



UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO  
PRÓ-REITORIA DE GRADUAÇÃO  
DEPARTAMENTO DE ENGENHARIAS E TECNOLOGIA  
CURSO INTERDISCIPLINAR EM TECNOLOGIA DA INFORMAÇÃO

MOISÉS HENRIQUE LIMA HOLANDA

**DESENVOLVENDO A INTERFACE DO APLICATIVO ROTINA DE ESTUDO PARA  
A ORQUESTRA FILARMÔNICA PAUFERRENSE**

PAU DOS FERROS - RN  
2025

MOISÉS HENRIQUE LIMA HOLANDA

**DESENVOLVENDO A INTERFACE DO APLICATIVO ROTINA DE ESTUDO PARA  
A ORQUESTRA FILARMÔNICA PAUFERRENSE**

Monografia apresentada à Universidade Federal Rural do Semi-Árido como requisito para obtenção do título de Bacharel Interdisciplinar em Tecnologia da Informação.

Orientador: Reudismam Rolim de Sousa, Prof. Dr.

PAU DOS FERROS - RN

2025

© Todos os direitos estão reservados a Universidade Federal Rural do Semi-Árido. O conteúdo desta obra é de inteira responsabilidade do (a) autor (a), sendo o mesmo, passível de sanções administrativas ou penais, caso sejam infringidas as leis que regulamentam a Propriedade Intelectual, respectivamente, Patentes: Lei nº 9.279/1996 e Direitos Autorais: Lei nº 9.610/1998. O conteúdo desta obra tomar-se-á de domínio público após a data de defesa e homologação da sua respectiva ata. A mesma poderá servir de base literária para novas pesquisas, desde que a obra e seu (a) respectivo (a) autor (a) sejam devidamente citados e mencionados os seus créditos bibliográficos.

H722d Holanda, Moisés Henrique Lima.  
Desenvolvendo a interface do aplicativo rotina  
de estudos para a Orquestra Filarmônica  
Pauferrense / Moisés Henrique Lima Holanda. -  
2025.  
31 f. : il.

Orientador: Reudismam Rolim Sousa.  
Monografia (graduação) - Universidade Federal  
Rural do Semi-árido, Curso de Tecnologia da  
Informação, 2025.

1. Orquestra. 2. Engenharia de software. 3.  
Metodologias de ensino. 4. Aplicativo. 5.  
Aplicativo rotina de estudos. I. Sousa, Reudismam  
Rolim, orient. II. Título.

Ficha catalográfica elaborada por sistema gerador automático em conformidade  
com AACR2 e os dados fornecidos pelo(a) autor(a).

Biblioteca Campus Pau dos Ferros  
Bibliotecária: Erlanda Maria Lopes da Silva  
CRB: 15/966

O serviço de Geração Automática de Ficha Catalográfica para Trabalhos de Conclusão de Curso (TCC's) foi desenvolvido pelo Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo (USP) e gentilmente cedido para o Sistema de Bibliotecas da Universidade Federal Rural do Semi-Árido (SISBI-UFERSA), sendo customizado pela Superintendência de Tecnologia da Informação e Comunicação (SUTIC) sob orientação dos bibliotecários da instituição para ser adaptado às necessidades dos alunos dos Cursos de Graduação e Programas de Pós-Graduação da Universidade.

MOISÉS HENRIQUE LIMA HOLANDA

**DESENVOLVENDO A INTERFACE DO APLICATIVO ROTINA DE ESTUDO PARA  
A ORQUESTRA FILARMÔNICA PAUFERRENSE**

Monografia apresentada à Universidade Federal Rural do Semi-Árido como requisito para obtenção do título de Bacharel Interdisciplinar em Tecnologia da Informação.

Defendida em: 19/12/2025.

**BANCA EXAMINADORA**

---

Reudismam Rolim de Sousa, Prof. Dr. – (UFERSA)  
Presidente

---

Anne Valeska Lopes da Costa, Profa. Ma. – (UERN)  
Membro Examinador

---

João Batista de Souza Neto, Prof. Dr. (UFERSA)  
Membro Examinador

---

Glécia Mesquita Freire Fernandes, Ma. (UFERSA)  
Membro Examinador

## DEDICATÓRIA

Dedico este trabalho aos meus pais, que sempre me apoiaram e são meu alicerce e exemplo de vida; à minha noiva Raigna Maristela, que sempre acreditou em mim e me faz nunca desistir; aos meus professores do curso de Bacharelado Interdisciplinar em Tecnologia da Informação; aos meus amigos e colegas da Orquestra Filarmônica Pauferrense e à todas as pessoas que fizeram e fazem parte da minha caminhada.

## **AGRADECIMENTOS**

Primeiramente, gostaria de agradecer aos meus pais, Maria Lucilene e Roberto Pereira, que sempre acreditaram em mim; a minha noiva Raigna Maristela, pelo apoio emocional e por compartilhar comigo momentos incríveis. Essa conquista e conclusão deste trabalho não seriam possíveis sem o apoio dos meus colegas da Orquestra Filarmônica Pauferrense, na minha caminhada musical, em especial, o maestro Elton Souza e o Maestro Leandro Oliveira. Agradeço também aos professores do curso de Bacharelado Interdisciplinar em Tecnologia da Informação (BTI) da UFERSA – Campus Pau dos Ferros, pois foi por meio deles que trilhei minha trajetória acadêmica; e, por fim, quero agradecer à banca examinadora para a qual vou apresentar este projeto. Agradeço a Deus pela dádiva de ter pessoas tão fantásticas em minha vida.

## RESUMO

O desenvolvimento de habilidades performáticas, técnicas e teóricas está ligado diretamente a um estudo regular individual, algo desafiador para alunos iniciantes em aulas de música, podendo levar a falhas na prática de estudo pelos alunos. Diante desse contexto, para auxiliar a rotina de estudos de alunos de música, foi feita a proposta de um aplicativo que visa facilitar o estudo individual constante, melhorando habilidades instrumentais, possibilitando ao estudante o alcance de seus objetivos musicais. O ambiente foi proposto considerando as dificuldades enfrentadas pelos músicos iniciantes da *Orquestra Filarmônica Pauferrense* e trata-se de uma interface gráfica interativa focada em reduzir a evasão de músicos iniciantes na orquestra e criar um hábito de estudo individual constante. Dentro do ciclo de desenvolvimento de software, esse sistema encontra-se como uma interface amigável e interativa, que permite ao aluno interagir com o professor de forma *online*, em que poderá executar as tarefas propostas pelo professor, ter material de apoio compartilhado por ele e comentários pontuais a cada tarefa entregue, trazendo de maneira gradativa o costume de estudar regularmente de maneira eficaz. Dessa forma, a proposta deste trabalho é o desenvolvimento da interface de usuário *mobile (front-end)* dessa plataforma. Para desenvolver o sistema, foram utilizadas a linguagens e ferramentas de programação *TypeScript*, *React Native*, *React.js* e *CSS*, que contribuem para a construção eficiente e escalável da interface. Através do uso dessas ferramentas e linguagens, foi possível desenvolver a parte *front-end* do aplicativo proposto.

**Palavras-chave:** aplicativo; metodologias de ensino; engenharia de software; orquestra; aplicativo rotina de estudos

## ABSTRACT

The development of performance, technical, and theoretical skills is directly linked to regular individual study, which can be challenging for beginner music students and may lead to failures in their study practice. In this context, to assist music students in their study routine, we proposed an application to facilitate consistent individual study, improve instrumental skills, and enable students to achieve their musical goals. The environment was designed based on difficulties faced by beginner musicians in the Orquestra Filarmonica Pauferrense and consists of an interactive graphical interface focused on reducing the dropout rate of beginner musicians in the orchestra and creating a habit of consistent individual study. In the software development cycle, this system is a user-friendly and interactive interface that allows students to interact with the teacher online, performing tasks assigned by the teacher, accessing support material shared by the teacher, and receiving specific comments on each submitted task, gradually fostering the habit of studying regularly and effectively. Therefore, the aim of this work is to develop a mobile user interface (front-end) for this platform. To develop the system, the programming languages and tools TypeScript, React Native, React.js, and CSS were used, contributing to the efficient and scalable construction of the interface. Through the use of these tools and languages, we were able to develop the front-end of the proposed application.

