNPJ da instituição, e-mail, nome completo, data de nascimento e senha;
	Devo possuir uma tela para edição dos meus dados;
	Devo possuir a opção para criação de turmas no qual devo fornecer um ano e um nome;
Professor	Devo possuir a opção para criação de turmas no qual devo fornecer um ano e um nome;
	Devo possuir uma tela que lista as turmas criadas por mim, acessar as mesmas, e ter autorização para editá-las;
	Devo possuir uma tela que lista as turmas criadas por mim, acessar as mesmas, e ter autorização para editá-las.
	Devo possuir uma tela para criação de questões, e editá-las, fornecendo: tipo de questão, assunto, enunciado e as alternativas;
	Devo possuir a opção de criar uma avaliação, fornecendo nome da avaliação, nota de corte, nota máxima e número de questões, e a opção de embaralhar as questões;
	Posso listar minhas avaliações;
	Posso listar minhas questões e favoritar as mesmas;
	Posso verificar os agendamentos e alunos por turma;
Posso fazer uma avaliação a partir do banco de questões no qual criei uma questão.	
Usuário	Devo fornecer o e-mail e a senha para entrar no sistema;

Fonte: Elaborado pelo autor, 2023

A separação de recursos do projeto já existente, com escopos mais bem definidos, resultou na definição de dois módulos distintos.

O primeiro módulo, denominado "Módulo Alunos", engloba todas as histórias de usuário associadas ao público-alvo de alunos. Este módulo visa atender às necessidades específicas dos usuários que se identificam como alunos do sistema. As funcionalidades e recursos abrangidos por este módulo incluem:

- Cadastro de novos alunos;
- Processo de autenticação e login para alunos;
- Listagem de turmas disponíveis;
- Visualização de exames agendados;
- Edição de informações pessoais por parte dos alunos;

O segundo módulo, denominado "Módulo Professores", abrange todas as histórias de usuário associadas ao público de professores que utilizam o sistema. Este módulo foi projetado para atender às necessidades específicas dos usuários identificados como professores e inclui as seguintes funcionalidades:

- Cadastro de novos professores;
- Autenticação e login exclusivos para professores;
- Listagem de turmas sob sua responsabilidade;
- Visualização de questões disponíveis;
- Listagem de agendamentos de exames;
- Listagem de alunos inscritos em suas turmas;
- Cadastro e edição de informações de turmas;
- Criação e edição de exames;
- Criação e edição de questões para exames.

A definição dos módulos proporcionou uma organização mais eficaz do projeto, o que resultou em uma facilitação no gerenciamento das tarefas. Isso assegurou que todos os aspectos da aplicação fossem devidamente considerados e tratados conforme as necessidades do desenvolvimento do projeto.

3.3. BASES PARA A CRIAÇÃO DO SISTEMA

Para iniciar o processo de desenvolvimento, foi essencial realizar a preparação do ambiente. Optou-se por utilizar o *runtime* JavaScript Node na sua versão 18 LTS (Long-Term Support) e a linguagem Java na versão 11. Essa escolha foi fundamentada na garantia de compatibilidade com os módulos instalados pelo

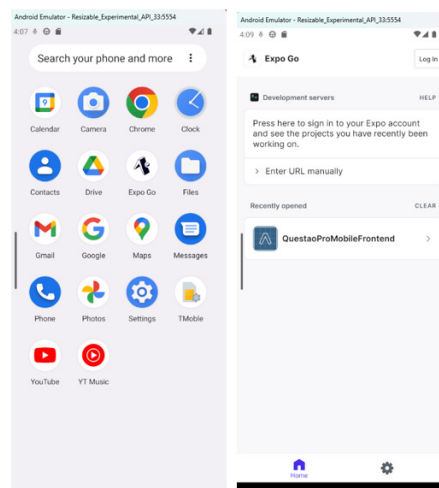
framework React Native, com o intuito de evitar potenciais erros que poderiam surgir ao instalar versões mais recentes dessas tecnologias (React Native, 2023).

De acordo com Igor Almenara (2021), a sigla LTS, que representa “Long-term support” ou “suporte de longo prazo” em português, é uma versão de software que visa principalmente garantir estabilidade ao longo de períodos prolongados.

Conforme recomendação expressa na documentação oficial do React Native, no contexto de um ambiente acadêmico, a escolha recaiu sobre a Expo CLI como a ferramenta de linha de comando (CLI) do projeto. Essa decisão foi motivada, não apenas pela qualidade do suporte oferecido para desenvolvimento, mas também pelos recursos adicionais destinados a novos desenvolvedores no ambiente mobile (React Native, 2023).

Outra necessidade crucial consistiu na configuração de um ambiente de simulação de dispositivo móvel (Figura 1), o qual desempenharia um papel fundamental no processo de desenvolvimento. Para atender a essa necessidade, foi realizado o processo de instalação do Android Studio, incluindo a versão Android 13 Tiramisu. Tal escolha foi fundamentada no atendimento a um requisito estipulado pela documentação oficial do React Native.

Figura 1 - ExpoGo instalado no smartphone



Fonte - Elaborado pelo autor, 2023

Com o Android Studio, Expo Go, Java e Node devidamente configurados, foi iniciado o desenvolvimento no ambiente. Na próxima seção, é abordado o tópico do *design system*.

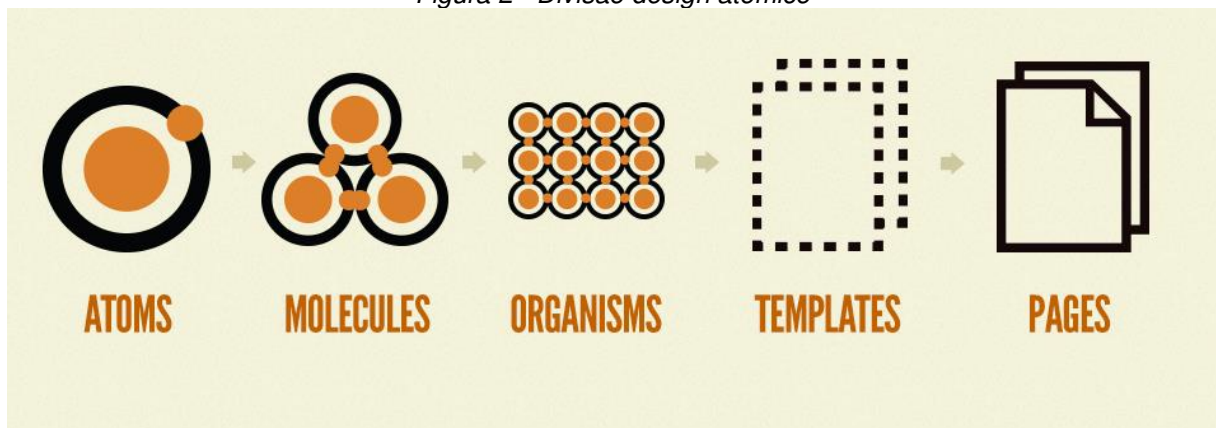
3.4. DESIGN SYSTEM COM O DESIGN ATÔMICO

Frost (2013), propôs uma metodologia com uma abordagem mais estrutural, desenvolvendo um modelo mental denominado design atômico, com o propósito de auxiliar na concepção de interfaces de usuário de forma fragmentada e, simultaneamente, abrangente. Esse modelo propõe uma analogia com a química, baseando-se na premissa de que toda matéria é composta por átomos que se agregam para criar moléculas, que, por sua vez, se combinam para formar organismos mais complexos, culminando na composição das páginas (KRONE 2019 apud FROST 2013).

Segundo Frost (2013), o design atômico proporciona *insights* relevantes que contribuem para a criação de sistemas de design de interface de usuário (UI) mais eficazes e intencionais.

O design atômico é uma metodologia composta por cinco estágios distintos que colaboram harmoniosamente na criação de sistemas de design de interface de maneira deliberada e hierárquica. Os cinco estágios do design atômico compreendem (FROST, 2013): *atoms*, *molecules*, *organisms*, *templates* e por último *pages* (Figura 2).

Figura 2 - Divisão design atômico



Fonte: FROST, 2013

Atoms são elementos de interface de usuário (UI) que não podem mais ser divididos e servem como blocos de construção elementares de uma interface. *Molecules* são coleções de átomos que formam componentes de UI relativamente simples. Os *organisms* são componentes relativamente complexos que formam seções discretas de uma interface. Os *templates* posicionam esses componentes em

um layout e demonstram a estrutura de conteúdo subjacente do design. As *pages* aplicam conteúdo real aos modelos e articulam variações para demonstrar a IU final e testar a resiliência do sistema de design. Algumas das vantagens de usar o modelo atômico, são:

Melhor Usabilidade: Um design bem elaborado torna a interface mais fácil de usar, melhorando a experiência do usuário.

Consistência: Um sistema de design coeso garante que todos os elementos da interface sigam as mesmas diretrizes, criando uma identidade visual consistente.

Eficiência no Desenvolvimento: Reutilizar componentes de design economiza tempo e recursos durante o desenvolvimento.

Melhor Comunicação: Um sistema de design claro facilita a comunicação entre designers, desenvolvedores e partes interessadas.

Testes Aprimorados: A articulação de variações em páginas permite testar a resiliência do sistema de design em diferentes cenários.

