



GUILHERME LORINI ANCELLONI

LUCAS SANTANA SATURNINO

LUCAS RAMALHO DA SILVA

TASKMASTER

São Caetano do Sul

2023

GUILHERME LORINI ANCELLONI
LUCAS SANTANA SATURNINO
LUCAS RAMALHO DA SILVA

TASKMASTER

TaskMaster é um aplicativo de gerenciamento de tarefas que visa ajudar os usuários a planejar e acompanhar suas atividades diárias.

Trabalho de Conclusão de Curso apresentado à Faculdade de Tecnologia de São Caetano do Sul, como requisito parcial para a obtenção do grau de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: Prof. Rodrigo Amorim Motta
Carvalho

São Caetano do Sul

2023

DEDICATÓRIA

Aos nossos queridos pais, cujo amor, apoio incondicional e crença em nosso potencial nos guiaram nesta jornada. Este trabalho é dedicado a vocês, em agradecimento por serem nossa inspiração e alicerce para o sucesso.

AGRADECIMENTOS

Gostaríamos de expressar nossa sincera gratidão à Fatec São Caetano do Sul e aos nossos dedicados professores. O apoio, conhecimento e orientação que recebemos foram fundamentais para nosso crescimento acadêmico e pessoal. Obrigado por tornar possível essa conquista.

EPÍGRAFE

*Em planejamento, nunca uma proposição inútil. Em
estratégia, nenhum passo dado em vão."*

Sun Tzu

RESUMO

Este trabalho apresenta um gerenciador de tarefas que tem como objetivo auxiliar os usuários na organização de suas atividades diárias de forma simples e intuitiva. O aplicativo utiliza tecnologias modernas e uma interface amigável para facilitar a priorização, delegação e monitoramento das tarefas, possibilitando uma gestão mais eficiente do tempo e aumento da produtividade.

Uma das funcionalidades interessantes do aplicativo é a gamificação, onde os usuários ganham pontos a cada tarefa concluída. Esses pontos podem ser trocados por recompensas, tornando a experiência ainda mais agradável e incentivando o cumprimento das atividades. Com essa funcionalidade, o gerenciador de tarefas apresentado neste trabalho busca contribuir para o debate acerca da gestão de tempo e organização de tarefas, oferecendo uma solução prática e eficiente para esses desafios.

PALAVRAS CHAVE: Gestão de tempo; Tarefas; Produtividade;

ABSTRACT

This work presents a task manager that aims to assist users in organizing their daily activities in a simple and intuitive way. The application uses modern technologies and a user-friendly interface to facilitate prioritization, delegation, and monitoring of tasks, enabling more efficient time management and increased productivity.

One of the interesting features of the application is gamification, where users earn points for each completed task. These points can be exchanged for rewards, making the experience even more enjoyable and encouraging the completion of activities. With this feature, the task manager presented in this work seeks to contribute to the debate on time management and task organization, offering a practical and efficient solution to these challenges.

Keywords: Time management; Tasks; Productivity

SUMÁRIO

INTRODUÇÃO.....	9
1. REFERENCIAL TEÓRICO.....	11
1.1. Android Studio:.....	11
1.2. Firebase:.....	11
1.3. Parsec:.....	11
2. METODOLOGIA.....	12
2.1. Métricas de teste.....	13
3. DESENVOLVIMENTO.....	14
3.1. Requisitos funcionais.....	14
3.2. Requisitos não funcionais.....	15
3.3. Diagramas.....	16
3.3.1. Diagrama de contexto.....	16
3.3.2. Diagrama de Casos de Uso.....	17
3.3.3. Casos de Uso.....	18
3.3.4.	18
3.3.5. Diagrama de Classe.....	22
3.3.6. Diagrama de Sequência.....	23
3.3.6.3. Login.....	23
3.3.6.4. Tarefas.....	24
3.3.6.5. Recompensas.....	24
3.4. PROTÓTIPO DO APLICATIVO.....	25
3.4.1. Tela de Login:.....	25
3.4.2. Calendário:.....	26
3.4.3. Menu:.....	27
3.4.4. Tarefas e Cadastro de tarefas:.....	28
3.4.5. Recompensas e Cadastro de Recompensas:.....	29
4. RESULTADOS E DISCUSSÕES.....	30
4.1. Métricas De Teste:.....	30
4.1.1. Avaliação do Desempenho:.....	30
4.2. Funcionalidade:.....	30
4.3. Usabilidade:.....	30
4.4. Compatibilidade:.....	31
4.5. Tela Final do Aplicativo.....	32
4.5.1. Tela de Login:.....	32
4.5.2. Calendário:.....	33
4.5.4. Grupos:.....	35
4.5.5. Tarefas e Cadastro das Tarefas:.....	35
4.5.6. Recompensas e Cadastro das recompensas:.....	37
4.6. CÓDIGO.....	39
4.6.1. Resgate de recompensas:.....	39
4.7. Banco de dados.....	40
4.7.1. Grupos:.....	40
4.7.2. Tarefas:.....	41

5. CONSIDERAÇÕES FINAIS.....	42
5.1. REFERÊNCIAS.....	43

INTRODUÇÃO

A gamificação, uma abordagem que incorpora elementos de jogos em situações fora do contexto lúdico, tem se mostrado uma estratégia promissora para melhorar e organizar diversas atividades em nosso dia a dia. Essa abordagem inovadora tem o potencial de motivar e envolver as pessoas de uma maneira única, tornando-as mais propensas a alcançar metas e realizar tarefas de forma eficiente. Lidar com o gerenciamento efetivo do tempo é um desafio constante em um mundo cada vez mais acelerado, onde as demandas pessoais e profissionais competem por nossa atenção. Este estudo tem como objetivo explorar como a gamificação pode ser aplicada no contexto do gerenciamento de tempo para otimizar o desempenho e a satisfação individual.