**Keywords:** application; teaching methodologies; software engineering; orchestra; study routine app



## LISTA DE FIGURAS

|                                                                    |    |
|--------------------------------------------------------------------|----|
| Figura 1 - Tela de escolha de perfil e de cadastro de usuário..... | 24 |
| Figura 2 - Telas de perfil .....                                   | 25 |
| Figura 3 - Telas de gerência de turma .....                        | 26 |
| Figura 4 - Tela de turma do aluno .....                            | 27 |
| Figura 5 - Telas envio de tarefa .....                             | 28 |
| Figura 6 - Telas listagem de tarefas .....                         | 29 |
| Figura 7 - Telas de gerência de participantes .....                | 30 |

## SUMÁRIO

|                                                                                          |           |
|------------------------------------------------------------------------------------------|-----------|
| <b>1 INTRODUÇÃO .....</b>                                                                | <b>11</b> |
| <b>1.1 Problematização.....</b>                                                          | <b>12</b> |
| <b>1.2 Justificativa .....</b>                                                           | <b>12</b> |
| <b>1.3 Objetivos.....</b>                                                                | <b>14</b> |
| 1.3.1 Objetivo geral .....                                                               | 14        |
| 1.3.2 Objetivos específicos .....                                                        | 14        |
| <b>1.4 Organização .....</b>                                                             | <b>15</b> |
| <b>2 REFERENCIAL TEÓRICO .....</b>                                                       | <b>15</b> |
| <b>2.1 Desenvolvimento Mobile .....</b>                                                  | <b>15</b> |
| <b>2.2 Interface de usuário (UI) e experiência de usuário (UX).....</b>                  | <b>15</b> |
| <b>2.3 TypeScript .....</b>                                                              | <b>16</b> |
| <b>2.4 Node.js .....</b>                                                                 | <b>16</b> |
| <b>2.4 Front-end.....</b>                                                                | <b>17</b> |
| <b>2.5 React .....</b>                                                                   | <b>17</b> |
| <b>2.6 React Native .....</b>                                                            | <b>17</b> |
| <b>2.7 Expo .....</b>                                                                    | <b>18</b> |
| <b>2.8 Expo GO .....</b>                                                                 | <b>18</b> |
| <b>3 TRABALHOS RELACIONADOS .....</b>                                                    | <b>19</b> |
| <b>4 METODOLOGIA DE PESQUISA .....</b>                                                   | <b>20</b> |
| <b>4.1 Implementação da Metodologia Utilizada no Desenvolvimento do Aplicativo .....</b> | <b>20</b> |
| <b>4.2 Requisitos .....</b>                                                              | <b>21</b> |
| 4.2.1 Requisitos funcionais.....                                                         | 21        |
| 4.2.2 Requisitos não funcionais.....                                                     | 22        |
| <b>5 PROPOSTA DA SOLUÇÃO.....</b>                                                        | <b>23</b> |
| <b>4.1 Diagrama de caso de uso .....</b>                                                 | <b>23</b> |
| <b>5.2 Desenvolvimento da Proposta.....</b>                                              | <b>24</b> |
| 5.2.1 Telas de Cadastro e Login .....                                                    | 24        |

|                                                                 |           |
|-----------------------------------------------------------------|-----------|
| 5.2.2 Telas de cards de grupo com turmas Professor e Aluno..... | 25        |
| 5.2.3 Tela de turmas .....                                      | 26        |
| 5.2.4 Seção de Tarefas .....                                    | 27        |
| 5.2.5 Seção de Tarefas Concluídas .....                         | 28        |
| 5.2.6 Seção de participantes .....                              | 29        |
| <b>6 CONSIDERAÇÕES FINAIS.....</b>                              | <b>31</b> |
| <b>REFERÊNCIAS.....</b>                                         | <b>32</b> |

## 1 INTRODUÇÃO

O estudo contínuo na música constitui um dos pilares fundamentais para o desenvolvimento técnico, cognitivo e expressivo do músico. A prática regular permite não apenas o aprimoramento da técnica e da *performance*, mas também o aprofundamento da compreensão teórica e estética que sustenta a prática musical (Swanwick, 2003). Nesse processo, dois elementos assumem papel central na formação do estudante: a comunicação entre aluno e professor e o estudo individual sistematizado, ambos determinantes para a construção de uma aprendizagem musical sólida, progressiva e significativa.

Segundo McPherson e Zimmerman (2002), músicos iniciantes tendem a enfrentar desafios significativos no estudo individual, uma vez que ainda não desenvolveram competências de autorregulação, como planejamento, organização e escolha adequada de estratégias durante a prática diária. Em muitos casos, a ausência de um direcionamento preciso pode levar o músico iniciante a desenvolver hábitos ineficazes, como repetir mecanicamente exercícios sem compreender seus objetivos, o que compromete o desenvolvimento técnico e interpretativo (Gordon, 1997). Além disso, Hentschke e Del Ben (2003) destacam que a dificuldade em perceber e corrigir erros de forma autônoma é uma das principais limitações na fase inicial da formação musical, tornando o processo de estudo solitário um desafio emocional e cognitivo. Com todos esses fatores, muitos alunos iniciantes desistem ou se sentem frustrados por não conseguirem avançar tecnicamente.

Diante desse cenário, é necessário buscar soluções tecnológicas e didáticas que venham como facilitador desse estudo individual, reduzindo a evasão de alunos em orquestras e cursos de música. A tecnologia, quando bem aplicada, vem a ser uma poderosa ferramenta de aprendizagem uma vez que ferramentas educacionais inovadoras possibilitam transformar a forma como o conhecimento é adquirido, promovendo um aprendizado mais acessível, dinâmico e interativo (Kenski, 2018).

O aplicativo de rotina de estudos vem como facilitador, trazendo melhorias e uma nova visão em como o ensino à distância é transmitido e compartilhado, resultando em uma ferramenta inovadora para melhorar o desempenho dos alunos e promover um aprendizado mais eficaz. O sistema oferece uma diversidade de recursos interativos que trazem uma melhor comunicação entre o aluno e o professor, uma vez que as aulas presenciais podem não ser suficientes ou ser necessárias

tarefas de reforço, compartilhamento de materiais essenciais para o avanço performático e adição de comentários como dicas para melhoria da técnica individual.

### **1.1 Problematização**

A Orquestra Filarmônica Pauferrense foi criada a partir do sonho de um cidadão Pauferrense Antônio de Holanda Cavalcante, que destinou parte de seus bens para a criação de uma orquestra filarmônica na cidade de Pau dos Ferros. Atualmente, a orquestra conta com mais de 30 músicos, em que a maioria são alunos musicistas e mais de 10 instrutores instrumentistas (Portal Sociedade Filarmônica Pauferrense, 2025). Para o desenvolvimento performático e técnico de um instrumentista, a prática instrumental e o estudo da teoria musical são fatores primordiais e devem fazer parte da rotina diária do aluno. Neste contexto, os professores e monitores auxiliam na evolução do aluno.