Melhor Experiência do Usuário Final: Tudo isso se traduz em uma experiência do usuário final aprimorada, o que pode aumentar a satisfação e a fidelidade do cliente.

A alguns exemplos de *atoms* que serão desenvolvidos em nosso projeto são apresentados a seguir (Figura 3), cada um com suas propriedades, entre as quais a indivisibilidade.

Figura 3 - Representação de átomos da aplicação

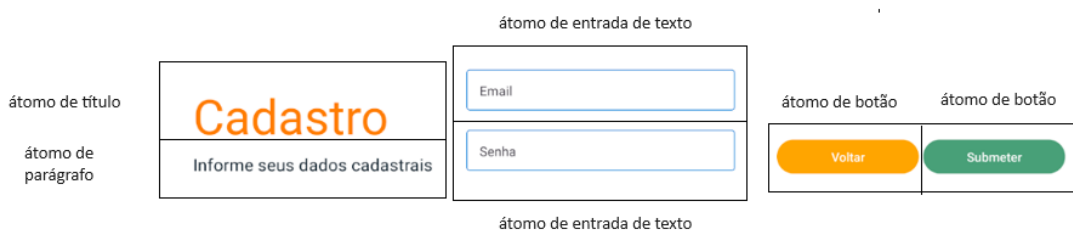


Fonte: Elaborado pelo autor, 2023

É importante observar que os botões “submeter” e “voltar” exibem um formato idêntico, com curvaturas nas bordas, ambos devem executar uma ação, indicando que compartilham a mesma estrutura atômica, divergindo apenas no texto e nas cores.

A seguir, tem-se uma conjunção de *atoms*, resultando em uma *molecule* dentro do nosso aplicativo (Figura 4).

Figura 4- Representação de moléculas da aplicação



Fonte: Elaborado pelo autor, 2023

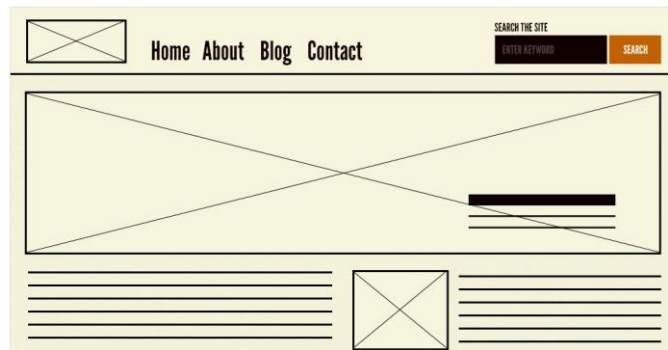
Ao agrupar moléculas, tem-se como resultado nosso organismo, que poderia ser denominado como um formulário de cadastro de usuário (Figura 5).

Figura 5 - Representação de um organismo da aplicação

Fonte: Elaborado pelo autor, 2023

A próxima camada é a de *templates*, cuja concepção não está mais ligada ao conceito de química. É nesta camada que organizamos como será a estrutura da nossa página sem adicionar o conteúdo final, ou seja, podemos criar várias páginas a partir dela aproveitando sua estrutura. Na Figura 6 a seguir o *template* apresenta um organismo de cabeçalho, o corpo da página com seus respectivos organismos, porém seu conteúdo final ainda não foi adicionado.

Figura 6 - Representação de um template



Fonte: FROST, 2013

Por fim, chegamos à última camada que é a de *pages*. Nesta camada é adicionado de fato todo o conteúdo da página, como ilustrado na Figura 7.

Figura 7 - Representação de uma page



Fonte: FROST, 2013

Com as vantagens do design atômico apresentadas, temos o modelo de *design system* a ser aplicado na migração do *software* QuestõesPro.

3.5. ORGANIZAÇÃO DOS DIRETÓRIOS

A organização de pastas em um projeto de *software* desempenha um papel crucial para a eficiência, manutenção e colaboração bem-sucedida. Com isso, a estrutura de pastas do projeto foi separada em: *api*, *assets*, *auth*, *components*, *context*, *hooks*, *models*, *pages*, *providers*, *stacks*, *theme*, *utils*, segundo a Tabela 2.

Tabela 2 - Pastas do projeto

Nome da pasta	Atribuições
---------------	-------------

api	Pode haver arquivos que contenham configurações específicas relacionadas à comunicação com a API, como a URL base da API, chaves de autenticação, entre outros. A pasta "api" pode conter funções ou utilitários responsáveis por fazer chamadas HTTP à API. Essas funções podem incluir manipulação de cabeçalhos, tratamento de erros, etc.
assets	Qualquer imagem, ou fonte que seja utilizada na interface do usuário pode ser armazenada na pasta "assets". Exemplos: Logotipos, ícones, imagens.
auth	Este módulo abrange funcionalidades destinadas à persistência local de informações do usuário no que se refere à autenticação.
components	Este módulo engloba alguns dos módulos do design atômico (<i>atoms</i> , <i>molecules</i> , <i>organisms</i> e <i>templates</i>). Adicionalmente, incluirá um módulo de componentes de layout, responsáveis pelo posicionamento dos elementos e estilização personalizada, um módulo de componentes globais, os quais podem ser invocados em qualquer outra componente sem seguir padrões do design atômico.
context	Contém a implementação dos contextos utilizando a Context API do React. Esses contextos são utilizados para gerenciar o estado global da aplicação e compartilhar dados entre componentes sem a necessidade de passar as propriedades manualmente através da hierarquia de componentes.
hooks	É utilizada para armazenar <i>hooks</i> personalizados. <i>Hooks</i> são funções especiais do React que permitem a você usar o estado e outros recursos do React em componentes de função. Ao criar <i>hooks</i> personalizados, você encapsula lógica específica em funções reutilizáveis, promovendo a modularidade e a reutilização de código.
models	É utilizada para armazenar definições de modelos ou tipos de dados que representam a estrutura de dados específica da aplicação. Esses modelos podem ser usados para representar objetos, entidades ou dados que são manipulados pela aplicação.
pages	Utilizada para organizar os componentes que representam as páginas individuais da aplicação. Cada arquivo dentro desta pasta geralmente corresponde a uma página específica da aplicação, e esses componentes de página costumam conter a lógica e a estrutura visual específicas para aquela rota ou seção da aplicação.

providers	Utilizada para armazenar os provedores de contexto. Provedores de contexto são componentes que fornecem o contexto para outros componentes da aplicação. Essa prática é especialmente comum quando se utiliza a Context API do React para gerenciar o estado global e compartilhar dados entre componentes.
stacks	Contém configurações ou componentes específicos relacionados à navegação em pilha.
theme	Usada para armazenar definições de temas ou estilos que são aplicados globalmente à interface do usuário. Essa abordagem é particularmente útil quando você deseja centralizar as configurações de estilo para garantir consistência em toda a aplicação.
utils	Utilizada para armazenar utilitários, funções auxiliares ou módulos que não se encaixam em uma categoria específica e são compartilhados em toda a aplicação. Essa pasta é comumente usada para manter código reutilizável e lógica que não está diretamente associada a componentes ou funcionalidades principais.

Fonte: Elaborado pelo autor, 2023

3.6. ARQUITETURA DO SISTEMA

A arquitetura (Figura 8) do projeto QuestõesPro foi concebida com o objetivo de proporcionar uma interface intuitiva e eficiente para os usuários, integrando-se harmoniosamente com o já construído *backend* em Spring Boot e o banco de dados PostgreSQL.

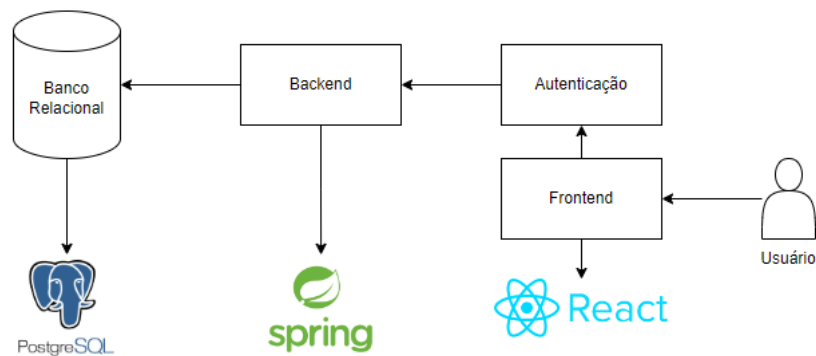
A arquitetura foi estruturada para uma perfeita integração com o *backend* desenvolvido em Spring Boot. Essa escolha, pelo reaproveitamento do *backend* foi motivada pela robustez, flexibilidade e eficiência que o Spring Boot oferece no desenvolvimento de aplicativos Java. A comunicação entre o *frontend* e o *backend* ocorre de maneira eficaz por meio de APIs RESTful, garantindo a transferência segura de dados e a interação fluida entre as camadas da aplicação.

Para a persistência e recuperação de dados, optou-se por manter o sistema atual de gerenciamento de banco de dados, o PostgreSQL, visto que o mesmo atende as necessidades do projeto. O PostgreSQL é reconhecido por sua confiabilidade, escalabilidade e suporte a recursos avançados, sendo uma escolha sólida para aplicações que demandam alto desempenho e integridade dos dados.