Segundo Pavlov, I. P. (1953), um fisiologista russo renomado, o condicionamento clássico é um processo que envolve a criação de uma associação entre um estímulo neutro e uma resposta reflexa automática. Este conceito tem sido aplicado na gamificação, uma técnica que incorpora elementos de design de jogos em contextos não relacionados a jogos..

Por exemplo, na gamificação, um elemento de jogo, como pontos ou recompensas, pode ser usado como um estímulo para encorajar um comportamento desejado. Com o tempo, os usuários podem começar a associar esse comportamento com a recompensa, um processo que reflete o condicionamento clássico de Pavlov.

Nos últimos tempos, a gamificação ganhou destaque como uma estratégia para envolver as pessoas. Segundo Deterding et al. (2011), a gamificação refere-se à aplicação de elementos de jogos, como competição, recompensas e desafios, em cenários não relacionados a jogos, com o objetivo de promover a motivação intrínseca. Essa abordagem cria um ambiente onde as pessoas se sentem mais motivadas a participar e realizar tarefas que normalmente poderiam considerar entediantes. A pesquisa de Zichermann e Cunningham (2011) indica que a gamificação pode aumentar tanto a motivação quanto a produtividade, influenciando positivamente o comportamento humano em diversas áreas diferentes.

No mundo acelerado de hoje, dominar o gerenciamento do tempo é crucial. Com a abundância de disrupções tecnológicas e prioridades em constante mudança, é essencial manter-se atualizado. Embora Lakein (1973) e Covey (1989) tenham oferecido técnicas úteis de gerenciamento de tempo tradicional durante anos, os tempos mudaram. Na era digital de hoje, são necessárias abordagens modernas para permanecer à frente. Os dispositivos móveis e as redes sociais criaram novos obstáculos à gestão eficaz do nosso tempo, incluindo distrações e limites pouco claros.

Uma perspectiva intrigante para aumentar a produtividade e melhorar a qualidade de vida surge da combinação de gamificação e gerenciamento de tempo. Ao introduzir elementos de jogo, como pontos, níveis e recompensas, aplicativos e ferramentas como HabitRPG e a “Técnica Pomodoro” (Cirillo, 2006) incentivam os indivíduos a cultivar hábitos saudáveis e a concluir tarefas. Recompensas virtuais e feedback imediato motivam as pessoas a alocar tempo para tarefas importantes e a cumprir seus cronogramas.

Um estudo de caso notável é o aplicativo "Forest," desenvolvido por Shao et al. (2016), que utiliza uma abordagem de gamificação para ajudar os usuários a se concentrarem no trabalho, evitando distrações em seus dispositivos móveis. Quando um usuário ativa o aplicativo para se concentrar, uma árvore virtual começa a crescer, representando seu período de foco. Se o usuário

sair do aplicativo antes do tempo programado, a árvore morre. Esse método incentiva a concentração e o gerenciamento eficaz do tempo.

A gamificação oferece uma abordagem inovadora para otimizar o gerenciamento de tempo, motivando as pessoas a se envolverem em suas tarefas diárias de maneira mais eficaz. À medida que a sociedade continua a evoluir, a capacidade de gerenciar o tempo se torna ainda mais crucial. Este estudo explorou como a gamificação pode ser aplicada com sucesso nesse contexto, apresentando casos de sucesso e abordagens práticas. A integração de elementos de jogos no gerenciamento de tempo oferece a promessa de uma abordagem mais eficaz e motivadora para a otimização pessoal e profissional.

No primeiro capítulo, abordaremos o referencial teórico, focando nos softwares essenciais para o desenvolvimento do aplicativo. O segundo capítulo, metodologia, detalharemos as métricas de teste adotadas. O terceiro capítulo, desenvolvimento, apresentará o processo prático com diagramas e telas protótipos. No quarto capítulo, resultados e discussões, revelaremos os resultados dos testes, exibindo as telas finais, prints do código e insights do banco de dados. Por fim, no capítulo 5, considerações finais, faremos uma análise conclusiva, destacando aprendizados e apontando para futuras melhorias do aplicativo.

1. REFERENCIAL TEÓRICO

1.1. Android Studio:

O Android Studio é a IDE oficial para o desenvolvimento de aplicativos Android. Desenvolvido pelo Google, oferece uma série de ferramentas integradas que facilitam a criação, depuração e emulação de aplicativos. Algumas características notáveis incluem:

- Design Gráfico e Layout: Possui um editor gráfico intuitivo que permite criar interfaces de usuário visualmente atrativas e otimizadas para dispositivos móveis.
- Emulador Integrado: Inclui um emulador Android que permite testar aplicativos em diferentes dispositivos virtualmente, facilitando a detecção de problemas de compatibilidade.
- Suporte a Java e Kotlin: Permite o desenvolvimento de aplicativos Android em Java, uma linguagem amplamente utilizada, e Kotlin, uma linguagem moderna que oferece concisão e segurança.
- Integração com o Firebase: Oferece integração direta com serviços do Firebase, simplificando a implementação de recursos como banco de dados em tempo real, autenticação e armazenamento de arquivos.