No entanto, torna-se impossível para os instrutores acompanhar integralmente os alunos durante a semana, uma vez que os encontros não ocorrem diariamente e os estudantes podem ter dúvidas a qualquer instante. Ademais, alguns estudantes, principalmente os mais jovens, podem ter dificuldade em desenvolver uma rotina de estudos para acompanhar os conteúdos trabalhados em sala de aula, o que pode resultar em estagnação técnica, levando à frustração, podendo entender que não conseguem evoluir no manejo dos instrumentos estudados.

Durante o período pandêmico a orquestra experimentou o ensino a distância, em que as aulas eram ministradas de forma virtual e individual com cada aluno. Apesar de algumas dificuldades, a experiência proporcionou uma reflexão sobre a importância do acompanhamento diário entre aluno e professor e a criação de uma rotina de estudos diária que ajudasse o aluno na sua evolução técnica.

### **1.2 Justificativa**

Este trabalho aborda o desenvolvimento do *front-end* de um aplicativo, que busca minimizar dificuldades enfrentadas pelos alunos iniciantes em música. As aulas presenciais de música têm duração limitada, necessitando que o aluno busque praticar os conceitos e práticas apresentadas em outros momentos. No entanto, a dificuldade de realizar o acompanhamento do conteúdo pode se tornar uma

difficuldade, podendo levar os alunos iniciantes a desistirem do aprendizado musical. Ademais, pode se tornar inviável o acompanhamento individual do estudante pelo professor, devido ao quantitativo de alunos.

Segundo Hentschke e Del Ben (2003), o desempenho de um grupo musical depende diretamente da preparação individual de seus integrantes, de modo que a ausência de estudo pessoal prejudica o resultado coletivo. Trazendo uma frustração para o aluno, por ver outros colegas avançando tecnicamente. Essa dificuldade de acompanhamento também pode levar a um atraso nos ensaios coletivos, uma vez que exige pausa para que o aluno ou o grupo de alunos tenha que estudar a técnica durante os ensaios coletivos.

Artigos de educação musical, como os de Hentschke e Del Ben (2003), destacam que muitos alunos não conseguem estudar de forma autônoma, o que gera um alto nível de frustração no processo de ensino-aprendizagem. Em vez de terem um plano claro, o estudante acaba estudando sem qualquer guia, sem compreender como dividir os exercícios, sem conseguir diagnosticar os próprios erros e, conseqüentemente, sem perceber a própria evolução. Essa falta de clareza sobre o estudo individual contribui para a desmotivação, para a sensação de incapacidade e, em muitos casos, para o abandono das práticas em grupo e das formações orquestrais.

Diante dessa problemática, é necessário buscar soluções que possam mitigar e minimizar essas dificuldades, promovendo um ambiente de aprendizado eficaz e motivador, trazendo consciência e autonomia no estudo individual. A utilização de tecnologias para a criação de ferramentas educacionais, como o aplicativo de rotina de estudos, vem como uma alternativa para isso. O aplicativo pretende oferecer uma plataforma interativa, que facilita o aprendizado e a autonomia, por meio de interações entre aluno e professor, comentários direcionados para cada tarefa e compartilhamento de material didático, ajudando o aluno a superar as dificuldades encontradas na criação de um estudo individual de qualidade.

Portanto, este trabalho visa desenvolver a interface *front-end* do aplicativo como proposta para melhorar o processo de ensino-aprendizagem e a autonomia do estudo individual. Acredita-se que, com a implementação de ferramentas tecnológicas voltadas para o âmbito educacional, auxiliadas por metodologias de ensino-aprendizagem inovadoras, é possível desenvolver as habilidades criativas de cada

aluno para desenvolver e solucionar dificuldades individuais, reduzindo a alta evasão e trazendo uma satisfação ao aluno e ao professor.

### 1.3 Objetivos

O objetivo geral e os objetivos específicos da proposta são apresentados a seguir:

#### 1.3.1 Objetivo geral

Desenvolver a interface de um aplicativo *mobile* para auxiliar na rotina de estudos de alunos musicistas e na comunicação entre aluno e professor da Orquestra Filarmônica Pauferrense. A interface deve ser simples, de fácil entendimento e responsiva, em que os alunos poderão visualizar as tarefas e os professores enviá-las para os alunos. Toda essa comunicação deve ser feita por grupos específicos para cada professor, levando em consideração o tipo de instrumento que o profissional e o aluno trabalham.

#### 1.3.2 Objetivos específicos

A fim de atender ao objetivo geral, são apresentados os objetivos específicos a seguir:

- **Desenvolver uma interface intuitiva e responsiva:** melhorar a experiência do usuário por um ambiente prático e amigável, sendo esse de fácil uso e entendimento, além de responsivo para os principais dispositivos *mobile*.
- **Analisar os princípios da interface de usuário:** fazer uma pesquisa a respeito de como implementar de maneira eficiente e adequada a aparência visual da interface do aplicativo, com o propósito de garantir que a interface seja visualmente agradável e atraente para os usuários.
- **Analisar os princípios da experiência de usuário:** pesquisar sobre como assegurar que o usuário tenha uma melhor experiência no aplicativo, visando otimizar o funcionamento técnico e organização do aplicativo.
- **Criação de um ambiente de interação entre o aluno e o professor:** este ambiente deve ser criado levando em consideração o tipo de instrumento que

o aluno utiliza e que o professor é especializado, sendo divididos em 3 tipos (Cordas, Sopros e Percussão). Posteriormente, dependendo da opção escolhida no aplicativo, devem ser apresentadas opções do instrumento do tipo escolhido e, ao selecionar o tipo de instrumento, o aluno será adicionado ao grupo referente ao instrumento escolhido.

#### **1.4 Organização**

Este trabalho apresenta a seguinte estrutura: no Capítulo 2 é apresentado o referencial teórico, em são listadas as tecnologias utilizadas no desenvolvimento desse aplicativo; no Capítulo 3 é descrito trabalhos relacionados já feitos; no Capítulo 4 é descrita a metodologia utilizada no processo de desenvolvimento; no Capítulo 5 é apresentada a implementação do aplicativo através das telas e de um diagrama de caso de uso; e no Capítulo 6 são apresentadas as considerações finais sobre o projeto.

## **2 REFERENCIAL TEÓRICO**

Neste capítulo são apresentados conceitos e abordagens fundamentais para melhor entendimento do projeto, abordando termos e tecnologias utilizadas no desenvolvimento *mobile*, como linguagens de programação e *frameworks* utilizados na construção do aplicativo.

### **2.1 Desenvolvimento Mobile**

O desenvolvimento mobile refere-se ao processo de criação de aplicativos voltados para dispositivos móveis, como smartphones e tablets, utilizando tecnologias específicas para esse ambiente. Esse desenvolvimento pode ocorrer por meio de aplicações nativas, destinadas a um sistema operacional específico, ou híbridas, que permitem maior compatibilidade entre plataformas. O processo envolve etapas como planejamento, codificação, testes e implantação. Além disso, aspectos como desempenho, usabilidade e adaptação a diferentes tamanhos de tela são fundamentais. O uso de frameworks e ferramentas especializadas contribui para maior eficiência no desenvolvimento. Esse tipo de desenvolvimento tornou-se estratégico



diante da crescente utilização de dispositivos móveis no cotidiano (BRQ Digital Solutions, 2024).

## **2.2 Interface de usuário (UI) e experiência de usuário (UX)**

A interface de usuário (UI) refere-se à parte visual de um programa, com o propósito de garantir uma melhor experiência dos usuários. É importante que um *design* seja agradável e corresponda às ações dos usuários (Silveira, 2023).

A experiência de usuário (UX) é uma área dinâmica e estratégica, que visa melhorar a experiência das pessoas que utilizam um produto (sites, aplicativos, serviços e até produtos físicos) por meio do aperfeiçoamento da parte com a qual os usuários interagem, envolvendo a organização de componentes e navegação (Matos, 2025).

