



INSTITUTO FEDERAL DO PIAUÍ – IFPI
PRÓ-REITORIA DE ENSINO
DIRETORIA DE ENSINO
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E
DESENVOLVIMENTO DE SISTEMAS

RELATÓRIO TÉCNICO DE SOFTWARE

Projeto Tower: Uma Plataforma para o Gerenciamento de Tarefas em
Projetos de Tradução por Fãs

SAMUEL GONÇALVES MARTINS

Prof. Me. Aislan Rafael Rodrigues de Sousa

ORIENTADOR



PICOS, PI

2024

SAMUEL GONÇALVES MARTINS

Projeto Tower: Uma Plataforma para o Gerenciamento de Tarefas em Projetos
de Tradução por Fãs

Relatório Técnico de Software apresentado como exigência parcial para obtenção do diploma do Curso de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia do Piauí, Campus Picos.

Orientador: Prof. Me. Aislan Rafael Rodrigues de Sousa

PICOS

2024

RESUMO

A demanda no mercado de jogos eletrônicos no Brasil vem crescendo a cada ano, mas há casos onde o público não consegue consumir os jogos lançados pois não estão localizados para o Português Brasileiro. Em vista disso, várias equipes se formam para fazer um trabalho de tradução independente para estes jogos. Mas, costumeiramente, as equipes são incentivadas principalmente pela paixão, o que faz com que os projetos sejam mal-organizados e com baixa eficiência. Este trabalho propõe uma solução utilizando o método Kanban para organizar as equipes de tradução independente.

***Palavras-chave:** tradução, jogos eletrônicos, kanban

SUMÁRIO

1. INTRODUÇÃO	5
1.1. OBJETIVOS	6
1.1.1. Geral	6
1.1.2. Específicos	6
2. TECNOLOGIAS ENVOLVIDAS	6
3. MODELAGEM DO PROJETO	8
3.1. LEVANTAMENTO DE REQUISITOS	8
3.2. ARQUITETURA DO SISTEMA	8
3.3. INTERFACE	9
4. SOFTWARE	13
4.1. IMPLANTAÇÃO	13
4.2. TESTES	13
5. CONSIDERAÇÕES FINAIS	13
REFERÊNCIA	13

1. INTRODUÇÃO

A demanda e o público do mercado de jogos eletrônicos vem crescendo cada vez mais no Brasil. De acordo com a Pesquisa Game Brasil 2023, cerca de 70,1% da população brasileira afirma jogar algum tipo de jogo eletrônico (GO GAMERS; SX GROUP, 2023).

É bastante comum que os jogos eletrônicos sejam lançados em inglês, e às vezes seja considerado uma localização para português brasileiro. Isso deixa uma lacuna para os consumidores que não possuem conhecimento da língua.

Em vista disso, começaram a surgir diversos grupos organizados por fãs para a realização do trabalho de tradução e localização de jogos eletrônicos que não a receberam. Muitas das vezes, eles são motivados apenas pela paixão pelo jogo e não costumam possuir experiência na área de tradução, o que costuma gerar projetos demorados, mal-organizados e muitas vezes com baixa qualidade.

Com o Brasil sendo o maior mercado consumidor de jogos eletrônicos da América Latina, com 74,5% da população nacional declarando que consome jogos eletrônicos com frequência (GO GAMERS; SX GROUP, 2022), o país se torna bastante chamativo para o que lançamentos de jogos, mas com apenas 10% da população brasileira com 18 a 23 anos de idade declarando que falam inglês (British Council, 2014), isso torna bastante difícil para que a grande parte da população do público-alvo consuma os jogos que não são lançados com a localização. Em vista disso, fãs se juntam para criar e disponibilizar uma tradução por maneiras extra-oficiais. Mas é comum que as equipes criadas pelos fãs estejam mal organizadas.

Desta forma, é proposto a criação de uma ferramenta para organizar a equipe, as tarefas e os recursos de uma tradução independente, utilizando conceitos de engenharia de software, como metodologias ágeis, para a organização e priorização de tarefas, assim como ferramentas para revisão e avaliação das mesmas.