1.2. Firebase:

O Firebase é uma plataforma de desenvolvimento oferecida pelo Google, projetada para facilitar o desenvolvimento eficiente de aplicativos móveis e web. Suas principais características incluem:

- Banco de Dados em Tempo Real: Fornece um banco de dados NoSQL em tempo real que permite sincronização instantânea de dados entre dispositivos. Ideal para aplicativos que necessitam de atualizações em tempo real.
- Autenticação: Oferece serviços de autenticação seguros, permitindo que os desenvolvedores integrem facilmente sistemas de login e registro em seus aplicativos.
- Armazenamento: Possui um serviço de armazenamento em nuvem para hospedar e disponibilizar arquivos, como imagens e documentos.

1.3. Parsec:

O Parsec é um software de compartilhamento de tela projetado para proporcionar colaboração eficiente em ambientes de desenvolvimento e jogos. Suas principais características são:

- Compartilhamento de Tela Remoto: Permite que dois ou mais usuários compartilhem a tela de um computador remotamente, possibilitando colaboração em tempo real.
- Baixa Latência: Oferece baixa latência durante o compartilhamento de tela, garantindo uma experiência de colaboração suave e responsiva.
- Controle Compartilhado: Possibilita que usuários remotos controlem o mouse e teclado da máquina hospedeira, facilitando a colaboração em projetos de desenvolvimento.
- Suporte Multiplataforma: Está disponível para Windows, macOS e Linux, garantindo flexibilidade e compatibilidade em diferentes ambientes de trabalho.

2. METODOLOGIA

O desenvolvimento do aplicativo Task Master envolverá a integração eficaz de diversas tecnologias para proporcionar uma experiência mobile otimizada. O Android Studio será a principal ferramenta de desenvolvimento, utilizando as linguagens Java e Kotlin para a construção do aplicativo. O armazenamento e gerenciamento de dados serão implementados por meio do Firebase, um serviço de banco de dados em tempo real fornecido pela Google. Além disso, para colaboração eficiente durante o processo de desenvolvimento, será utilizado o software Parsec, permitindo que duas pessoas compartilhem a máquina simultaneamente.

Android Studio:

O Android Studio é o ambiente de desenvolvimento integrado (IDE) oficial para o desenvolvimento de aplicativos Android. Oferece recursos avançados para design de interface, depuração e emulação. A escolha de utilizar tanto Java quanto Kotlin como linguagens de programação permite uma flexibilidade eficiente durante o desenvolvimento, aproveitando a familiaridade do Java e as vantagens do Kotlin em termos de concisão e segurança.

Firebase:

O Firebase é uma plataforma de desenvolvimento de aplicativos móveis e web oferecida pela Google. No contexto do TaskMaster, será utilizado como serviço de banco de dados em tempo real para armazenar e sincronizar dados entre os dispositivos dos usuários. Além disso, suas funcionalidades de autenticação, armazenamento de arquivos e hospedagem serão exploradas para uma experiência de usuário completa e integrada.

Parsec:

O Parsec é um software de compartilhamento de tela que permite a colaboração simultânea de duas ou mais pessoas na mesma máquina. Essa escolha visa otimizar a eficiência durante o desenvolvimento, permitindo que os colaboradores compartilhem e interajam com o ambiente de desenvolvimento de forma remota e em tempo real.

Hardware Utilizado:

O hardware empregado para o desenvolvimento do aplicativo Task Master consiste em uma máquina equipada com processador Pentium G4560, 8GB de RAM, placa de vídeo GTX 1650, e um armazenamento de 750GB. Essa configuração oferece um ambiente de desenvolvimento robusto, garantindo uma execução eficiente do Android Studio e das demais ferramentas utilizadas.

A integração dessas tecnologias e ferramentas visa criar um aplicativo Task Master eficiente, capaz de oferecer uma gestão intuitiva de tarefas para dispositivos móveis, aliando desempenho e colaboração durante o processo de desenvolvimento.

2.1. Métricas de teste

Foram utilizadas diversas métricas para mensurar o desenvolvimento da aplicação e testar suas funcionalidades, são elas:

Funcionalidade: Verificar se todas as funcionalidades do aplicativo estão funcionando corretamente, sem erros ou bugs.

Usabilidade: Avaliar a facilidade de uso do aplicativo, como a navegação entre as telas, a clareza das informações apresentadas e a facilidade de realizar as tarefas propostas.

Compatibilidade: Testar o aplicativo em diferentes dispositivos móveis, para garantir que ele funcione corretamente em todos os dispositivos Android..

3. DESENVOLVIMENTO

Nesta seção serão apresentados diagramas e alguns trechos de código que fazem parte do desenvolvimento do projeto.