## **2.3 TypeScript**

É uma linguagem de programação criada pela Microsoft em 2012 que veio com a proposta de ser um “*JavaScript melhorado*”, adicionando tipagem estática e recursos avançados ao Javascript, ampliando a linguagem para melhor produtividade e manutenção e seguindo os paradigmas da orientação a objetos.

## **2.4 Node.js**

O Node.js é amplamente utilizado no desenvolvimento front-end de aplicações como ambiente de execução para ferramentas e bibliotecas responsáveis pela construção, automação e gerenciamento do projeto. No contexto de aplicativos front-end, o Node.js possibilita o uso de gerenciadores de pacotes, como o npm, além da execução de scripts de build, testes e compilação de código, sendo essencial em frameworks modernos como React e React Native. Dessa forma, o Node.js atua como um facilitador do processo de desenvolvimento, contribuindo para maior produtividade e organização do fluxo de trabalho (NODE.JS FOUNDATION, 2024).

## 2.4 Front-end

O front-end mobile corresponde à parte visual e interativa de um aplicativo voltado para dispositivos móveis, como smartphones e tablets, sendo responsável pela interface com a qual o usuário interage diretamente. Essa camada envolve a construção de elementos visuais e funcionais que garantem a adaptação a diferentes tamanhos de tela e uma navegação fluida no contexto móvel, utilizando tecnologias e frameworks modernos que facilitam a criação de aplicações atraentes e responsivas. No desenvolvimento mobile com ferramentas como o React Native, o front-end é implementado por meio de componentes desenvolvidos em JavaScript ou TypeScript, que são convertidos em elementos nativos das plataformas Android e iOS, proporcionando uma experiência semelhante à de aplicativos nativos. A construção dessa interface é fundamental para a satisfação do usuário final, pois influencia diretamente a usabilidade, o desempenho e a aceitação do aplicativo no mercado, evidenciando o front-end mobile como uma área que integra design, lógica de apresentação e tecnologia (FLAMMA DESIGN, 2024).

## 2.5 React

O React é uma biblioteca de código aberto em JavaScript amplamente utilizada no desenvolvimento do front-end para a criação e renderização de interfaces de usuário. Sua principal característica é a utilização de componentes, que consistem em blocos de código independentes e reutilizáveis, responsáveis por definir partes específicas da interface. Essa abordagem baseada em componentes favorece a organização do projeto, a reutilização de código e a manutenção da aplicação, além de facilitar o desenvolvimento de interfaces dinâmicas e eficientes. O React adota um modelo declarativo, permitindo que o desenvolvedor descreva como a interface deve se comportar em diferentes estados da aplicação, o que contribui para maior clareza e previsibilidade no desenvolvimento (REACT, 2024).

## 2.6 React Native

O React Native é um *framework* que permite aos desenvolvedores criarem aplicativos nativos para Android e iOS. Esse *framework* utiliza a ferramenta Expo para

facilitar o desenvolvimento dos aplicativos com mais agilidade, Javascript para acessar APIs, como também React para o uso de componentes (Corti, 2024).

## **2.7 Expo**

O Expo é uma ferramenta de código aberto utilizada no desenvolvimento de aplicações com React Native, cujo objetivo é simplificar e acelerar o processo de criação de aplicativos móveis. Essa plataforma oferece um conjunto abrangente de bibliotecas, APIs e ferramentas prontas que permitem o acesso a recursos nativos do dispositivo, como câmera, sensores, armazenamento e notificações, sem a necessidade de configurações complexas. Além disso, o Expo facilita o processo de testes e desenvolvimento ao reduzir a dependência de compilação manual, tornando o fluxo de trabalho mais ágil e acessível, especialmente em fases iniciais do projeto e em ambientes educacionais. Dessa forma, o Expo contribui para maior produtividade e eficiência no desenvolvimento de aplicações mobile baseadas em React Native

## **2.8 Expo GO**

O Expo Go é um aplicativo compatível com os sistemas Android e iOS que permite a execução e visualização de projetos desenvolvidos em React Native de forma imediata, sem a necessidade de compilar o código ou gerar arquivos APK ou IPA. Ele funciona como um ambiente de execução que possibilita ao desenvolvedor testar a aplicação diretamente em um dispositivo físico por meio da leitura de um QR Code, agilizando significativamente o processo de desenvolvimento e validação. O Expo Go oferece acesso a diversas APIs nativas do dispositivo, como câmera, sensores, geolocalização e armazenamento, sem exigir configurações complexas ou conhecimentos avançados de compilação nativa. Dessa forma, essa ferramenta contribui para maior produtividade, rapidez nos testes e facilidade no desenvolvimento de aplicativos mobile, sendo amplamente utilizada em protótipos e projetos educacionais

### 3 TRABALHOS RELACIONADOS

Nos últimos anos, diferentes aplicativos e plataformas digitais têm sido desenvolvidos com o objetivo de auxiliar estudantes na organização de rotinas de estudo e no aprimoramento do aprendizado musical. Ferramentas voltadas à gestão do tempo e da produtividade, como Forest, Todoist e Notion, têm sido amplamente utilizadas para estruturar atividades diárias, definição de metas e acompanhamento do progresso, sendo frequentemente adaptadas por estudantes de música para a criação de planos de prática instrumental. Embora não sejam específicas da área musical, essas plataformas contribuem para o desenvolvimento da autonomia e da disciplina, aspectos fundamentais para o estudo individual.

No campo específico da educação musical, destacam-se aplicativos que combinam tecnologia interativa com propostas pedagógicas voltadas à prática instrumental e ao aprendizado musical. O Yousician, por exemplo, oferece lições guiadas e feedback em tempo real para instrumentos como guitarra, piano e baixo, auxiliando o estudante na correção de erros durante a prática. De forma semelhante, o Simply Piano e o Simply Guitar apresentam abordagens progressivas voltadas principalmente para iniciantes, incentivando a continuidade dos estudos por meio de exercícios estruturados.

Outras plataformas têm como foco o apoio técnico e teórico ao estudo musical. O MuseScore possibilita a leitura, edição e criação de partituras, funcionando como um recurso complementar tanto para o estudo individual quanto para o ensino formal. Já ferramentas como TonalEnergy e EarMaster são amplamente utilizadas no desenvolvimento de habilidades essenciais, como afinação, ritmo, percepção auditiva e leitura musical, contribuindo para uma prática mais consciente e direcionada.

Apesar da existência dessas soluções, observa-se que grande parte dos aplicativos se concentra em aspectos isolados do estudo musical, como técnica, teoria ou organização do tempo, sem integrar de forma abrangente a construção de uma rotina de estudo personalizada. Nesse contexto, trabalhos que proponham o desenvolvimento de plataformas capazes de unir planejamento, acompanhamento e estratégias pedagógicas específicas para o estudo musical apresentam grande relevância, especialmente por favorecerem a autonomia, a organização e a eficiência da prática individual.

## 4 METODOLOGIA DE PESQUISA

Este capítulo descreve a abordagem metodológica utilizada para o desenvolvimento do aplicativo, desenvolvido centrado nos usuários.

Para garantir que o sistema atenda às necessidades dos usuários, foi definido um processo de coleta de requisitos, com planejamento, que incluiu um ciclo de entregas e coleta de *feedbacks* ao longo da evolução do software. A cada entrega, foi feita uma coleta de dados para incorporação futura ao sistema, assegurando que o produto final atenda ao público-alvo.

### 4.1 Implementação da Metodologia Utilizada no Desenvolvimento do Aplicativo

Neste projeto, a metodologia utilizada para o desenvolvimento foi baseada em uma abordagem de entregas contínuas por reuniões com os usuários, permitindo ajustes frequentes conforme as necessidades dos usuários. A seguir, são descritas as etapas da metodologia aplicada no desenvolvimento do aplicativo.