O *frontend* foi implementado utilizando o React, uma biblioteca JavaScript conhecida por sua flexibilidade, eficiência e facilidade de manutenção. A escolha do React possibilitou a criação de interfaces de usuário dinâmicas e responsivas, contribuindo para uma experiência de usuário aprimorada.

Em resumo, a arquitetura do *frontend* no projeto QuestõesPro foi meticulosamente planejada para garantir uma integração eficaz com o *backend* construído com Spring Boot e PostgreSQL, onde o React foi empregado para criar uma interface de usuário moderna e dinâmica. O uso de práticas como componentização e a React Context API contribuiu para a criação de um *frontend* robusto, escalável e orientado à experiência do usuário.

Figura 8 – Esboço arquitetural da aplicação



Fonte: Elaborado pelo autor, 2023

3.7. ARQUITETURA DO FRONTEND

A ContextAPI é uma funcionalidade incorporada à biblioteca React, amplamente utilizada para o desenvolvimento de interfaces de usuário em JavaScript. Essa API oferece uma abordagem que simplifica o compartilhamento de dados, como estados e métodos, entre componentes na hierarquia do React, eliminando a necessidade de passagem explícita de *props* por diversos níveis. Composta pelos componentes Provider (provedor) e Consumer (consumidor), a ContextAPI permite criar um contexto global que pode ser acessado por componentes específicos sem a complexidade de propagação manual de *props* através de intermediários desnecessários.

O componente Provider é responsável por envolver a seção superior da árvore de componentes, onde os dados globais, como estados, precisam estar

disponíveis. Por meio da propriedade *value*, o Provider define o valor do contexto compartilhado. Já o componente Consumer é utilizado nos componentes que desejam consumir esses dados globais, utilizando uma função de renderização que recebe o valor do contexto como argumento. A Context API, ao simplificar a comunicação entre componentes, torna-se especialmente útil em cenários nos quais diversos componentes necessitam acessar os mesmos dados globais, aprimorando a eficiência e a organização do estado em aplicações React.

A arquitetura do *frontend* no projeto QuestõesPro foi cuidadosamente delineada para proporcionar uma experiência de usuário fluida e eficiente, atendendo às demandas específicas do contexto educacional. No desenvolvimento do *frontend*, a escolha estratégica recaiu sobre a utilização da Context API, uma ferramenta poderosa no ecossistema React.

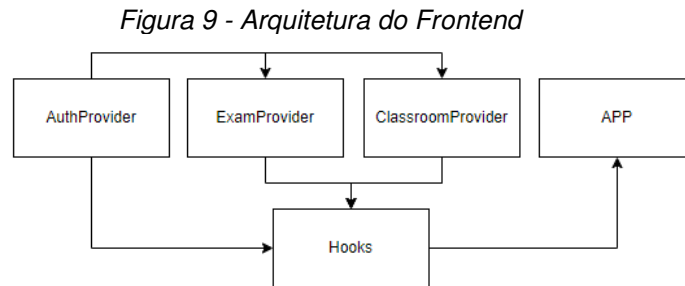
A Context API do React foi adotada como um pilar fundamental na arquitetura do *frontend* devido à sua capacidade de gerenciar o estado global da aplicação de maneira eficiente. No contexto educacional, onde diversas informações, como dados do usuário, configurações da aplicação e informações específicas do contexto de aprendizado, precisam ser acessadas em vários componentes, a Context API oferece uma solução elegante para o gerenciamento compartilhado de estado.

A escolha da Context API alinha-se com a busca por uma arquitetura que promova a coesão e a reutilização de componentes, contribuindo para a manutenibilidade do código e a escalabilidade da aplicação. A capacidade de evitar a chamada repetitiva de propriedades entre componentes e proporcionar um acesso simplificado ao estado global faz da Context API uma escolha estratégica para otimizar o desenvolvimento e aprimorar a eficiência da aplicação QuestõesPro.

Na prática, a Context API foi empregada para gerenciar dados cruciais, como o progresso do usuário, configurações personalizadas e informações sobre as questões. Essa abordagem permitiu que diferentes partes da aplicação acessassem e atualizassem esses dados de maneira centralizada, simplificando a lógica do componente e melhorando a consistência das informações exibidas.

Em resumo, a escolha da Context API como parte integrante da arquitetura do *frontend* no projeto QuestõesPro foi motivada pela necessidade de eficiência no gerenciamento de estado global, promovendo coesão, reutilização de componentes e uma experiência do usuário mais aprimorada. Dessa forma, segundo a Figura 9 e a Tabela 3, estabelecemos a arquitetura do *frontend* do nosso sistema utilizando a

Context API, na qual os estados são organizados em três provedores principais: AuthProvider, ExamProvider e ClassroomProvider. Essa abordagem estruturada permite uma gestão eficiente dos estados relacionados à autenticação, avaliações e salas de aula, utilizando a Context API para facilitar o compartilhamento e acesso a essas informações entre os diversos componentes da aplicação.



Fonte: Elaborado pelo autor, 2023

Tabela 3 - Tabela de Contextos da aplicação

Contextos	Responsabilidade
AuthProvider	Tem a responsabilidade de gerenciar os dados do Usuário, perante a aplicação. Ele possui métodos que conectam com a API, acessando informações do usuário, como o <i>token</i> . Além disso é responsável por acessar o estado mais elevado da aplicação com relação ao usuário, controlando se o usuário está ou não logado.
ExamProvider	Tem a responsabilidade de gerenciar a entidade de Exames, possuindo métodos que conectam com API, realizando chamadas para realizar atualização, criação e leitura de exames.
ClassroomProvider	Tem a responsabilidade de gerenciar a entidade de Turmas, possuindo métodos que conectam com API, realizando chamadas para realizar atualização, criação e leitura de turmas.

Fonte: Elaborado pelo autor, 2023

4. RESULTADOS

Este capítulo apresenta os resultados do desenvolvimento do QuestõesPro *mobile*, com o propósito de ilustrar suas principais funcionalidades e fluxos. Para execução do fluxo, são empregados dados fictícios e imagens das telas da aplicação, apresentadas no simulador do Android Studio.

4.1. LOGIN E CRIAÇÃO DE CONTA

Ao iniciar o aplicativo QuestõesPro *mobile* no dispositivo móvel, o usuário é conduzido a uma tela (Figura 10) que apresenta dois fluxos alternativos que o usuário pode seguir (Tabela 4).

Tabela 4 - Tabela de Fluxo de usuário

Fluxos	Responsabilidade
Login	O fluxo tem como finalidade encaminhar o usuário ao formulário destinado à autenticação na aplicação.
Cadastrar	O fluxo tem como propósito encaminhar o usuário ao formulário destinado ao cadastro na aplicação

Fonte: Elaborado pelo autor, 2023

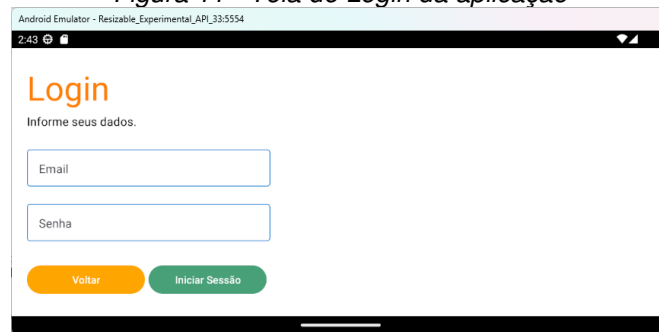
Figura 10 - Menu principal da aplicação



Fonte: Elaborado pelo autor, 2023

No caso de o usuário já possuir uma conta na aplicação, é suficiente clicar no botão “Login” para ser direcionado à tela (Figura 11) que apresenta o formulário, onde os campos de e-mail e senha devem ser preenchidos.

Figura 11 - Tela de Login da aplicação



Fonte: Elaborado pelo autor, 2023

Se o usuário ainda não possuir uma conta, ele deverá clicar no botão “Cadastrar”, na tela inicial da aplicação. Ao interagir com ele, a aplicação apresentará um modal (Figura 12), onde o usuário deverá escolher que tipo de usuário corresponde as suas necessidades (Tabela 5).

Figura 12 - Menu de decisão de fluxo de cadastro



Fonte: Elaborado pelo autor, 2023

Tabela 5 - Tipos de usuário

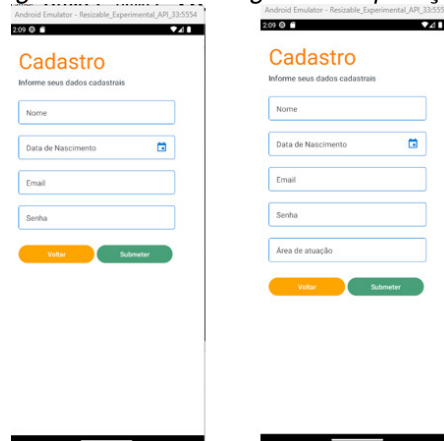
Tipo de usuário	Permissões
Aluno	Entrar em uma turma, visualização de turmas e agendamentos, e responder um exame.
Professor	Gerenciamento de turmas, agendamentos, questões e exames.