O Kanban é um sistema visual utilizado para organizar o trabalho realizado por uma equipe, garantindo a execução das atividades de forma equilibrada e que a equipe tenha condições de exercê-las. Para isso, o fluxo de entrega da equipe é

dividido em etapas e cada tarefa é separada em cada uma dessas etapas, passando de uma a outra durante o seu ciclo de vida (ANDERSON, 2016). Portanto, este trabalho propõe a criação de uma ferramenta que utiliza do método Kanban para a organização das tarefas de uma equipe de tradução independente para que o trabalho seja mais eficiente e que melhore a qualidade do resultado final.

1.1. OBJETIVOS

Este projeto visa estabelecer uma abordagem sistemática e eficiente para o gerenciamento de projetos de tradução, especialmente no contexto de traduções de jogos realizadas por fãs. A seguir, apresentamos o objetivo geral e os objetivos específicos que norteiam este trabalho.

1.1.1. Geral

Desenvolver uma solução para o gerenciamento de projetos de tradução e suas respectivas tarefas, utilizando o método Kanban como base para a delimitação do estado das tarefas.

1.1.2. Específicos

- Pesquisar sobre o mercado de games e a organização das equipes de traduções de jogos feitas por fãs.
- Propor uma solução que possibilita a organização das equipes evitando retrabalho e acelerando a tradução.
- Implementação da Solução da proposta através de uma plataforma *online*.
- Avaliação da implementação da solução proposta através de um grupo controlado de tradutores usando uma pesquisa com o método System Usability Scale (BROOKE, 1995).

2. TECNOLOGIAS ENVOLVIDAS

A seleção das tecnologias Visual Studio Code, Flutter, `firebase_auth`, `cloud_firestore` e `firebase_storage` para o desenvolvimento do projeto em questão é fundamentada visando a otimização do processo de desenvolvimento e a garantia de um produto final robusto, escalável e com alta disponibilidade.

O Visual Studio Code é adotado como ambiente de desenvolvimento integrado (IDE) devido à sua natureza extensível, leve e compatível com diversas linguagens de programação, além de oferecer suporte integrado para o controle de versão Git, facilitando assim a gestão de código-fonte e colaboração entre desenvolvedores.

O Flutter é selecionado como o framework de desenvolvimento devido à sua capacidade de compilar para código nativo, permitindo o desenvolvimento de aplicações multiplataforma (iOS e Android) a partir de uma única base de código. Esta escolha assegura eficiência no desenvolvimento e consistência na experiência do usuário final, além de proporcionar alta performance das aplicações desenvolvidas.

Para a gestão de autenticação de usuários, armazenamento e sincronização de dados em tempo real, bem como armazenamento de arquivos, são escolhidas as bibliotecas `firebase_auth`, `cloud_firestore` e `firebase_storage`, respectivamente. Estas tecnologias, parte do ecossistema Firebase, foram selecionadas por sua robustez, escalabilidade automática e segurança. Elas oferecem uma solução integrada que facilita o desenvolvimento de funcionalidades complexas, como autenticação de usuários e sincronização de dados, permitindo aos desenvolvedores concentrarem-se na lógica de negócios e na experiência do usuário, ao invés de questões de infraestrutura.

Em suma, a escolha dessas tecnologias para o projeto é justificada pela busca de eficiência no desenvolvimento, pela capacidade de oferecer uma experiência de usuário superior em diferentes plataformas e pela necessidade de uma solução escalável e segura para gestão de dados e autenticação de usuários. Este conjunto de tecnologias proporciona uma base sólida para o desenvolvimento de aplicações modernas, atendendo às exigências de projetos de alta complexidade e relevância.

Quadro 1 - Ferramentas e versões das tecnologias envolvidas

Ferramenta	Versão
Visual Studio Code	1.85.2
Flutter	3.16.0
firebase_auth	4.16.0
cloud_firestore	4.14.0
firebase_storage	11.6.0

3. MODELAGEM DO PROJETO

A aplicação foi pensada como uma aplicação para desktop, como mais uma ferramenta no conjunto que os membros da equipe de tradução usam. Ao mesmo tempo, as equipes costumam ter mais de um projeto, então a separação de cada projeto e suas tarefas é essencial.