- **Coleta e análise de requisitos:** foi realizada uma reunião via *google meet* para a coleta de requisitos, em que os usuários puderam informar as necessidades do sistema, o propósito, o que o sistema deveria ter e como ele iria interagir com os usuários. Foram realizadas anotações ao longo desse processo. Essas reuniões foram realizadas durante o processo de desenvolvimento do sistema, garantindo um fluxo de entregas contínuas e coleta de *feedbacks*.
- **Definição de prioridades:** foi feita uma priorização para garantir um fluxo de evolução contínuo e eficaz, levando em consideração a complexidade e necessidade de cada requisito. O objetivo desta etapa é garantir que todas as funcionalidades essenciais pudessem ser finalizadas rapidamente.
- **Desenvolvendo a Interface *Front-end*:** Com as prioridades estabelecidas e os requisitos levantados, deu-se início ao processo de desenvolvimento do aplicativo utilizando tecnologias e ferramentas adequadas ao sistema. Nesta fase, houve a realização de reuniões para discutir o progresso e novas sugestões de implementação. O ciclo de processo iterativo e incremental utilizado no projeto permitiu entregas e evoluções contínuas.

A metodologia foi aplicada de maneira cíclica, em que, a cada nova iteração, foram feitos ajustes e atualizações no sistema, baseados nos *feedbacks* dos usuários. A entrega contínua e a coleta de *feedbacks* garantiram um processo de evolução eficaz, permitindo um alinhamento constante das expectativas do usuário e da boa qualidade do aplicativo.

## 4.2 Requisitos

Nesta seção, são descritos os requisitos funcionais e não funcionais definidos para o ambiente.

### 4.1.1 Requisitos funcionais

- **[RF001]** — Realizar Cadastro do usuário como Aluno: o aplicativo deve permitir que o usuário possa realizar o cadastro como aluno, inserindo seus dados e escolhendo a credencial “Aluno” na tela de cadastro.
- **[RF002]** — Realizar Cadastro do usuário como Professor: o aplicativo deve permitir que o usuário possa realizar o cadastro como aluno, inserindo seus dados e escolhendo a credencial “Professor” na tela de cadastro.
- **[RF003]** — Realizar *Login* como Aluno: o sistema deve permitir que o usuário possa realizar seu *login* a partir dos dados cadastrados e da credencial “Aluno” configurada anteriormente.
- **[RF004]** — Realizar *Login* como Professor: o sistema deve permitir que o usuário possa realizar seu *login* a partir dos dados cadastrados e da credencial “Professor” configurada anteriormente.
- **[RF005]** — Cadastrar Turmas: o sistema deve permitir que o professor possa criar turmas nos grupos (Sopro, Cordas e Percussão).
- **[RF006]** — Visualizar informações da turma: o sistema deve permitir que o professor ou aluno possa visualizar as informações da turma, como (nome, *id*, nome do usuário e tipo de usuário) na turma que ele entrou.
- **[RF007]** — Cadastrar Tarefa: o aplicativo deve permitir que o professor possa cadastrar uma nova tarefa na turma, adicionando informações da tarefa.

- **[RF008]** — Inserir Mídias de Referência: o software deve permitir que o professor possa inserir mídias de referência para o aluno, como vídeos e documentos PDFs.
- **[RF009]** — Visualizar Tarefas cadastradas: o sistema deve permitir que o professor ou o aluno possa visualizar as tarefas cadastradas.
- **[RF010]** — Visualizar Tarefas concluídas: o sistema deve permitir que o professor ou aluno possa visualizar as tarefas concluídas pelo aluno.
- **[RF011]** — Visualizar participantes da Turma: o sistema deve permitir que o professor ou aluno possa visualizar os participantes que estão na turma.
- **[RF012]** — Adicionar novo participante: o sistema deve permitir que o professor possa adicionar um novo participante na turma.
- **[RF013]** — Adicionar Comentário na Tarefa: o sistema deve permitir que o professor possa adicionar um comentário nas tarefas concluídas pelo aluno.
- **[RF014]** — Realizar Tarefa: o aplicativo deve permitir que o aluno possa concluir a tarefa cadastrada.

#### 4.1.2 Requisitos não funcionais

- **[RNF001]** — Interface Gráfica: a interface do sistema deve satisfazer as premissas da Engenharia de Software, garantindo assim a satisfação dos usuários.

## 5 PROPOSTA DA SOLUÇÃO

Neste capítulo, é detalhado o desenvolvimento do aplicativo, focando na interface gráfica e nas funcionalidades propostas pelos usuários. O objetivo deste capítulo é apresentar, por meio de telas e de um diagrama, como o sistema se comporta e interage com o usuário, mostrando como foi implementada a estrutura do projeto e como o usuário pode acessar as funcionalidades de envio de documentos didáticos, criação de turmas e de tarefas, e o fluxo de navegação entre telas.

### 5.1 Diagrama de Caso de Uso

Nessa seção será apresentado o diagrama de como o usuário poderá interagir com o sistema.





## 5.2 Desenvolvimento da Proposta

Nessa seção, são apresentados os resultados referentes ao desenvolvimento da interface proposta, focando nas telas e interações principais propostas para a solução dos problemas apresentados.

### 5.2.1 Telas de Cadastro e *Login*

Na Figura 1a é apresentada a tela de escolha de perfil, que possibilita ao usuário entrar no sistema. O usuário pode escolher entre dois tipos de credenciais: professor e aluno. A escolha é feita clicando nos botões “Sou Professor” ou “Sou Aluno”. Caso o usuário não possua uma conta, ele pode realizar o cadastro, clicando no botão “Cadastre-se”, que o direciona para a tela da Figura 1b, em que ele poderá informar seus dados e escolher a sua credencial, onde caso seja professor haverá um campo a mais chamado “matrícula”. Após o usuário preencher as suas informações, ele pode clicar no botão “Entrar” para ser direcionado à tela de *login* do aplicativo (Figura 1c). Para realizar o *login*, o usuário precisa informar seus dados de entrada (e-mail e senha) e clicar no botão “Entrar”. Caso os dados estejam corretos, o usuário entrará no sistema.

**Figura 1** - Tela de escolha de perfil e de cadastro de usuário

(a) Tela escolha perfil

(b) Tela de cadastro

(c) Tela de *login*



Fonte: Autoria própria (2025)