3.1. Requisitos funcionais

Quadro 1 : Requisitos funcionais

ID REQUISITO	DESCRIÇÃO	CASO DE USO RELACIONADO
RF001	O usuário deve ser capaz de criar novos grupos.	CSU009
RF002	O usuário deve ser capaz de criar novas recompensas.	CSU001
RF003	O usuário deve ser capaz de editar ou excluir recompensas existentes.	CSU003, CSU004
RF004	O usuário deve ser capaz de resgatar as recompensas.	CSU002
RF005	O usuário deve ser capaz de criar novas tarefas.	CSU005
RF006	O usuário deve ser capaz de definir a prioridade da tarefa como nenhuma, baixa, média ou alta.	CSU002
RF007	O usuário deve ser capaz de editar ou excluir tarefas existentes.	CSU006, CSU007
RF008	O aplicativo deve fornecer notificações para as tarefas concluídas ou quando o resgate de alguma recompensa é efetuado.	CSU008

Fonte: Elaborado pelo Autor

3.2. Requisitos não funcionais

Quadro 2: Requisitos Não Funcionais

ID REQUISITO	DESCRIÇÃO	TIPO DE REQUISITO
RNF001	O aplicativo deve ser fácil de usar e ter uma interface de usuário intuitiva.	Requisito de Usabilidade
RNF002	O aplicativo deve ser rápido e responsivo, sem atrasos perceptíveis.	Requisito de Desempenho
RNF003	O aplicativo deve ser compatível com dispositivos Android.	Requisito de Interoperabilidade
RNF004	O aplicativo deve ser seguro e proteger as informações do usuário.	Requisito de Segurança
RNF005	O aplicativo deve ser capaz de sincronizar as informações do usuário com o servidor do Firebase.	Requisito de Interoperabilidade

Fonte: Elaborado pelo Autor

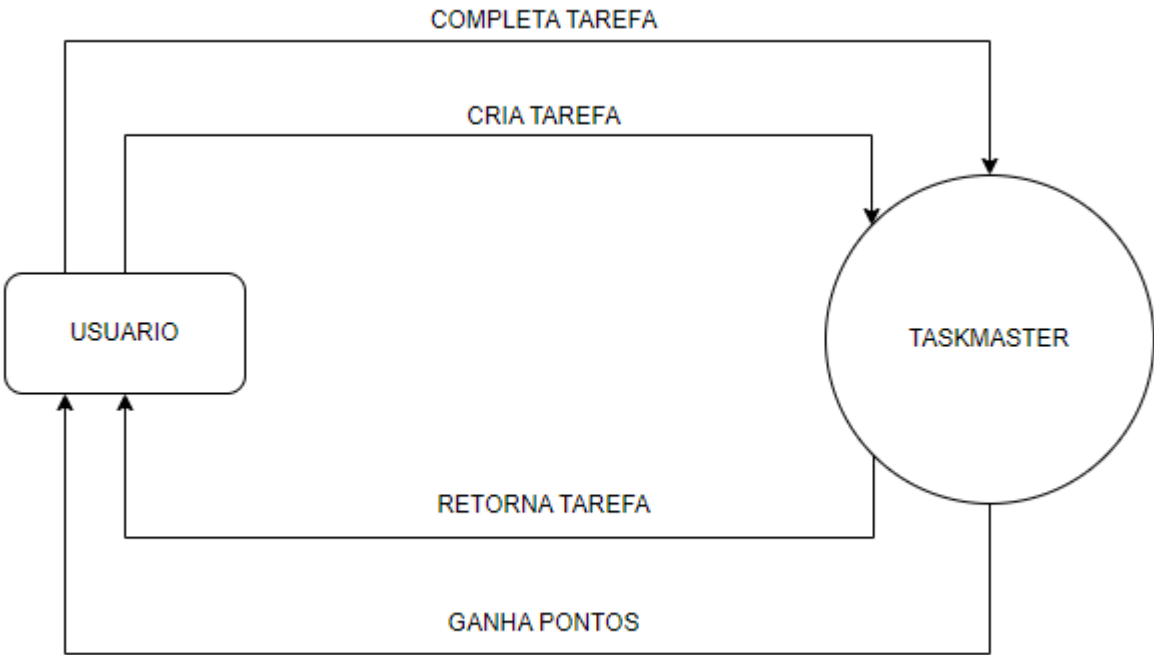
3.3. Diagramas

A seguir serão apresentados os diagramas elaborados para o trabalho.