3.1. LEVANTAMENTO DE REQUISITOS

Os requisitos são definidos em:

- Requisitos funcionais:
 1. Gerenciamento de Projetos - É possível criar e editar projetos.
 2. Gerenciamento da equipe de um projeto - Adicionar e remover outros usuários como membros da equipe de um projeto
 3. Gerenciamentos de tarefas - Criar, editar e apagar tarefas em um projeto. Além de assinalar membros a uma tarefa e alterar seu estado.
 4. Discussão em uma tarefa - Uma interface simples para que os usuários possam se comunicar dentro de uma tarefa.
- Requisitos não-funcionais:
 1. Disponibilidade – Os dados devem estar disponíveis a todo instante em que forem solicitados;

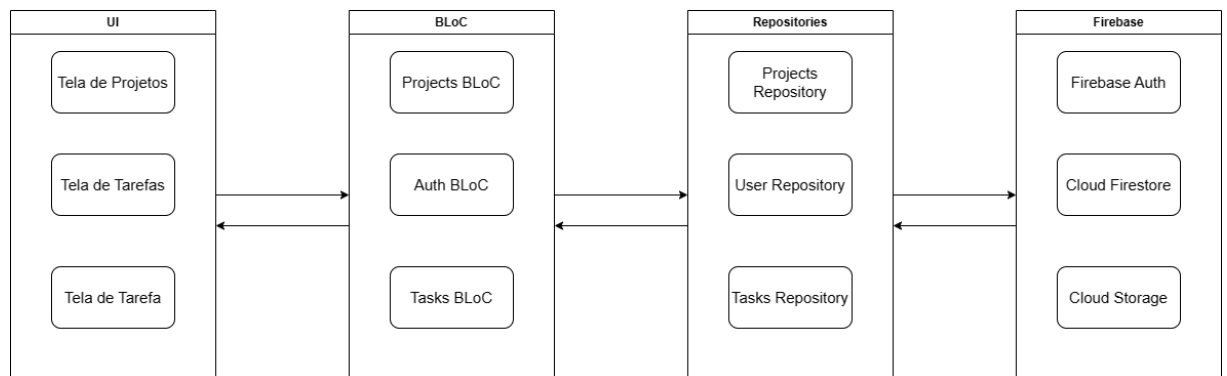
2. Plataforma - A aplicação deve ser executada no sistema operacional Microsoft Windows.

3.2. ARQUITETURA DO SISTEMA

A aplicação foi construída usando o padrão BLoC (*Business Logic Component*), um padrão onde a regra de negócio da aplicação é separada da interface usuário, separando as responsabilidades e facilitando seu reuso.

Também foi utilizado o Repository Pattern, um padrão onde o acesso e manipulação dos dados no banco de dados e/ou no serviço de armazenamento fica separado do resto da aplicação. Abaixo pode-se observar um diagrama descrevendo a arquitetura