### 5.2.2 Telas de cards de grupo com turmas Professor e Aluno

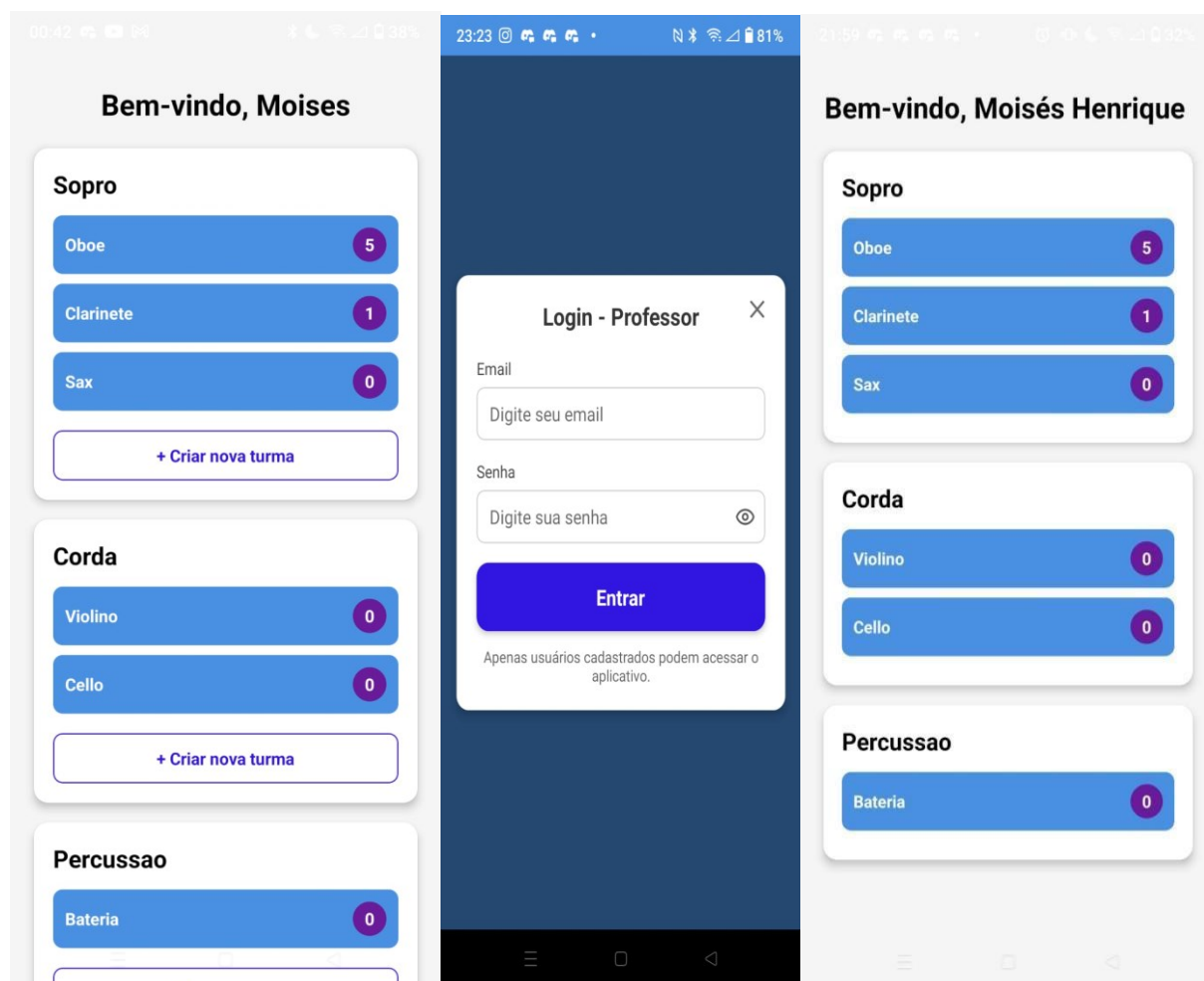
Após o usuário entrar no aplicativo, ele verá uma tela com 3 grupos, sendo eles: sopro, corda e percussão. A depender da credencial, a tela terá interações diferentes com o usuário. Caso a credencial seja do tipo “Professor”, o usuário poderá visualizar, entrar e/ou criar uma nova turma em um dos grupos, conforme mostra a Figura 2a. Ao professor clicar no botão “Criar uma nova turma”, um *modal* (*pequena tela sobreposta a tela principal*) será ativado para que o usuário possa inserir o nome da nova turma, como mostra a Figura 2b. Todavia, se a credencial for do tipo “Aluno”, o usuário só poderá visualizar e entrar nas turmas cadastradas no aplicativo (Figura 2c).

**Figura 2** - Telas de perfil

(a) Tela de perfil professor

(b) *Modal* criar turma

(c) Tela de perfil aluno



Fonte: Autoria própria (2025)

### 5.2.3 Tela da turma

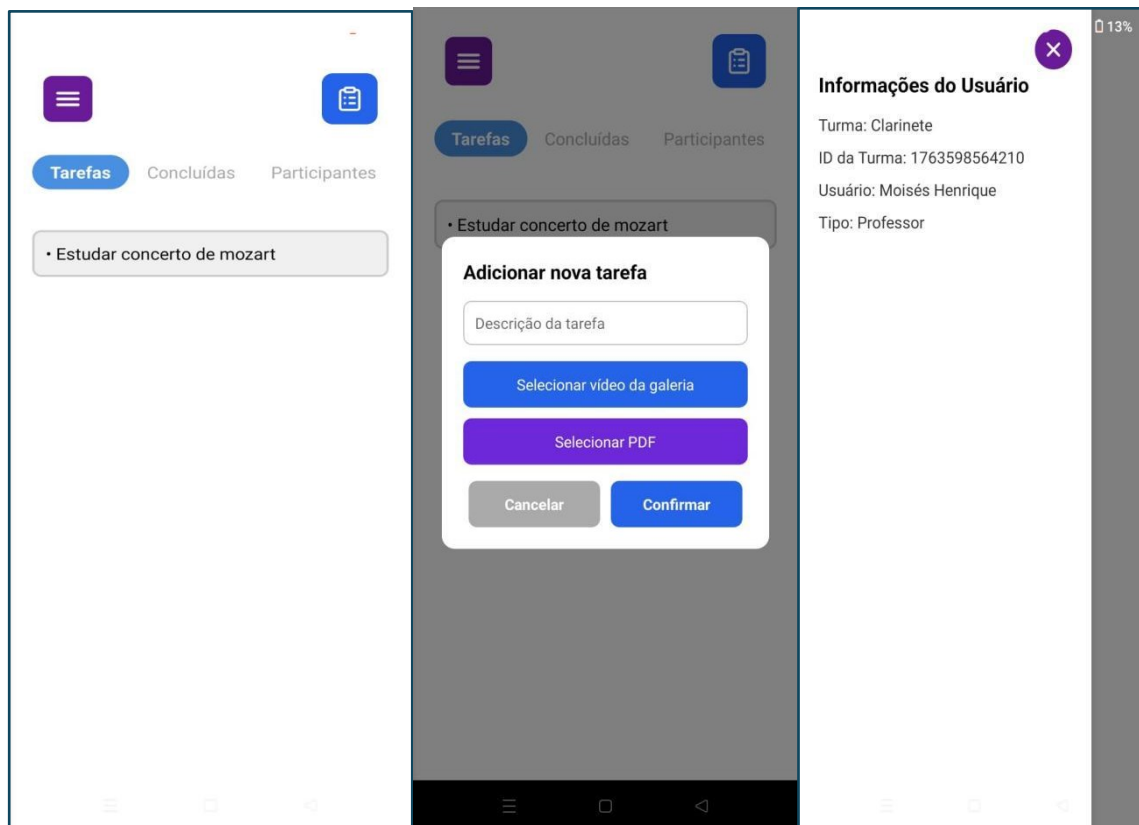
Ao entrar em uma das turmas cadastradas, o usuário será direcionado para uma tela, em que terá interações diferentes dependendo de sua credencial. Como o usuário seja um professor, ele verá duas opções no topo da tela e três seções abaixo dos botões, que são “Tarefas”, “Concluídas” e “Participantes”. As opções acima das seções permitem visualizar as informações da turma e adicionar a tarefa, respectivamente (Figura 3a). Ao clicar no botão de cadastrar nova tarefa (representada por um ícone de um livro), um *modal* é aberto para o professor adicionar o material de apoio (documento PDF e/ou vídeo) e o título da tarefa, conforme pode ser visto na Figura 3b, onde ao clicar em qualquer uma das opções o usuário será redirecionado para a galeria de arquivos de seu dispositivo. A opção para ver as informações da turma (representada por um ícone de 3 barras na horizontal) abre um “Drawer” (pequena tela lateral), em que o usuário poderá ver as informações da turma, como: ID, tipo de usuário, nome do usuário e nome da turma (Figura 3c).