Fonte: Elaborado pelo autor, 2023

Ao selecionar o botão “Aluno”, o usuário será redirecionado a um formulário no qual deve fornecer seu nome completo, data de nascimento, e-mail e uma senha para acessar a aplicação. Para usuários do nível professor, além dos campos

anteriormente mencionados, há a inclusão do campo adicional “Área de Atuação”. Ambos os formulários podem ser observados na Figura 13.

Figura 13 - Tela de registro da aplicação

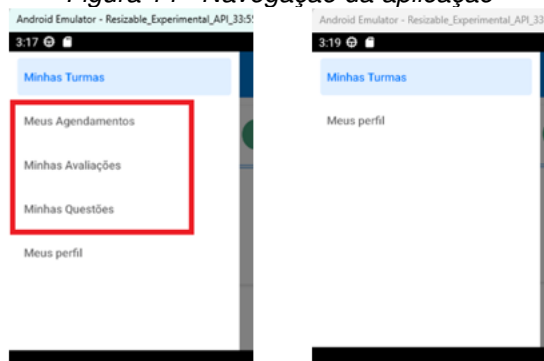


Fonte: Elaborado pelo autor, 2023

4.2. NAVEGAÇÃO DA APLICAÇÃO

Conforme mencionado anteriormente, a aplicação possui dois níveis de usuário, cada um com suas permissões específicas, impactando diretamente na navegação. Alguns recursos da aplicação estão restritos aos usuários professores, enquanto outros estão disponíveis apenas para os usuários alunos. Observe nas imagens a seguir, que o usuário professor possui acesso as páginas de agendamentos, avaliações e questões, diferentemente do usuário aluno (Figura 14).

Figura 14 - Navegação da aplicação



Fonte: Elaborado pelo autor, 2023

Embora o usuário do tipo aluno tenha acesso à página “Minhas Turmas”, ainda assim, ele possui restrições específicas. Da mesma forma, o usuário do tipo professor também enfrenta suas próprias restrições, como o acesso à página para

responder a um determinado exame. Esses aspectos serão discutidos nas próximas seções.

4.3. PROCESSO DE CRIAÇÃO, VISUALIZAÇÃO, EXCLUSÃO, ENTRADA E SAÍDA DE UMA TURMA.

A tela de “Minhas turmas”, apresenta uma semelhança entre os dois níveis de usuário da aplicação. Ambas vão apresentar *cards* (Figura 15) que representam as turmas ao qual o usuário pertence.

Figura 15 - Card representativo de uma turma



Fonte: Elaborado pelo autor, 2023

Cada *card*, possui informações (Tabela 6) atuais sobre a turma, oferecendo uma experiência prática, onde ele pode escolher a opção que mais se adequa às suas necessidades no momento.

Tabela 6 - Estrutura do Card de turmas

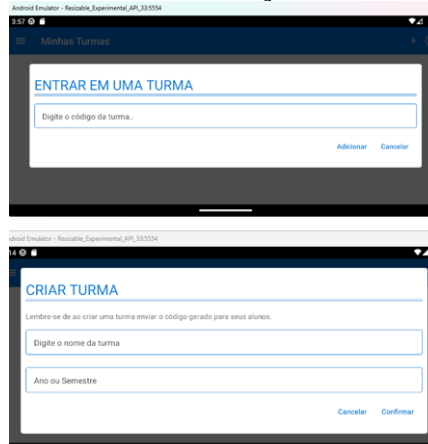
Estrutura do <i>Card</i>	Responsabilidade
Título	Informar o nome da turma ao usuário logado.
Professor	Informa ao usuário qual professor é responsável por gerenciar a turma.
Semestre	Informa ao usuário em que semestre a turma pertence.
Criada em	Informa ao usuário em que período do ano a turma foi criada.

Código da turma	Código usado para entrar na turma (possui funcionalidade de cópia para ser passado para demais usuários).
Número de Alunos	Informa o número de alunos pertencentes a turma.
Remover Turma (Apenas professor)	Permite ao usuário do tipo professor excluir a turma, removendo assim todos os dados relacionados a respectiva turma.
Sair (Apenas aluno)	Permite ao usuário do tipo aluno sair da turma.
Ver turma (Fluxo diferente para ambos os tipos de usuário)	Se o usuário for do tipo aluno, será redirecionado para a página que apresentará os seus agendamentos. Já para professor irá ser redirecionado para uma página no qual ele poderá gerenciar aquela turma.

Fonte: Elaborado pelo autor, 2023

Na página “Minhas Turmas”, o cabeçalho apresenta o botão “+”, embora o ícone seja o mesmo para ambos os tipos de usuários na aplicação, ele leva a fluxos distintos. No caso do aluno, um modal (Figura 16) será exibido, onde o campo “Código da Turma” deve ser preenchido, sendo fornecido pelo professor responsável. Já para o professor, os campos “Nome da Turma” e “Ano ou Semestre” devem ser preenchidos. (Figura 16).

Figura 16 - Formulário de criação e entrada de turma



Fonte: Elaborado pelo autor, 2023

4.4. GERENCIAMENTO DE TURMAS

A página de gerenciamento de turmas (Figura 17) só é permitida ao usuário de nível professor. A aplicação oferece recurso sobre informações da turma, como apresentados na Tabela 7.

Figura 17 - Página de gerenciamento de turma



Fonte: Elaborado pelo autor, 2023

Tabela 7 - Estrutura header da página de gerenciamento de turma

Estrutura do Header	Responsabilidade
Título	Informar o nome da turma ao usuário logado.
Semestre	Informa ao usuário em que semestre a turma pertence.
Criada em	Informa ao usuário em que período do ano a turma foi criada.
Código da turma	Código usado para entrar na turma (possui funcionalidade de cópia para ser passado para demais usuários).
Alunos	Informa o número de alunos pertencentes a turma.

Fonte: Elaborado pelo autor, 2023

Adicionalmente, a página apresenta dois botões, Alunos e Agendamentos (Figura 18) que alternam entre tabelas. Ao acionar o botão “Alunos”, uma tabela é exibida no aplicativo contendo a lista de alunos pertencentes àquela turma, acompanhada de suas respectivas informações, tais como: id, nome e média.

Figura 18 - Controle de agendamentos a alunos de uma turma

Alunos		Agendamentos
Id	Nome	Média
0	Rodrigo Cunha França	3
1	Rodrigo Cunha França	3

Fonte: Elaborado pelo autor, 2023

Interagindo com o botão “Agendamentos”, a página irá apresentar uma tabela apresentando a relação de agendamentos para aquela determinada turma, com seus respectivos: nome, prova, início do prazo, fim do prazo e seu status (aberto, fechado ou agendado). A tabela de agendamentos, conta ainda com uma componente de busca, onde o usuário poderá realizar a busca do agendamento por seu nome, melhorando a usabilidade para o professor. Além disso, o professor ainda possui o botão de ‘Detalhes’ para cada agendamento (Figura 19).

Figura 19 - Busca de agendamentos

Agendamento	Prova	Início do prazo	Fim do prazo	Status
asdasdsad	Ava1	06/11/2023 05:10	06/11/2023 06:10	Fechada
asdasd	Ava1	06/11/2023 05:10	06/11/2023 06:10	Fechada

Fonte: Elaborado pelo autor, 2023

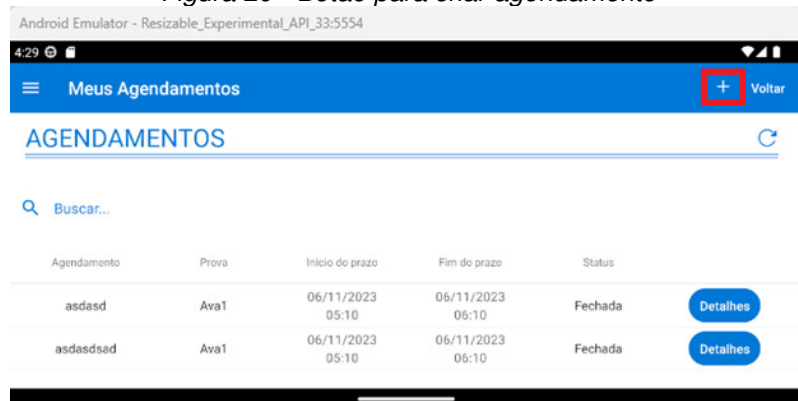
Ao acionar o botão “Detalhes”, um modal é ativado, exibindo algumas das informações previamente mencionadas, além do acréscimo do campo de duração do agendamento e um botão para sua exclusão.

4.5. AGENDAMENTOS

4.5.1. Visualização de agendamentos para usuário professor

A página de “Meus agendamentos” só é permitida ao usuário de nível professor. Ela apresentará uma tabela semelhante à apresentada na seção anterior, possuindo funcionalidades de exclusão, listagem e busca. Isso é resultado da implementação do design atômico, gerando reaproveitamento das componentes e de suas funcionalidades. A mesma tabela de agendamentos não é restrita a uma turma.

Figura 20 - Botão para criar agendamento



Fonte: Elaborado pelo autor, 2023

Além disso a página conta em seu *header*, com o botão “+” (Figura 20), que ao ser acionado redirecionará o professor à tela de criação de agendamento.