Figura 1 - Diagrama de Arquitetura

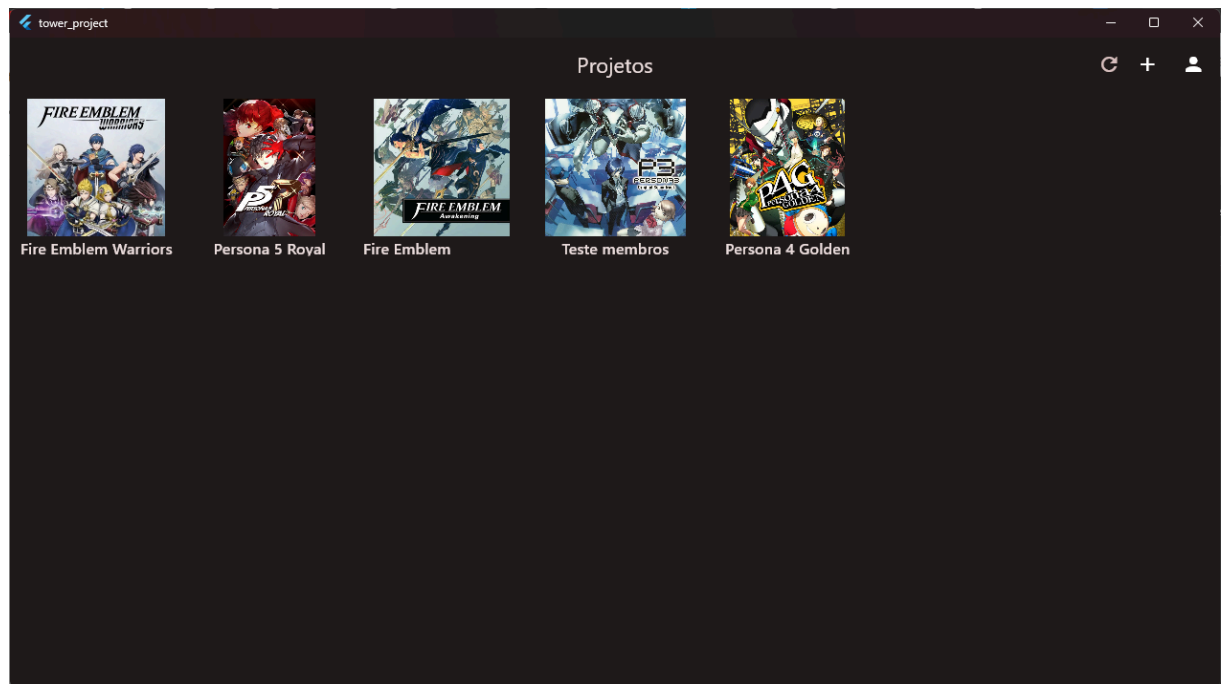


Fonte: Autoria própria.

3.3. INTERFACE

Aqui será relacionado as interfaces que implementam os requisitos. Cada jogo a ser traduzido é tratado como um projeto. Na tela de projetos, são exibidos todos os projetos onde o usuário faz parte como equipe e/ou criador do projeto. Como pode ser observado na figura abaixo.

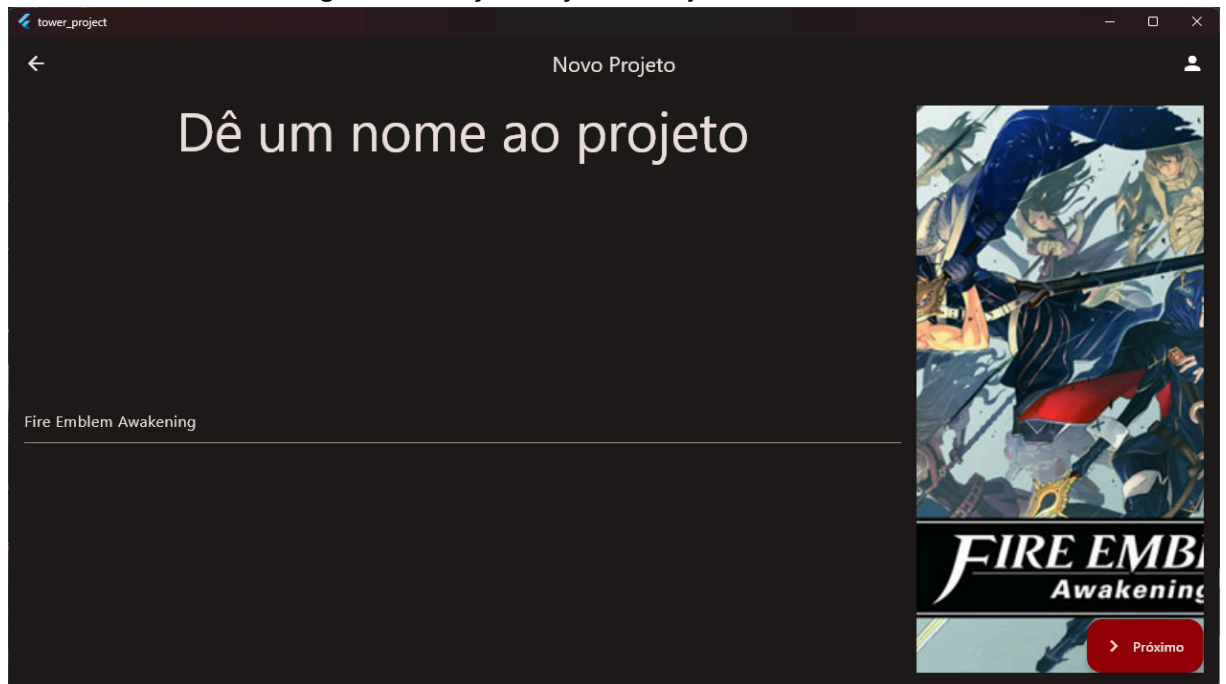
Figura 2 - Tela de Projetos



Fonte: De autoria própria.