Figura 3 - Telas de gerência de turma

(a) Tela de tarefas

(b) *Modal* adicionar tarefa

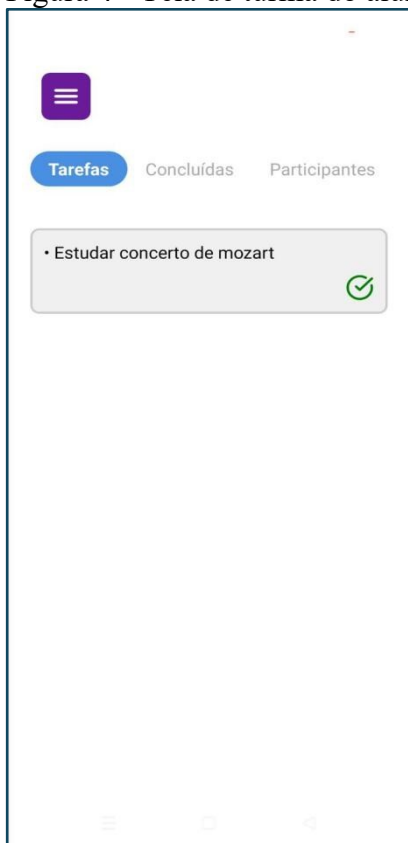
(c) Tela dados turma



Fonte: Autoria própria (2025)

Quando o usuário for um aluno, ele terá apenas o botão de visualizar as informações da turma e as três seções da turma (Figura 4)

Figura 4 - Tela de turma do aluno



Fonte: Autoria própria (2025)

#### 5.2.4 Seção de Tarefas

Nesta seção, será mostrada a interação com as tarefas. Caso o usuário seja do tipo professor, ele só poderá ver as tarefas cadastradas (Figura 3a — apresentada anteriormente). Já caso o usuário seja do tipo aluno, o *card* da tarefa mostrará um botão “Concluir Tarefa”, representado por um ícone verde (Figura 4 — apresentada anteriormente). Quando esse botão é clicado, um *modal* é aberto para que o aluno possa enviar um vídeo da tarefa concluída (Figura 5). Ao clicar em confirmar, a tarefa é enviada para análise do professor.

Figura 5 - Tela envio de tarefa



Fonte: Autoria própria (2025)

#### 5.2.5 Seção de Tarefas Concluídas

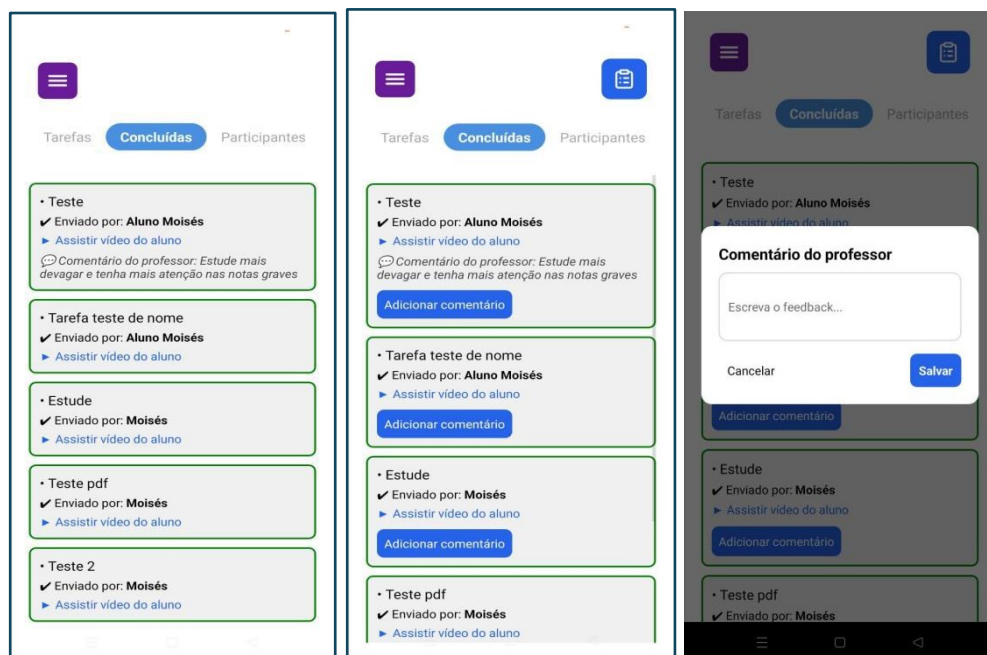
Nesta seção, são apresentadas telas relacionadas às tarefas concluídas pelo aluno. Cada tarefa possui um nome, o aluno que enviou, o vídeo e o comentário do professor (Figura 6a). Na Figura 6b, é mostrada a listagem de tarefas na tela de um usuário do tipo professor. Nessa tela, é possível ver uma ação adicional, que permite ao professor adicionar comentários. Para isso, ele deve clicar no botão “Adicionar comentário”, que faz com que seja aberto um *modal* para a adição de *feedback* para a tarefa escolhida (Figura 6c).

Figura 6 - Telas de listagem de tarefas

(a) Listagem tarefas

(b) Tarefas professor

(c) Envio de comentário



Fonte: Autoria própria (2025)

## 5.2.6 Seção de participantes

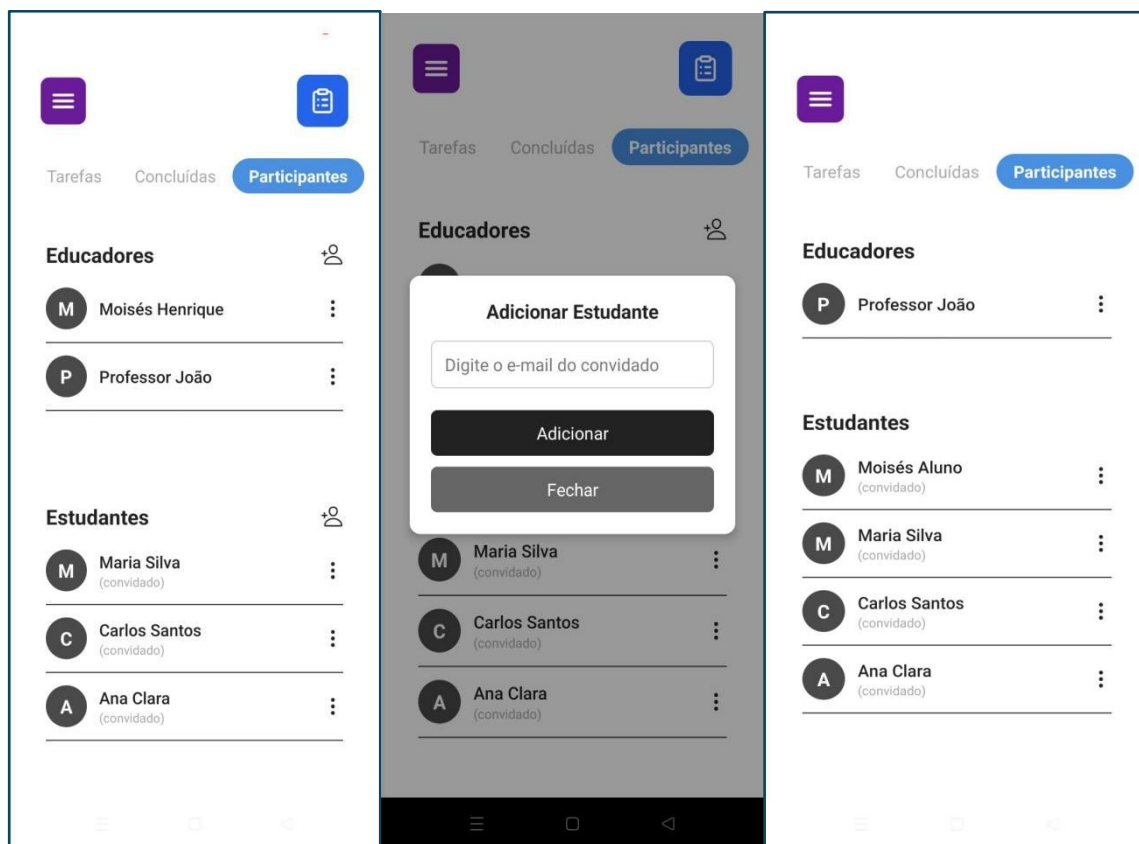
Nesta seção, os usuários poderão ver todos os participantes da turma, que são separados por “Estudantes” e “Educadores”. Caso o usuário seja do tipo professor, ele poderá convidar um novo membro para a turma, clicando no ícone em formato de boneco de usuário (Figura 7a). Ao clicar nesse ícone, é mostrado um *modal* para inserir o e-mail do participante a ser convidado (Figura 7b). Caso o usuário seja do tipo aluno, ele apenas poderá visualizar os participantes (Figura 7c).