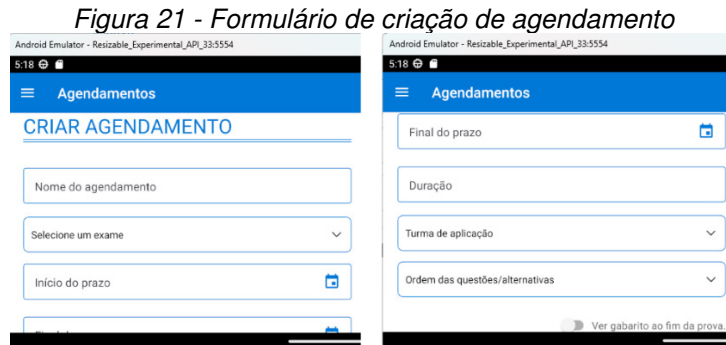
O usuário professor ainda conta com o recurso de criação de um agendamento. Nesta página será encontrado um formulário (Figura 21) cujo todos os campos (Tabela 8) são de preenchimento obrigatório.

Tabela 8 - Campos do formulário de criação de agendamento

Campo do formulário	Descrição
Nome do Agendamento	Deverá ser informado o título do agendamento a ser criado.
Exame	Deverá ser selecionado um exame já configurado pelo usuário.
Início do prazo	Deverá ser informado o prazo de início para aplicação do agendamento.
Fim do prazo	Deverá ser informado o prazo de fim para aplicação do agendamento.
Duração	Deverá ser informado o prazo de duração da aplicação do exame em minutos.
Turma de aplicação	Deverá ser informado a turma ao qual o exame deverá ser aplicado.

Ordem das questões/alternativas	Deverá ser selecionado, a ordem em que as questões e alternativas deverão ser apresentadas aos alunos.
---------------------------------	--

Fonte: Elaborado pelo autor, 2023



Fonte: Elaborado pelo autor, 2023

Ao finalizar o preenchimento, o usuário deve confirmar acionando o botão “Submeter”, onde ele será redirecionado à página de “Meus Agendamentos”, em que poderá ser visualizado o agendamento recém-criado.

4.5.2. Visualização de agendamentos para usuário aluno

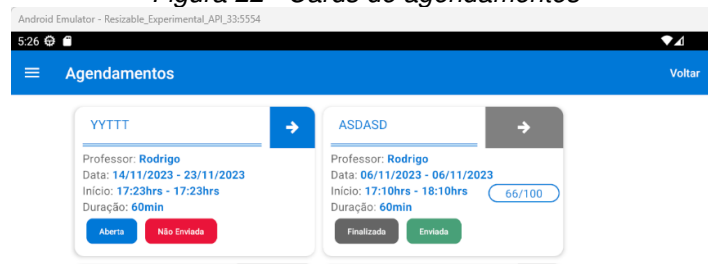
A tela de agendamentos para o aluno apresenta *cards* (Figura 22) representando o agendamento realizado pelo professor, para aquela determinada turma. Cada *card* possui informações para o agendamento, conforme a Tabela 9.

Tabela 9 - Campos do Card de agendamento

Campo do Card	Descrição
Título	Nome do respectivo agendamento.
Professor	Nome do professor que criou o agendamento.
Data	Data em que o agendamento deve iniciar e finalizar.
Início	Hora de início e fim do agendamento.
Duração	Duração do agendamento.
Tags coloridas	Representam o atual estado do agendamento para o aluno, podendo variar entre aberta, não enviada, agendada e enviada.

Fonte: Elaborado pelo autor, 2023

Figura 22 - Cards de agendamentos



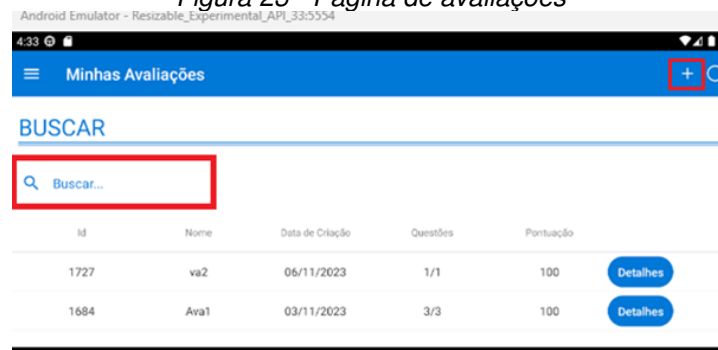
Fonte: Elaborado pelo autor, 2023

4.6. AVALIAÇÕES

4.6.1. Avaliações para o usuário professor

O usuário de professor, conta com a página que lista todos os tipos de avaliações já criadas por ele. A tabela de avaliações, em que o usuário pode conferir suas respectivas informações (Tabela 10), conta com um componente de busca por nome da avaliação para melhor experiência de usuário com o software (Figura 23).

Figura 23 - Página de avaliações



Fonte: Elaborado pelo autor, 2023

Tabela 10 - Colunas da tabela de avaliações

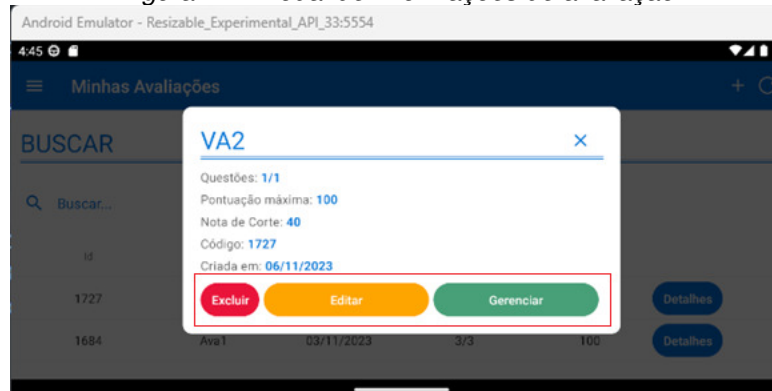
Campos da tabela	Descrição
Id	Número de identificação da avaliação.
Nome da avaliação	Título dado a avaliação pelo professor.
Data de criação	Período em que foi criado a avaliação
Questões	Informa ao usuário o número de questões que foram adicionadas à avaliação e o número máximo de questões permitidas ao mesmo.
Pontuação	Pontuação máxima que o aluno pode obter para aquela avaliação.

Detalhes	Ao ser acionado, abre um modal na tela, contendo as informações da avaliação, podendo ser extensível a outras informações.
----------	--

Fonte: Elaborado pelo autor, 2023

O botão “Detalhes”, quando ativado, abre um modal (Figura 24) que oferece uma visão mais detalhada da atividade, contribuindo para a escalabilidade do componente. Essa abordagem permite uma fácil adição de informações adicionais sobre a avaliação, uma vez que a restrição de espaço na tela é significativamente reduzida em comparação a uma linha de tabela.

Figura 24 - Modal de informações de avaliação



Fonte: Elaborado pelo autor, 2023

O modal ainda disponibiliza os recursos (Tabela 11), possibilitando ao usuário explorar todas as opções de fluxo diretamente na palma da mão. Essa adaptação às suas necessidades contribui para aprimorar a experiência durante a utilização da aplicação.

Tabela 11 – Funcionalidade dos botões do modal de avaliação

Funcionalidade	Descrição
Excluir	Permite o professor a remoção da avaliação do sistema.
Editar	Permite o professor editar as configurações de uma avaliação.
Gerenciar	Redireciona o professor para a página de gerenciamento de questões da avaliação.

Fonte: Elaborado pelo autor, 2023

Na página “Minhas Avaliações”, o professor tem a capacidade de criar uma avaliação por meio do botão “+” localizado no cabeçalho da página (Figura 23). Ao ser acionado, um formulário (Figura 25) é apresentado, exigindo que suas entradas (Tabela 12) sejam preenchidas obrigatoriamente.

Figura 25 - Formulário de criação de avaliação

Fonte: Elaborado pelo autor, 2023

Tabela 12 - Campos do formulário de criação de configuração de avaliação

Entrada	Descrição
Nome	Permite o professor dar um título a uma avaliação.
Nota de corte	Permite ao professor definir a nota mínima para passar na avaliação.
Nota máxima	Permite ao professor definir a nota máxima de uma avaliação.
Número de questões	Permite ao professor definir o número de questões presentes na avaliação.

Fonte: Elaborado pelo autor, 2023

O mesmo formulário (Figura 26) é reutilizado na funcionalidade do botão “Editar”, conforme mencionado na tabela. É relevante observar que, em ambos os casos, o formato, tamanho das fontes, cores e botões são idênticos, atribuído ao princípio do design atômico. Além de favorecer a reutilização de código, esse método contribui para uma interface do usuário mais coesa, evitando confusões relacionadas à estilização.

Figura 26- Formulário de edição de avaliação

Fonte: Elaborado pelo autor, 2023

Na opção de “Gerenciar” citado na tabela, o professor é redirecionado a página (Figura 27) onde ele pode conferir informações sobre a avaliação, acompanhar informações como o código do exame, nota de corte, pontuação máxima, limite de questões de exame, questões cadastradas, o que torna a página mais adaptada as necessidades do professor.