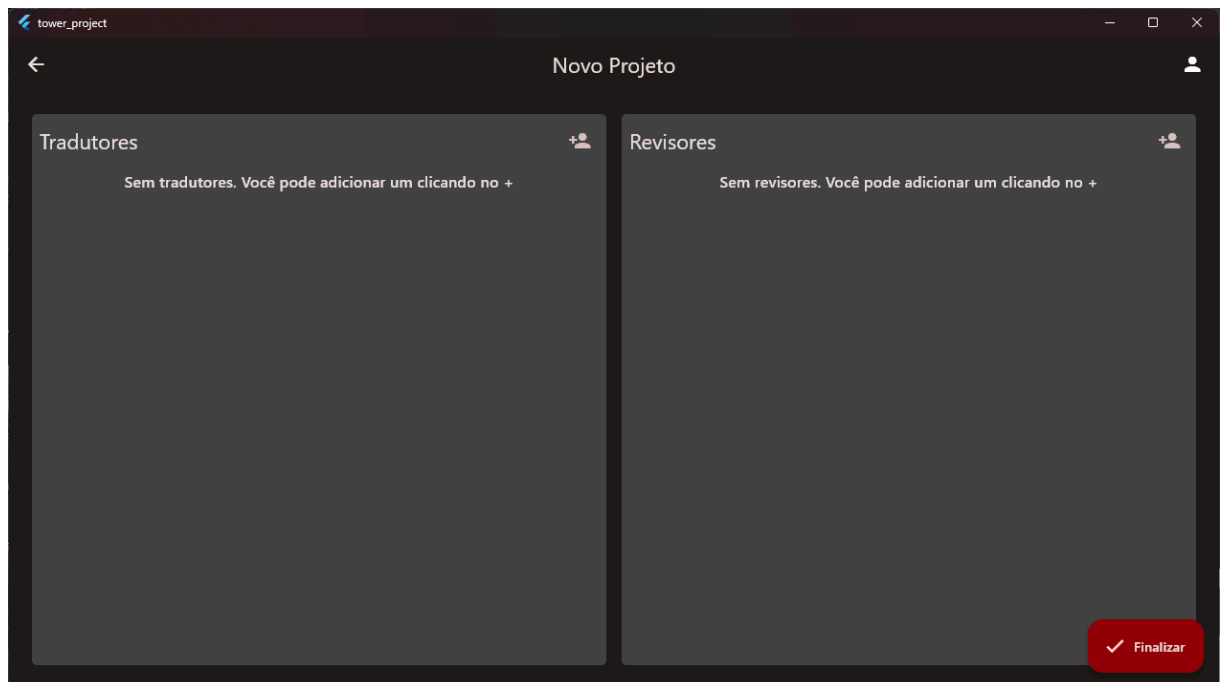
Na tela de Projetos, é possível ver todos os projetos onde o usuário é membro de sua equipe, seja como Gerente, Tradutor ou Revisor. Ele pode clicar em um projeto para ver suas tarefas.

Figura 3 - Criação/Edição de Projeto



Fonte: De autoria própria.