Figura 7 - Telas de gerência de participantes

(a) Listagem participantes

(b) Adicionar estudante

(c) Listagem participantes (aluno)



Fonte: Autoria própria (2025)

## 6 CONSIDERAÇÕES FINAIS

Este trabalho abordou o desenvolvimento de um aplicativo que visa auxiliar o ensino-aprendizado individual dos alunos iniciantes em música. A proposta busca a criação de uma rotina de estudos eficaz que facilite a aproximação do aluno com o professor de forma *online*. O sistema foi pensado voltado aos alunos iniciantes em música na Orquestra Filarmônica Pauferrense, focando especialmente na dificuldade do aluno em ter um estudo individual constante, uma vez que se torna inviável para o professor realizar um acompanhamento individual personalizado ao longo das aulas presenciais. Esse acompanhamento é importante para favorecer o aprendizado constante, seja teórico, técnico ou performático.

O aplicativo apresenta ser um potencial válido como auxiliador no processo de ensino-aprendizagem individual, trazendo para os alunos uma forma de criar e tornar uma rotina de estudos um hábito, o que torna o processo mais interativo e dinâmico. Contudo, o aplicativo ainda requer dados estatísticos comprovando sua eficácia, sendo necessária a implantação e integração com um *back-end* para armazenamento dos dados dos usuários, das informações das turmas e do material didático de apoio, além de estilizar melhor a interface com a proposta orquestral.

Concluindo, espera-se que o aplicativo seja um elemento fundamental para o estudo individual, ao processo de ensino-aprendizagem, na criação de uma rotina de estudos e na comunicação entre professor e aluno fora da sala de aula. O sistema não só oferece uma maneira de ter um estudo constante como também incentiva o aluno a ter o hábito de estudo de maneira regular, criando atividades de maneira a ajudá-lo a progredir.

Como trabalhos futuros, pretende-se desenvolver um *back-end* para posteriormente ser integrado na interface, além de ajustes de *layout* na interface. Também pretende-se incluir LLMs (*Large Language Models*), para auxiliar o professor na geração de tarefas para os alunos.



## REFERÊNCIAS

AQUINO, Tainá. **Front-end: o que é, quais as diferenças, linguagens e frameworks**. UDS. [s.l.], 2024. Disponível em: <https://uds.com.br/blog/front-end-o-que-e-linguagens-frameworks/>. Acesso em 01 de out. 2025.

BRQ DIGITAL SOLUTIONS. **Desenvolvimento mobile: o que é, desafios e tendências**. BRQ Digital Solutions. [s.l.], 25 de mar. 2024. Disponível em: <https://blog.brq.com/desenvolvimento-mobile/>. Acesso em: 01 de out. 2025.

CORTI, Nicola. **Use a framework to build React Native apps**. React Native, 2024. Disponível em: <https://reactnative.dev/blog/2024/06/25/use-a-framework-to-build-react-native-apps>. Acesso em: 01 de out. 2025.

DEVMEDIA. **TypeScript**. DevMedia, 2024. Disponível em: <https://www.devmedia.com.br/typescript/>. Acesso em: 13 out. 2025.

DEL BEN, Luciana; HENTSCHE, Liane. **Educação musical escolar: uma investigação a partir das concepções e ações de três professoras de música**. Revista da ABEM, Porto Alegre, V. 7, 49-57, set. 2002.

EXPO. **Create amazing apps that run everywhere**. Disponível em: <https://docs.expo.dev/>. Acesso em: 01 de out. 2025.

EARMATER. **EarMaster: Music theory and ear training**. Disponível em: <https://www.earmaster.com/>. Acesso em: 29 dez. 2025.

FLAMMA DESIGN. *O que é Front-end Mobile?* Disponível em: <https://flammadeign.com.br/glossario/o-que-e-front-end-mobile/>. Acesso em: 29 dez. 2025.

FREITAS, Mauricio. **Desenvolvimento mobile: conheça os principais tipos, metodologias e tendências para inovar**. SoftDesign. Porto Alegre, 2024. Disponível em: <https://softdesign.com.br/blog/desenvolvimento-mobile/>. Acesso em: 01 de out. 2025.

GORDON, Edwin. *Learning Sequences in Music*. Chicago: GIA Publications, 1997.  
MCIPHERSON, Gary E.; ZIMMERMAN, Barry J. **Self-regulation of musical learning: A social cognitive perspective**. In: COLWELL, R.; RICHARDSON, C. (org.). *The new handbook of research on music teaching and learning*. Oxford: Oxford University Press, 2002.

IMPACTA. **Typescript x JavaScript: entenda o que é e saiba as diferenças**. BlogImpacta, 2024. Disponível em: <https://www.impacta.com.br/blog/typescript-javascript-entenda-que-saiba-diferencas/>. Acesso em: 13 out. 2025.

KENSKI, Vani Moreira. **Tecnologias e ensino híbrido: caminhos para a inovação educacional**. Revista Educação & Linguagem, v. 21, n. 2, p. 11–28, 2018.

MDN WEB DOcs. **O que é JavaScript?**, 2025. Disponível em: [https://developer.mozilla.org/pt-BR/docs/Learn\\_web\\_development/Core/Scripting/What\\_is\\_JavaScript](https://developer.mozilla.org/pt-BR/docs/Learn_web_development/Core/Scripting/What_is_JavaScript). Acesso em: 01 de out. 2025.

MUESCORE. **MuseScore:** Free music notation software. Disponível em: <https://musescore.org/>. Acesso em: 29 dez. 2025.

NODE.JS FOUNDATION. About Node.js. Disponível em: <https://nodejs.org/en/about>. Acesso em: 29 dez. 2025.

REACT. Introducing React. Disponível em: <https://react.dev/learn>. Acesso em: 29 dez. 2025.

SACRAMENTO, Gabriel. **O que é React e como usar essa biblioteca javascript na programação?** Tera Blog. [s.l.]. Disponível em: <https://blog.somostera.com/desenvolvimento-web/o-que-e-react>. Acesso em: 01 de out. 2025.

SOCIEDADE FILARMÔNICA PAUFERRENSE. **Orquestra Filarmônica Pauferrense “Antônio de Holanda Cavalcanti”**. Disponível em: <https://portalsfp.com/nossa-historia/> acesso em: 25 dez. 2025.

SIMPLY PIANO. **Simply Piano by JoyTunes**. Disponível em: <https://www.joytunes.com/>. Acesso em: 29 dez. 2025.

YOUSICIAN. **Yousician:** Learn to play music. Disponível em: <https://yousician.com/>. Acesso em: 29 dez. 2025.