Figura 27 - Página de gerenciamento de exame

Fonte: Elaborado pelo autor, 2023

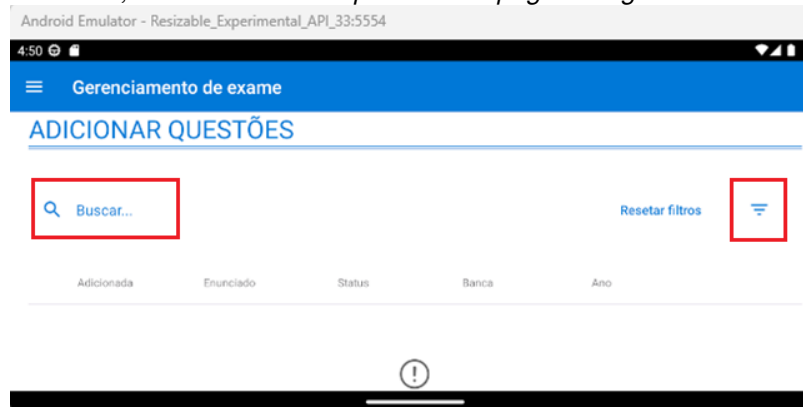
Adicionalmente, a página contém uma seção na qual o professor pode monitorar as questões que foram previamente adicionadas ao exame (Figura 28), possibilitando a remoção de questões, a verificação de suas respostas, e a opção de marcá-las como favoritas.

Figura 28 - Gerenciamento da questão no exame

Fonte: Elaborado pelo autor, 2023

Outra funcionalidade presente na página de gestão de avaliações é a inclusão de questões na avaliação. Uma tabela está disponível, possibilitando ao professor realizar buscas (Figura 29) pelo enunciado da questão e conferir informações conforme Tabela 13. Essa abordagem visa aprimorar a experiência do usuário com o produto, reduzindo o tempo necessário para localizar e adicionar uma questão específica.

Figura 29 - Tabela, busca e filtros de questões na página de gerenciamento de exame



Fonte: Elaborado pelo autor, 2023

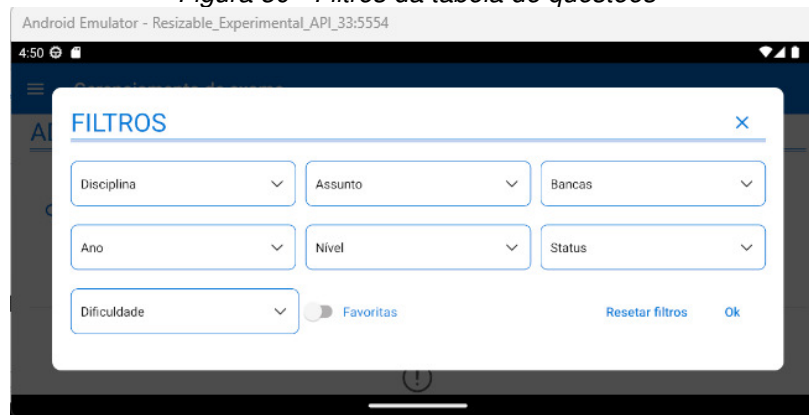
Tabela 13 - Colunas da tabela de questões

Nome da coluna	Descrição
Adicionada	Preenchida, caso a questão já esteja presente na avaliação atual.
Enunciado	Texto presente no enunciado da questão
Status	Se a questão está disponível para todos os usuários da plataforma, ou só para o usuário atual.
Banca	Bancas ao qual a questão pertence.
Ano	Ano em que o professor adicionou a questão.

Fonte: Elaborado pelo autor, 2023

A tabela possui uma componente de filtros (Figura 30), para busca de questões, onde o professor é capaz de deixar suas buscas ainda mais dinâmicas, e precisas. Os campos selecionados são: disciplina, assunto, bancas, ano, nível, *status*, dificuldade e favoritos (Tabela 14).

Figura 30 - Filtros da tabela de questões



Fonte: Elaborado pelo autor, 2023

Tabela 14 - Filtros de questões

Nome da entrada	Descrição
Disciplina	Nome da disciplina ao qual a questão pertence.
Assunto	Assunto da questão.
Bancas	Bancas ao qual a questão pertence.
Ano	Ano em que o professor adicionou a questão.
Status	Se a questão está disponível para todos os usuários da plataforma, ou só para o usuário atual.
Nível	Nível correspondente a questão.
Dificuldade	Dificuldade que corresponde a questão.
Favoritas	Se a questão buscada, é ou não favoritada daquele usuário logado.

Fonte: Elaborado pelo autor, 2023

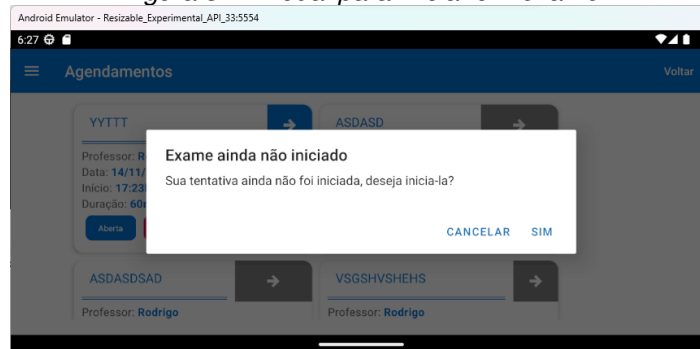
Com isso, apresentamos uma página que concede ao professor total controle sobre a gestão de questões e avaliações em uma única interface, proporcionando uma experiência ágil.

4.6.2. Avaliações para o usuário Aluno

A página do exame pode ser acessada por meio de seu *card* sendo o acesso permitido apenas se o prazo da avaliação estiver dentro do limite estabelecido pelo professor, se houver tempo restante para a duração da prova ou se o aluno ainda não tiver respondido à avaliação. Todos esses estados podem ser verificados nas etiquetas presentes nos *cards*.

Ao escolher iniciar o exame, o aluno será questionado se deseja dar início ao teste (Figura 31). Esse modal proporciona ao usuário uma sensação de maior segurança e controle em relação às ações na aplicação.

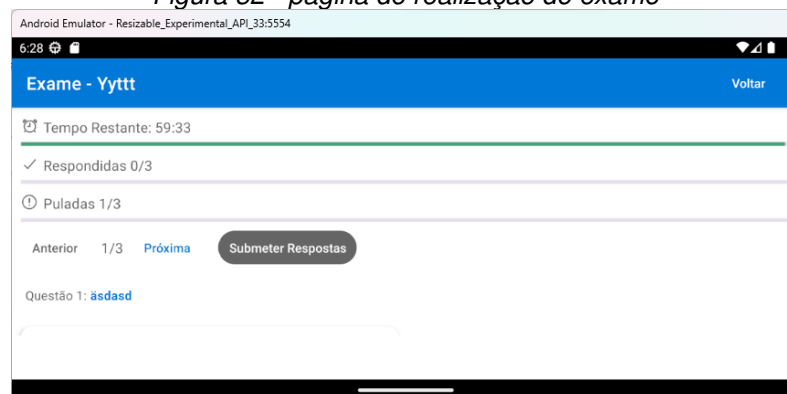
Figura 31 - Modal para iniciar um exame



Fonte: Elaborado pelo autor, 2023

Ao acessar a página do exame (Figura 32), o aluno poderá monitorar o tempo restante da prova, as questões que foram respondidas e aquelas que foram puladas durante a sessão. É permitido ao aluno transitar entre as questões, proporcionando agilidade, especialmente se ele tiver afinidade com uma questão específica.

Figura 32 - página de realização do exame

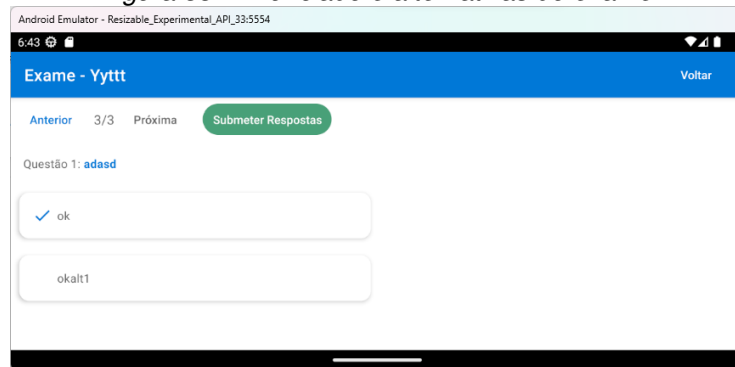


Fonte: Elaborado pelo autor, 2023

O aluno tem a possibilidade de acompanhar o enunciado da questão bem como suas alternativas, incluindo aquelas que já foram selecionadas. Além disso, ele pode escolher alternativas e, caso não tenha certeza, é permitido transitar entre as páginas sem perder as respostas previamente registradas. O aluno só poderá concluir

o exercício se todas as questões foram respondidas, situação em que o botão “Submeter” ficará disponível (Figura 33).