Figura 4 - Gerenciamento de Equipe

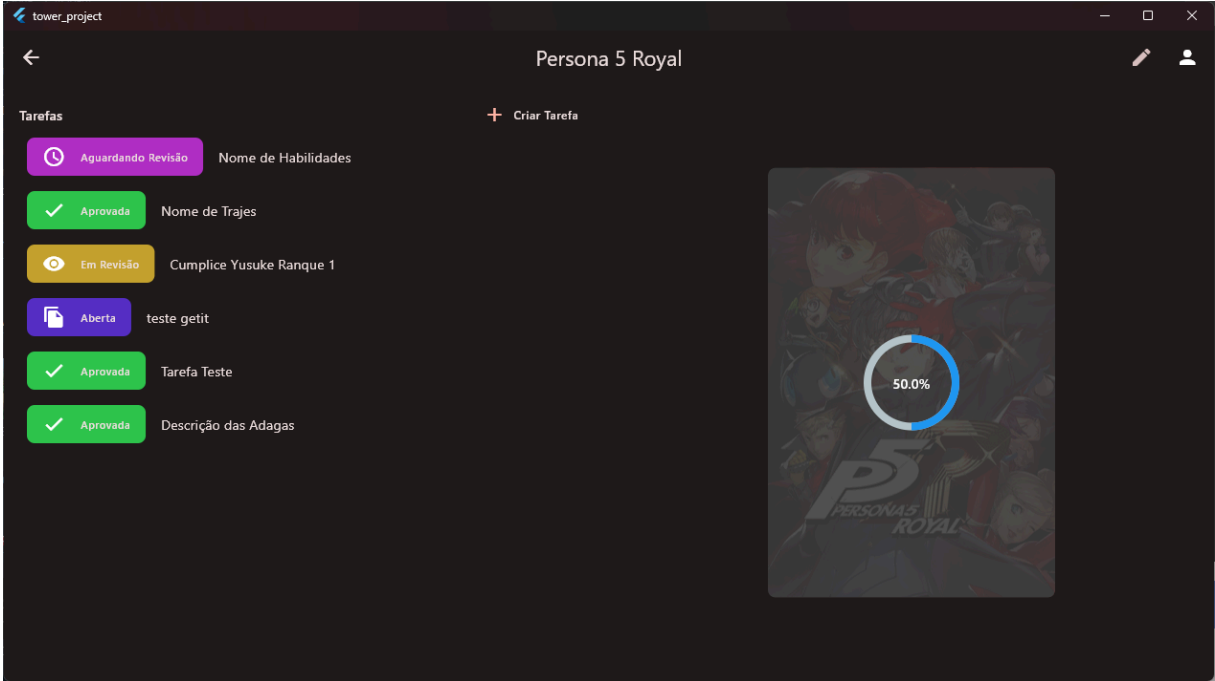


Fonte: De autoria própria.

Para criar um projeto, é necessário um nome e opcionalmente uma imagem para o projeto. Após isso, é exibido a tela de gerenciamento de equipe, onde o usuário pode adicionar membros a um projeto. Em cada projeto é possível cadastrar

os tradutores e revisores. Este gerenciamento pode ser feito posteriormente ao editar um projeto

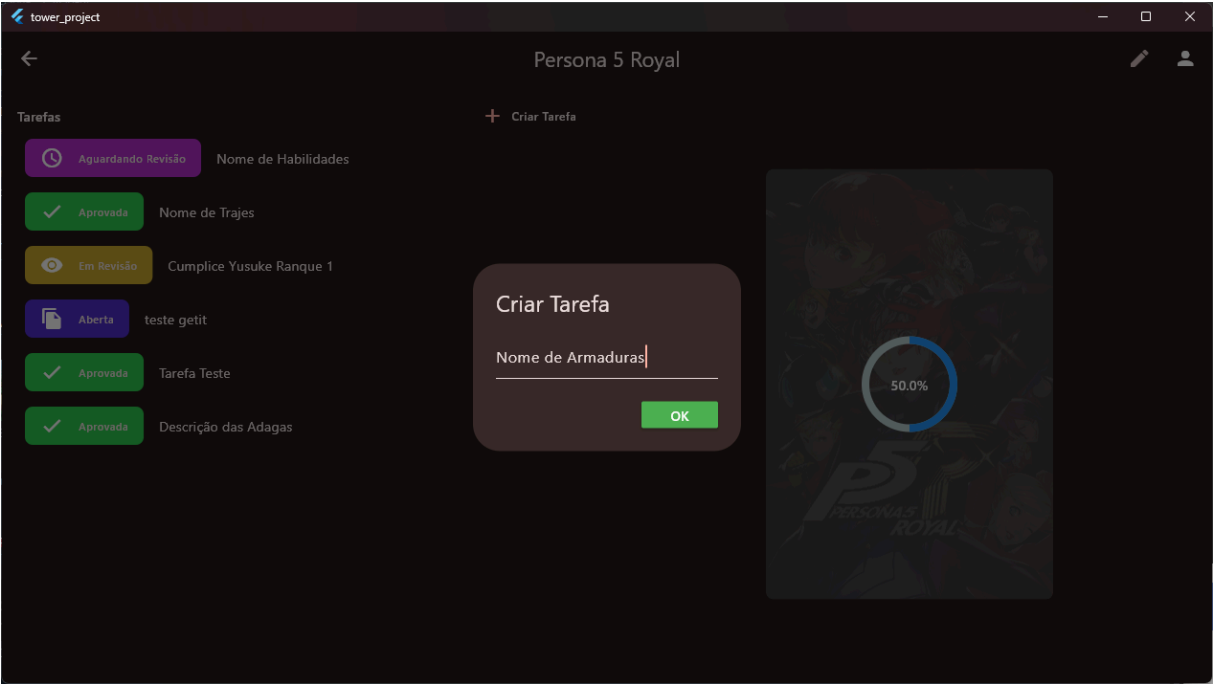
Figura 5 - Visualização de Tarefas



Fonte: De autoria própria.