3.3.1. Diagrama de contexto

A Figura a seguir representa o diagrama de contexto

Figura 1 - Diagrama de Contexto

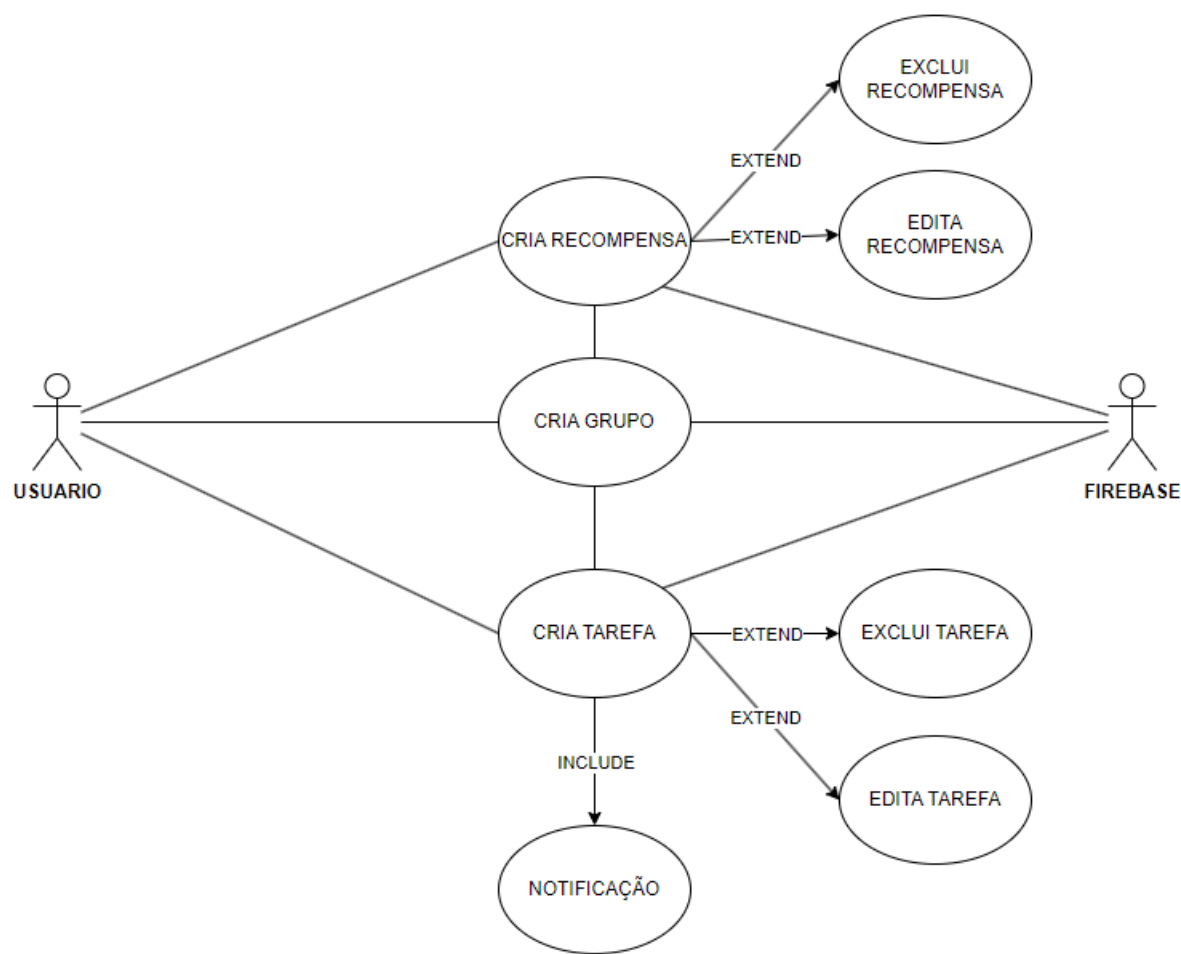


Fonte: Elaborado pelo autor

3.3.2. Diagrama de Casos de Uso

O diagrama a seguir representa o diagrama de Casos de Uso.

Figura 2 - Diagrama de Casos de Uso



Fonte: Elaborado pelo Autor

3.3.3. Casos de Uso

CSU001 - CRIA RECOMPENSA
Descrição: O usuário pode criar uma recompensa na qual ele mesmo pode colocar uma certa pontuação, após criada, o resgate da recompensa só pode ser feito se o usuário tiver a quantidade de pontos necessária.
Pré-Condição: (1) É necessário que o usuário esteja logado no aplicativo.
Fluxo de Eventos: (1) O usuário logado no sistema tem acesso a tela de recompensas (2) Existe um botão para criar a recompensa nesta tela. (3) O usuário insere todas as informações da tarefa como: Título, descrição, pontuação e se necessário o grupo desejado. (4) Após isso, o sistema salva os dados e o usuário volta para a tela de recompensas.
Pós-Condição: (1) A recompensa é criada e exibida na tela de recompensa.
Fluxo Alternativo: Fluxo Alternativo 1: Campos incompletos. (1) No passo 3 do fluxo de eventos o usuário não preenche todos os campos (2) O sistema notifica e retorna ao passo 2 do fluxo de eventos solicitando o preenchimento dos campos. (3) O caso de uso termina.

Fonte: Elaborado pelo Autor

CSU002 - RESGATAR RECOMPENSA
Descrição: Após a conclusão das tarefas o usuário ganha pontos que podem ser utilizados para o resgate de recompensas
Pré-Condição: (1) Deve existir uma recompensa para poder efetuar o resgate da mesma (2) O usuário deverá possuir a quantidade exata ou maior de pontos necessários para resgatar a recompensa desejada.
Fluxo de Eventos: (1) O usuário logado no sistema tem acesso a tela de recompensas. (2) O usuário escolhe a recompensa que deseja resgatar. (3) O sistema faz o resgate da recompensa e debita os pontos do usuário. (4) O caso de uso é encerrado
Pós-Condição: (1) O sistema envia uma notificação para o email do usuário cadastrado com as informações do resgate da recompensa.
Fluxo Alternativo: Fluxo Alternativo 1: Saldo insuficiente. (1) No passo 3 do fluxo de eventos caso o usuário possua menos pontos do que o necessário para o resgate das recompensas. (2) O sistema irá informar ao usuário e retornar ao passo 1 do fluxo de eventos.