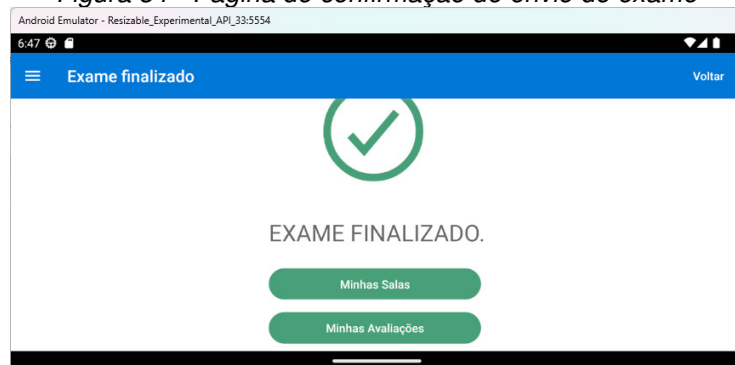
Figura 33 - Enunciado e alternativas do exame



Fonte: Elaborado pelo autor, 2023

Ao pressionar o botão “Submeter”, o usuário será redirecionado para a página de sucesso (Figura 34), que informará que a atividade foi concluída. Essa abordagem visa evitar dúvidas por parte do aluno quanto ao envio de seu exame.

Figura 34 - Página de confirmação de envio do exame



Fonte: Elaborado pelo autor, 2023

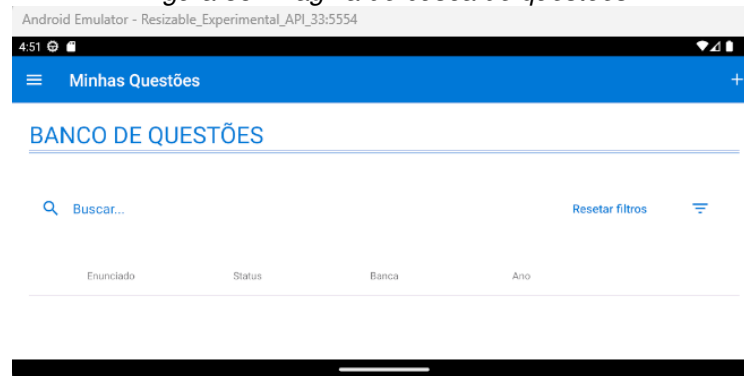
Com o intuito de aprimorar a aplicação de acordo com as necessidades do cliente, a página de sucesso ainda disponibiliza os botões “Minhas Salas”, para o usuário retornar à tela de suas turmas, e “Minhas Avaliações”, para retornar às avaliações da mesma disciplina.

4.7. QUESTÕES

A página de questões (Figura 35) pode ser acessada através da navegação se o usuário for do tipo professor. Esta página apresentará uma tabela com

semelhanças ao que encontramos na página de gerenciamento de exame. A diferença é que ela não referenciará um exame específico.

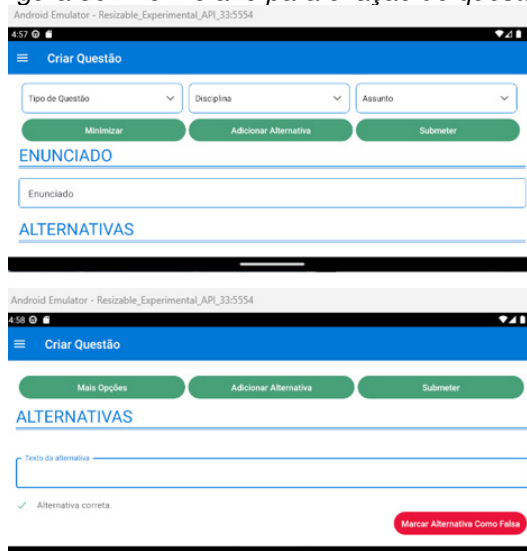
Figura 35 - Página de busca de questões



Fonte: Elaborado pelo autor, 2023

A tabela desta página apresenta o mesmo sistema de busca e filtros para uma melhor experiência do usuário, como apresentado na seção 4.3.1. Esta página ainda permite a criação de tabelas acionando o botão “+”, presente no *header* da página. Ao ser acionado, o usuário será redirecionado a um formulário (Figura 36) para preencher entradas de texto, e suas demais funcionalidades (Tabela 15).

Figura 36 - Formulário para criação de questão



Fonte: Elaborado pelo autor, 2023

Tabela 15 - Campos do formulário de criação de questão

Nome da entrada	Descrição
Tipo de Questão	Deve ser selecionada e pode ser alternada entre Verdadeiro e Falso ou Múltipla escolha.
Assunto	Deve ser selecionado o assunto ao qual a questão pertence.
Enunciado da questão	Texto enunciado da questão que deverá ser apresentado no exame.
Texto da alternativa.	Texto da alternativa que deverá ser apresentada no exame.
Marcar alternativa como verdadeira	Define a alternativa como verdadeira.
Marcar alternativa como falsa	Define a alternativa como falsa.
Remover alternativa	Remove a alternativa da questão.
Adicionar alternativa	Adiciona uma alternativa no exame, sendo no máximo 10 alternativas por exame.

Fonte: Elaborado pelo autor, 2023

Com todas as informações preenchidas o usuário deve acionar o botão “Submeter”, para que a questão seja criada no sistema. Logo ele poderá confirmar que a questão criada estará presente na tabela de questões presente na página anterior.

É possível visualizar os detalhes da questão, como favorita lá, acessar seu enunciado, suas alternativas, quais são as respostas verdadeiras e faltas, basta o usuário acionar o botão “Detalhes”, presente na linha que representa uma questão, ele apresentará o modal da Figura 37.

Figura 37 - Modal de informações de questões



Fonte: Elaborado pelo autor, 2023

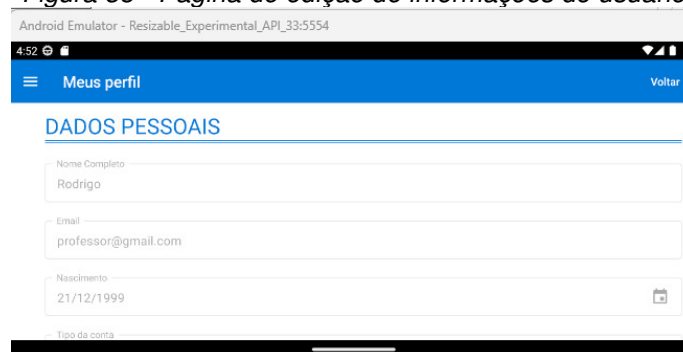
O modal permite que o usuário edite a questão por meio do botão “Editar Questão”, o professor será redirecionado ao formulário com as mesmas características presentes na Tabela 15.

Com isso o professor tem acesso a todas as funcionalidades de criação, visualização, e edição de questões. Dando a ele total controle e confiabilidade, e agilidade utilizando o sistema por meio de uma página.

4.8. INFORMAÇÕES DO USUÁRIO

A página de informações do usuário (Figura 38) estará disponível para ambos os níveis de usuário da aplicação. O usuário poderá visualizar todas as suas informações (Tabela 16) utilizadas durante o cadastro do sistema.

Figura 38 - Página de edição de informações do usuário



Fonte: Elaborado pelo autor, 2023

Tabela 16 - Campos do formulário de edição de informações do usuário

Nome da entrada	Descrição
Nome Completo	Deverá ser informado o nome completo.
E-mail	Deverá ser informado o e-mail utilizado para login na plataforma.

Nascimento	Deverá ser informado a data de nascimento do usuário.
Tipo de conta	Nível de usuário da aplicação.
Marcar alternativa como verdadeira	Define a alternativa como verdadeira.
Marcar alternativa como falsa	Define a alternativa como falsa.
Remover alternativa	Remove a alternativa da questão.
Adicionar alternativa	Adiciona uma alternativa no exame, sendo no máximo 10 alternativas por exame.

Fonte: Elaborado pelo autor, 2023

Nesta página também será possível a edição de algumas das informações do usuário, basta acionar o botão “editar”, logo as entradas de “Nome Completo” e “Nascimento” como segue na Figura 39. Isso permite que o usuário corrija informações que ele preencheu erroneamente durante o cadastro da aplicação.

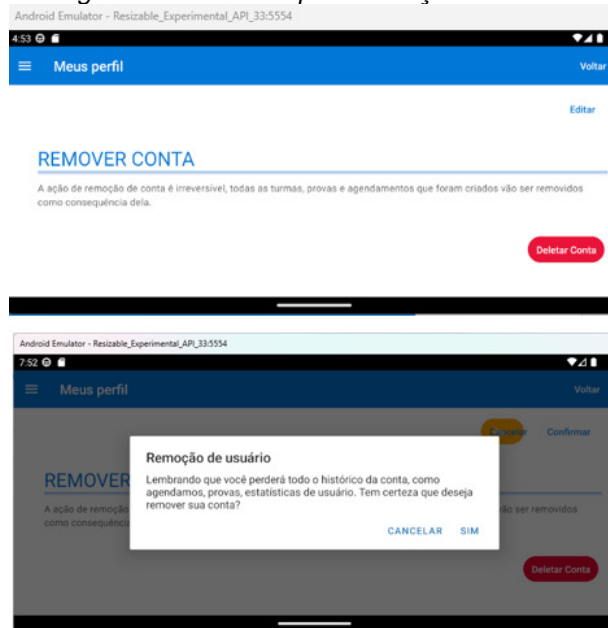
Figura 39 - Formulário de edição de informações do usuário

The image shows a screenshot of an Android emulator displaying a user profile editing screen. The screen has a blue header with a menu icon on the left, the text 'Meus perfil' in the center, and a 'Voltar' button on the right. Below the header, there are four input fields: 'Nome Completo' containing 'Rodrigo', 'Email' containing 'professor@gmail.com', 'Nascimento' containing '21/12/1999' with a calendar icon to its right, and 'Tipo da conta' containing 'Professor'. The emulator's status bar at the top shows the time as 7:50 and various system icons.