Em cada projeto, é possível criar tarefas para a tradução. A tela de projeto mostra todas as tarefas e a porcentagem de tarefas completas

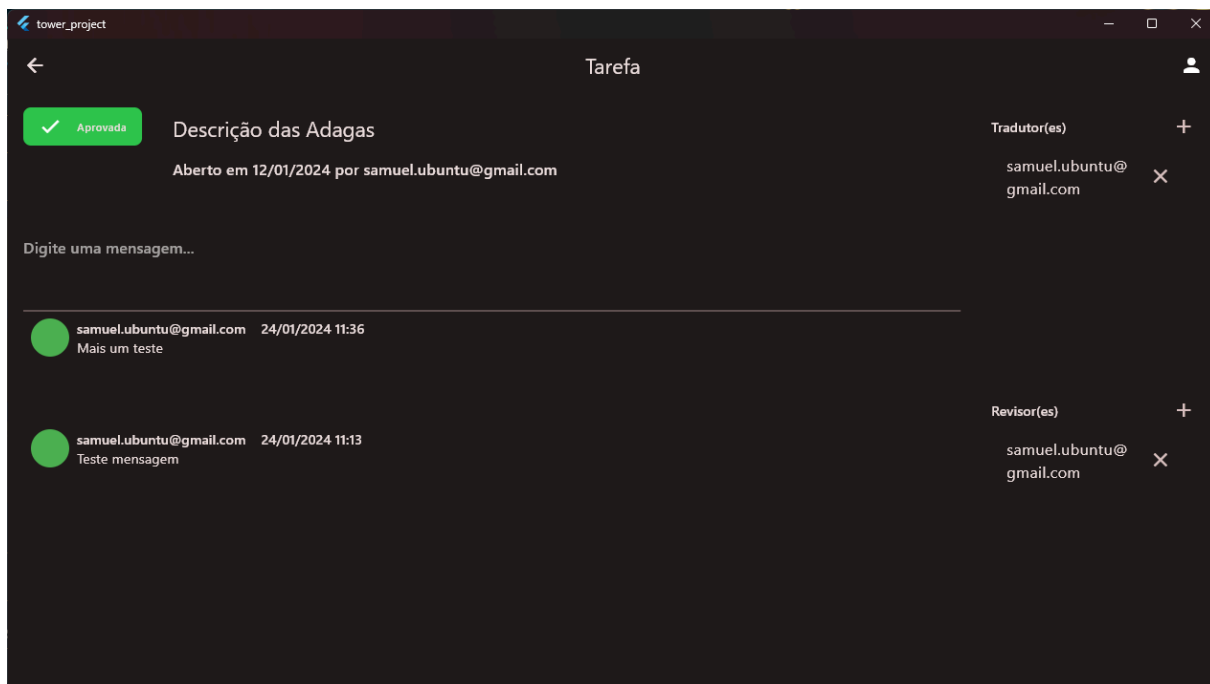
Figura 6 - Criação de Tarefa



Fonte: De autoria própria.

Para criar uma tarefa, é necessário apenas o nome da tarefa. A aplicação criará toda a estrutura da tarefa.

Figura 7 - Detalhes da Tarefa



Fonte: De autoria própria.

Em uma tarefa, é possível ver seu estado atual, os membros assinalados para aquela tarefa e uma interface simples de conversa entre os membros.

4. SOFTWARE

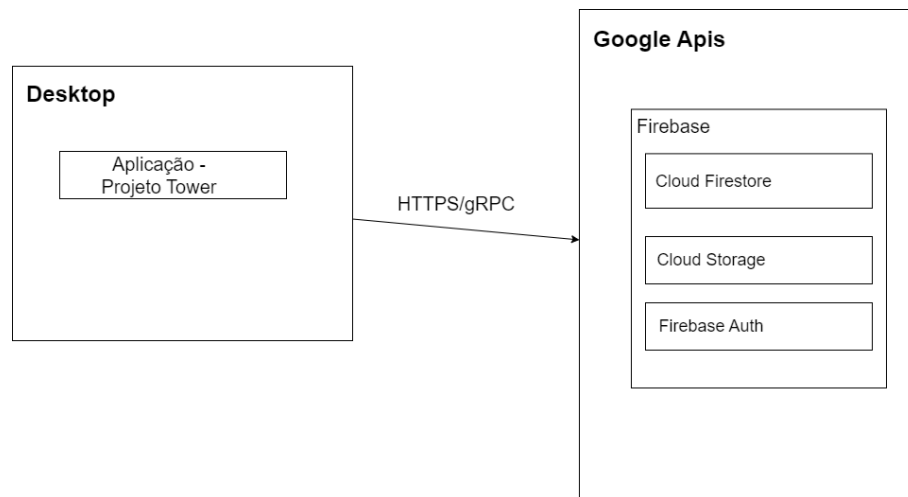
Nesta seção, detalhamos os aspectos cruciais do desenvolvimento e implantação do software. A seguir, apresentamos as estratégias específicas utilizadas na implantação do software e os resultados obtidos nos testes iniciais.