Fonte: Elaborado pelo Autor

CSU003 - EXCLUI RECOMPENSA
Descrição: O usuário pode excluir uma recompensa já existente.
Pré-Condição: (1) A tarefa na qual o usuário deseja deletar deve existir no aplicativo ou no grupo.
Fluxo de Eventos: (1) O usuário logado no sistema tem acesso a tela de recompensas. (2) O usuário escolhe a recompensa que deseja excluir. (3) O sistema deleta a recompensa. (4) O caso de uso é encerrado.
Pós-Condição: (1) A recompensa deixa de existir.
Fluxo Alternativo:

Fonte: Elaborado pelo Autor

CSU004 - EDITAR RECOMPENSA
Descrição: O usuário pode alterar uma recompensa já existente.
Pré-Condição: (1) É necessário já existir uma recompensa cadastrada.
Fluxo de Eventos: (1) O usuário logado no sistema tem acesso a tela de recompensa. (2) O usuário escolhe a recompensa que deseja alterar. (3) O sistema preenche a recompensa com os dados a serem alterados (4) O caso de uso é encerrado.
Pós-Condição: (1) A recompensa é atualizada.
Fluxo Alternativo:

Fonte: Elaborado pelo Autor

CSU005 - CRIAR TAREFA
Descrição: O usuário pode criar uma tarefa na qual ele escolhe o nível de prioridade desta tarefa, título, data de término e o descrição desta tarefa.
Pré-Condição: (1) O usuário deve estar logado no aplicativo.
Fluxo de Eventos: (1) O usuário logado no sistema tem acesso a tela de tarefas. (2) Existe um botão para criar tarefa nesta tela. (3) O usuário insere todas as informações da tarefa como: Nível de prioridade, título, descrição e data de término. (4) Após isso, o sistema salva os dados e o usuário volta para a tela de tarefas.
Pós-Condição: (1) A tarefa é criada e exibida na tela de tarefas.
Fluxo Alternativo: Fluxo Alternativo 1: Campos incompletos. (1) No passo 3 do fluxo de eventos o usuário não preenche todos os campos (2) O sistema notifica e retorna ao passo 2 do fluxo de eventos solicitando o preenchimento dos campos. (3) O caso de uso termina.

Fonte: Elaborado pelo Autor

CSU006 - EXCLUIR TAREFA
Descrição: O usuário pode excluir uma tarefa já existente.
Pré-Condição: (1) É necessário já existir uma tarefa cadastrada.
Fluxo de Eventos: (1) O usuário logado no sistema tem acesso a tela de tarefas. (2) O usuário escolhe a tarefa que deseja excluir. (3) O sistema deleta a tarefa. (4) O caso de uso é encerrado.
Pós-Condição: (1) A tarefa deixa de existir.
Fluxo Alternativo:

Fonte: Elaborado pelo Autor

CSU007 - EDITAR TAREFA
Descrição: O usuário pode alterar uma tarefa já existente.
Pré-Condição: (1) É necessário já existir uma tarefa cadastrada.
Fluxo de Eventos: (1) O usuário logado no sistema tem acesso a tela de tarefas. (2) O usuário escolhe a tarefa que deseja alterar. (3) O usuário preenche a tarefa com os dados a serem alterados (4) O caso de uso é encerrado.
Pós-Condição: (1) A tarefa é atualizada.
Fluxo Alternativo:

Fonte: Elaborado pelo Autor

CSU008 - NOTIFICAÇÃO
Descrição: Assim que a tarefa for se aproximando da data limite é emitida uma notificação para o usuário.
Pré-Condição: (1) É necessário uma tarefa já cadastrada. (2) A tarefa deve estar próxima ou dentro da data de término.
Fluxo de Eventos: (1) O sistema identifica a necessidade de notificar o usuário. (2) O sistema coleta as informações necessárias como a data final, título e descrição da tarefa. (3) O sistema envia a notificação para o usuário. (4) O caso de uso é encerrado.
Pós-Condição: (1) A notificação é enviada com sucesso ao usuário.
Fluxo Alternativo:

Fonte: Elaborado pelo Autor

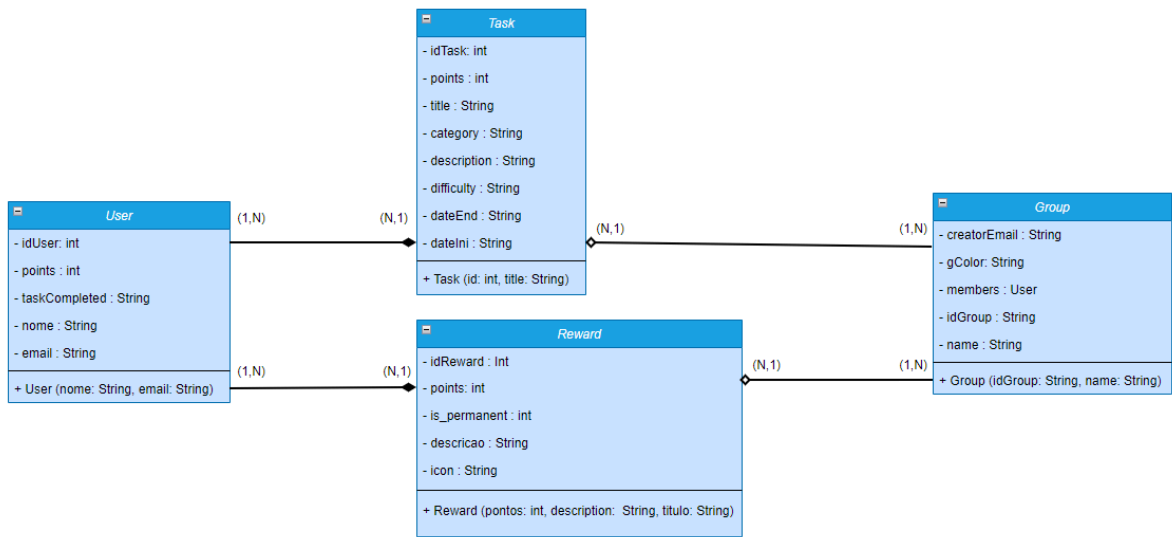
CSU009 - CRIAR GRUPO
Descrição: É possível através do grupo colocar outras pessoas e atribuir tarefas e recompensas para todos os indivíduos presentes naquele grupo.
Pré-Condição: (1) É necessário que todos os integrantes do grupo, inclusive o que criou o grupo, estejam previamente cadastrados no aplicativo.
Fluxo de Eventos: (1) O usuário cria o grupo. (2) O usuário adiciona as pessoas desejadas naquele grupo. (3) O usuário atribui tarefas e recompensas para aquele grupo.
Pós-Condição: (1) Será possível resgatar as recompensas do grupo, e caso você seja o administrador, editar ou excluir as tarefas e recompensas.
Fluxo Alternativo:

Fonte: Elaborado pelo Autor

3.3.4. Diagrama de Classe

O diagrama a seguir representa o diagrama de Classe.

Figura 3 - Diagrama de Classe



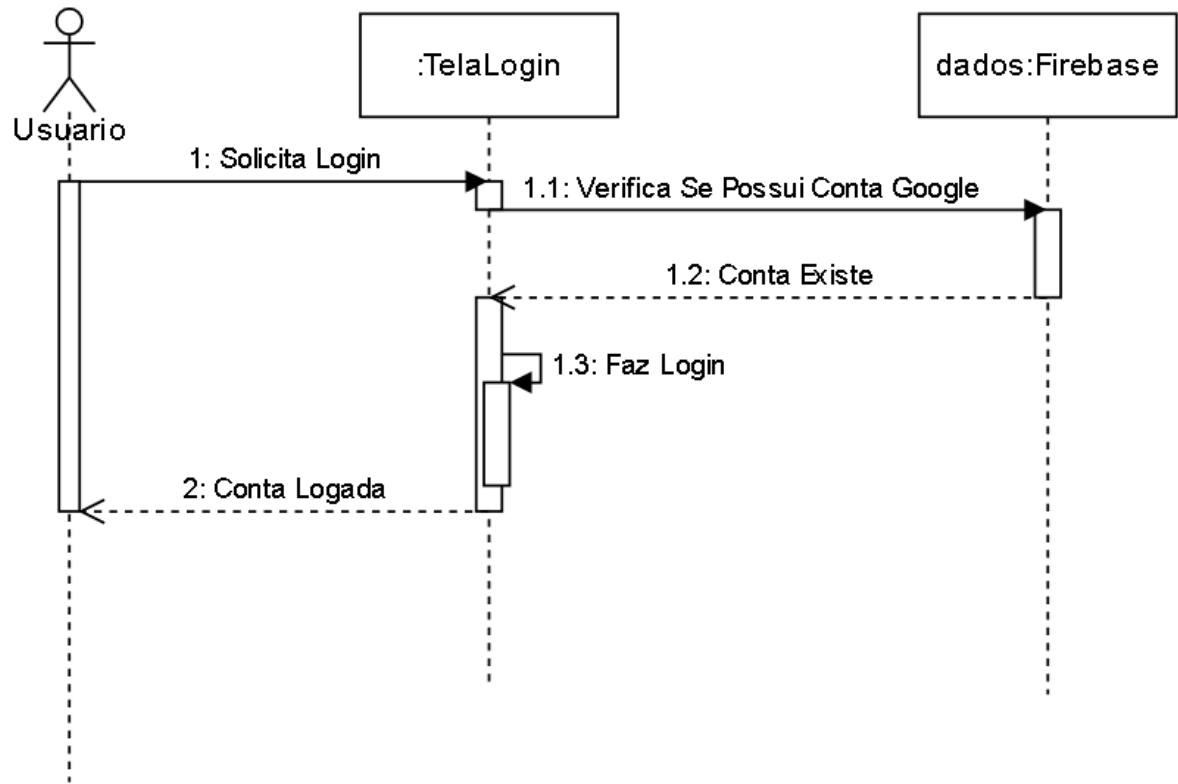
Fonte: Elaborado pelo Autor

3.3.5. Diagrama de Sequência

O diagrama a seguir representa o diagrama de Sequência.

3.3.5.3. Login

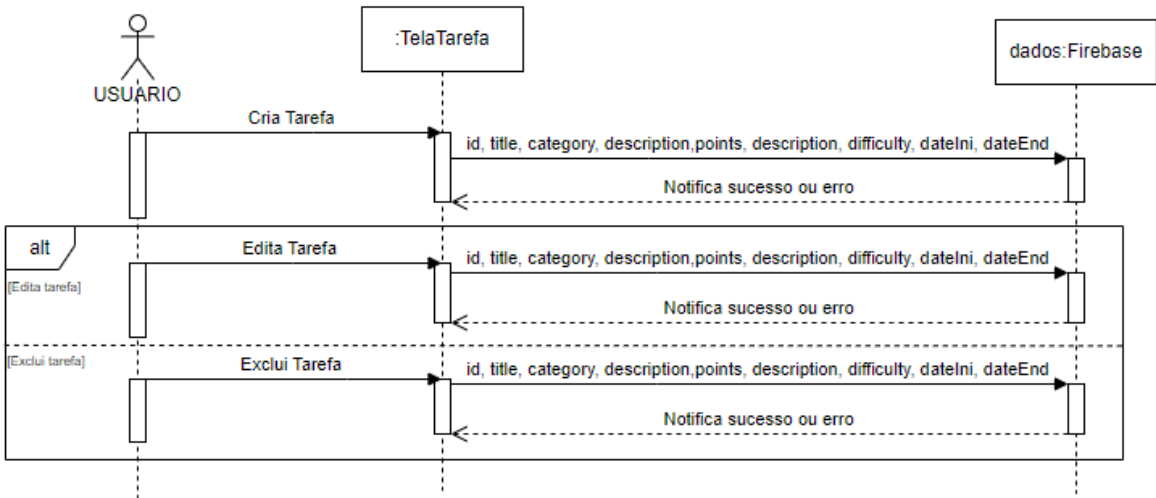
Figura 4 - Diagrama de Sequência - Login