Fonte: Elaborado pelo autor, 2023

O recurso de informações, possui uma sessão para remoção de conta, isso permite ao usuário que ele todos os seus dados sejam removidos do sistema. Para isso o usuário deve acionar o botão “Deletar Conta” (Figura 40), que acionará o modal de permissão, garantindo a ele não remova seus dados de maneira acidental precisando de uma confirmação para o processo acontecer.

Figura 40 - Recurso para remoção de conta



Fonte: Elaborado pelo autor, 2023

5. CONSIDERAÇÕES FINAIS

A migração do *software* QuestõesPro *web* para o ambiente *mobile* constitui um marco significativo no panorama educacional, trazendo consigo vantagens notáveis para os professores no tocante ao gerenciamento eficaz de agendamentos, criação de provas, elaboração de questões e leitura de informações cruciais por meio de dispositivos móveis. Esta transição representa um passo essencial na adaptação da tecnologia educacional às exigências contemporâneas, proporcionando maior flexibilidade e acessibilidade para os profissionais da educação. Contudo, ao término deste estudo, é imperativo abordar pendências críticas, destacando a necessidade premente da implementação dos tipos de questões verdadeira ou falsa e a realização de testes unitários, integração, sistema e aceitação abrangentes para assegurar a estabilidade e confiabilidade da aplicação.

O comprometimento contínuo com o desenvolvimento e teste emerge como elemento essencial para otimizar plenamente o *software* no ambiente *mobile*, maximizando seu potencial de contribuição ao cenário educacional. Embora a migração tenha proporcionado benefícios inegáveis, a efetiva resolução das pendências identificadas é fundamental para consolidar o sucesso desta transição. A implementação bem-sucedida dessas funcionalidades pendentes não apenas atenderá às expectativas dos usuários, mas também reforçará a aplicabilidade do *software* no contexto educacional, contribuindo para a modernização contínua dos processos educacionais.

Os desafios encontrados durante o desenvolvimento da presente pesquisa dizem respeito ao limite de tempo, o desenvolvimento do cronômetro do exame do aluno devido as limitações com funcionalidade de *timers* que envolvem React, Node e Javascript, e a autenticação do usuário limitada por tempo.

Paralelamente, recomenda-se a exploração de melhorias adicionais, incluindo a incorporação de métodos avançados de análise de respostas, a diversificação de tipos de questões e a integração de tecnologias emergentes que possam enriquecer ainda mais a experiência educacional, a exemplo da criação de questões por meio da IA generativa. Além disso, a realização de testes em diferentes ambientes educacionais e a coleta de *feedback* dos usuários são práticas cruciais para avaliar a adaptabilidade do *software* e garantir que ele atenda plenamente às necessidades dos educadores em diversos contextos. Dessa forma, os trabalhos

futuros devem concentrar-se na conclusão das pendências identificadas e na exploração de melhorias adicionais para garantir a eficácia contínua da migração do QuestõesPro *web* para o ambiente *mobile*.

REFERÊNCIAS

- ALMENARA, Igor. O que é uma versão LTS. Disponível em: <https://canaltech.com.br/apps/o-que-e-versao-lts-180894/>. Acesso em: 09 de novembro de 2023.
- ANJOS, Luiz Felipe Rosa dos. Evolução do Javascript em aplicações multiplataforma. 2017.
- BARROS, Claudia Cristiane Andrade et al. Precarização do trabalho docente: reflexões em tempos de pandemia e pós pandemia. *Ensino em Perspectivas*, v. 2, n. 2, p. 1-23, 2021.
- BATISTA, Silvia Cristina Freitas; BARCELOS, Gilmara Teixeira. Análise do uso do celular no contexto educacional. *Renote*, v. 11, n. 1, 2013.
- BORTOLAZZO, Sandro Faccin. Aprendendo com Artefatos de bolso: entre Smartphones e Educação. *Cadernos de Comunicação*, v. 24, n. 1, 2020.
- CRUZ, Vitor Silva; PRETUCELLI, Erick Eduardo. *Tecnologias Web para o Desenvolvimento Mobile Nativo*. 2017.
- DA FONSECA, Ana Graciela Mendes Fernandes. Aprendizagem, mobilidade e convergência: mobile learning com celulares e smartphones. *Revista mídia e cotidiano*, v. 2, n. 2, p. 265-283, 2013.
- DA SILVA MELO, Rafaela; CARVALHO, Marie Jane Soares. Aplicativos educacionais livres para mobile learning. In: *Anais do Encontro Virtual de Documentação em Software Livre e Congresso Internacional de Linguagem e Tecnologia Online*. 2014.
- DE MENDONÇA, Vinicius Rafael Lobo; BITTAR, Thiago Jabur; DE SOUZA DIAS, Márcio. Um estudo dos Sistemas Operacionais Android e iOS para o desenvolvimento de aplicativos. 2011.
- FERREIRA, Haline Kelly; ZUCHI, Jederson Donizete. Análise comparativa entre *frameworks* frontend baseados em javascript para aplicações web. *Revista Interface Tecnológica*, v. 15, n. 2, p. 111-116, 2018.
- FROST, Brad. Atomic Design Methodology. Disponível em: <https://atomicdesign.bradfrost.com/>. Acesso em: 09 de novembro de 2023.
- Google. Material Design. Disponível em: <https://m3.material.io>. Acesso em: 05 de outubro de 2023.
- KRONE, Caroline et al. Desenvolvimento e teste de componentes de interface acessíveis para um Design System. 2019.
- LOPES, Léo. Apple lança iOS 17; veja iPhones compatíveis, como atualizar e quais as novidades. Disponível em: <https://www.cnnbrasil.com.br/tecnologia/ios-17-como-atualizar-baixar-novidades-apple>. Acesso em: 05 de outubro de 2023.

PAULINO, Leonardo Soares. NODEJS. REFAQI-REVISTA DE GESTÃO EDUCAÇÃO EE TECNOLOGIA, v. 4, n. 2, p. 3-3, 2018.

PEREZ, Karine Vanessa. Jornada de trabalho real e invisível: uma análise sobre o cotidiano de docentes em instituições de ensino superior privadas. In: Anais do Congresso Brasileiro de Estudos Organizacionais. 2016.

QUEVEDO, Brener Nosse. Atomic design no contexto de desenvolvimento: um relato de experiência. 2023.

React Native Paper. Disponível em: <https://callstack.github.io/react-native-paper>. Acesso em: 05 de outubro de 2023.

React Native. Setting up the development environment. Disponível em: <https://reactnative.dev/docs/environment-setup>. Acesso em: 09 de novembro de 2023.

REINALDO, Francisco et al. Uso de smartphones na educação: Avaliação por grupos focais. CIAIQ2016, v. 1, 2016.

RODRIGUEZ, Iara. State of Mobile 2023: estudo mostra tendências do mercado de apps e comportamento dos usuários. Disponível em: <https://g4educacao.com/portal/state-of-mobile-2023>. Acesso em: 05 de outubro de 2023.

SANTIAGO, Felipe Issa. Atomic design aplicado no desenvolvimento de produtos. 2022.


SATO, Silvio Koiti. A estética publicitária da inovação: smartphones e tablets. Pensamento & Realidade, v. 26, n. 3, 2011.

SILVA, Leon Tenório da. Desenvolvimento de um sistema para colaboração de universitários com os estudos de pré-universitários utilizando Flutter e API RESTful em Typescript. 2022.

SILVA, Túlio de Souza. Minha Prova: automatizando o processo avaliativo nas escolas. 2019. Trabalho de Conclusão de Curso. Brasil.

SILVA, Werliton Carlos Sousa da. Aplicações móveis nativas com react native e firebase: um estudo de caso. 2018.

SOUZA, Andre Luiz et al. Tecnologia ou metodologia: aplicativos móveis na sala de aula. In: Anais do Encontro Virtual de Documentação em software livre e Congresso Internacional de linguagem e tecnologia online. 2016.

	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA
	Campus Campina Grande
	R. Tranquílino Coelho Lemos, 671, Dinâmérica, CEP 58432-300, Campina Grande (PB)
	CNPJ: 10.783.898/0003-37 - Telefone: (83) 2102.6200

Documento Digitalizado Restrito

Documento final TCC

Assunto:	Documento final TCC
Assinado por:	Rodrigo Cunha
Tipo do Documento:	Dissertação
Situação:	Finalizado
Nível de Acesso:	Restrito
Hipótese Legal:	Informação Pessoal (Art. 31 da Lei no 12.527/2011)
Tipo da Conferência:	Cópia Simples

Documento assinado eletronicamente por:

- **Rodrigo Cunha França, ALUNO (201911210025) DE TECNOLOGIA EM TELEMÁTICA - CAMPINA GRANDE**, em 19/01/2024 12:30:05.

Este documento foi armazenado no SUAP em 19/01/2024. Para comprovar sua integridade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifpb.edu.br/verificar-documento-externo/> e forneça os dados abaixo:

Código Verificador: 1056162

Código de Autenticação: d6ce697f63