4.1. IMPLANTAÇÃO

Este segmento do projeto aborda a estratégia e os métodos utilizados para a implantação da aplicação. A aplicação Desktop foi implementada utilizando o Cloud Firestore como banco de dados principal, complementado pelo mesmo serviço para

armazenamento de arquivos e pelo Firebase Auth para gerenciamento de usuários e autenticação. Esta abordagem assegura uma implantação eficiente e segura, adaptada às necessidades específicas do projeto.

Figura 7- Diagrama de Implantação



4.2. TESTES

A fase de testes é crucial para avaliar a eficácia e a usabilidade da aplicação. Nesta etapa, a aplicação foi distribuída a seis membros de uma equipe de tradução para uso prático. Após um período de utilização, eles participaram de uma avaliação de usabilidade através do questionário de SUS (System Usability Scale). O SUS é um questionário criado por John Brooke para medir de forma quantitativa a usabilidade de um sistema. Ele consiste de 10 perguntas sobre pontos subjetivos sobre a usabilidade do sistema, como eficácia, eficiência e satisfação. Os resultados preliminares indicaram uma média de 90% de satisfação com a aplicação, sugerindo uma recepção positiva. No entanto, reconhecemos a necessidade de expandir os testes para incluir mais equipes e usuários, a fim de obter uma amostra mais representativa e dados mais abrangentes.

5. CONSIDERAÇÕES FINAIS

Após a implantação da aplicação, foi possível notar uma melhora significativa na eficiência da equipe, onde cada membro tinha facilidade de saber quais tarefas estavam disponíveis e quais estavam assinaladas a ele, e assim realizar seu trabalho mais rapidamente.

REFERÊNCIA

GO GAMERS; SX GROUP. **Pesquisa Game Brasil 10º Edição Gratuita**. São Paulo: GO GAMERS, SX GROUP, 2023

GO GAMERS; SX GROUP. **Pesquisa Game Brasil 9º Edição Gratuita**. São Paulo: GO GAMERS, SX GROUP, 2022

BROOKE, John. **SUS: A quick and dirty usability scale**. Reino Unido. Usability Eval. Ind., 1995.

MICROSOFT CORPORATION. **Visual Studio Code**. 2024. Disponível em: <<https://code.visualstudio.com/>>. Acesso em: 04/03/2024.

GOOGLE LLC. **Flutter**. 2024. Disponível em: <<https://flutter.dev/>>. Acesso em: 04/03/2024.

GOOGLE LLC. **Firestore Authentication**. 2024. Disponível em: <<https://firebase.google.com/docs/auth>> . Acesso em: 04/03/2024.

GOOGLE LLC. **Cloud Firestore**. 2024. Disponível em: <<https://firebase.google.com/docs/firestore>> . Acesso em: 04/03/2024.

GOOGLE LLC. **Cloud Storage**. 2024. Disponível em: <<https://firebase.google.com/docs/storage>> . Acesso em: 04/03/2024.

Architecture | Bloc. **bloclibrary**. 2024. Disponível em: <<https://bloclibrary.dev/architecture/>>. Acesso em: 04/03/2024.

LALANDA, Philippe. **Shared repository pattern**. 1998. França. Thomson-CSF Corporate Research Laboratory, 1998.

BRITISH COUNCIL BRASIL, **Demandas de Aprendizagem de Inglês no Brasil**, São Paulo: British Council Brasil, 2014.

LEAN KABAN UNIVERSITY; ANDERSON, David J.; CARMICHAEL, Andy. **Kanban essencial condensado**. 2016.