Fonte: Elaborada pelo Autor

3.3.5.4. Tarefas

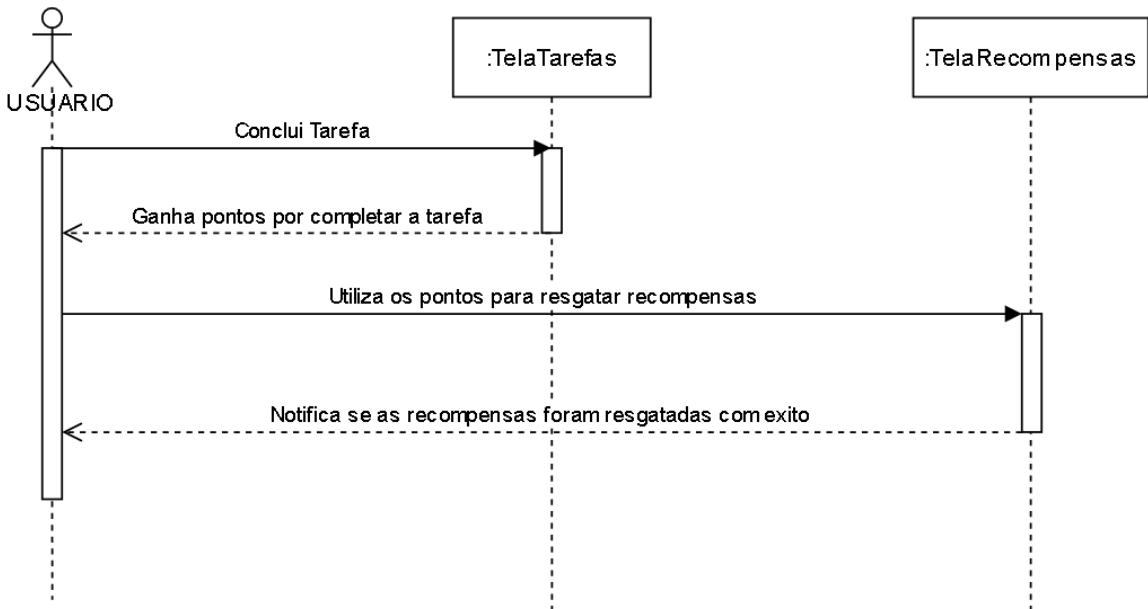
Figura 5 - Diagrama de Sequência - Tarefas



Fonte: Elaborado pelo Autor

3.3.5.5. Recompensas

Figura 6 - Diagrama de Sequência - Recompensas



Fonte: Elaborado pelo Autor

3.4. PROTÓTIPO DO APLICATIVO

3.4.1. Tela de Login:

Figura 7 - Tela de Login



Na tela de login, o usuário deverá possuir uma conta google para acessar o aplicativo.

Fonte: Elaborado pelo Autor

3.4.2. Calendário:

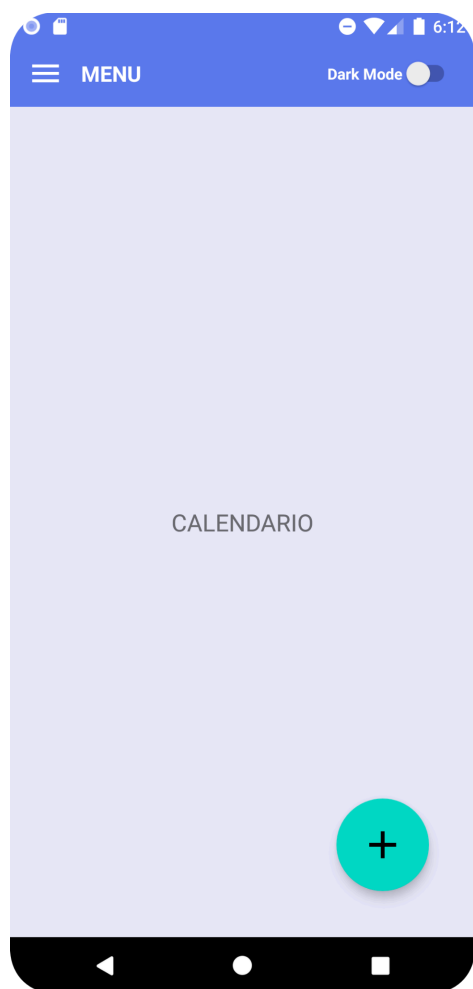


Figura 8 - Tela de calendário

Na tela de calendário, será possível visualizar as suas tarefas nos dias de início e término, assim, deixando mais intuitiva a visualização do fim da atividade.

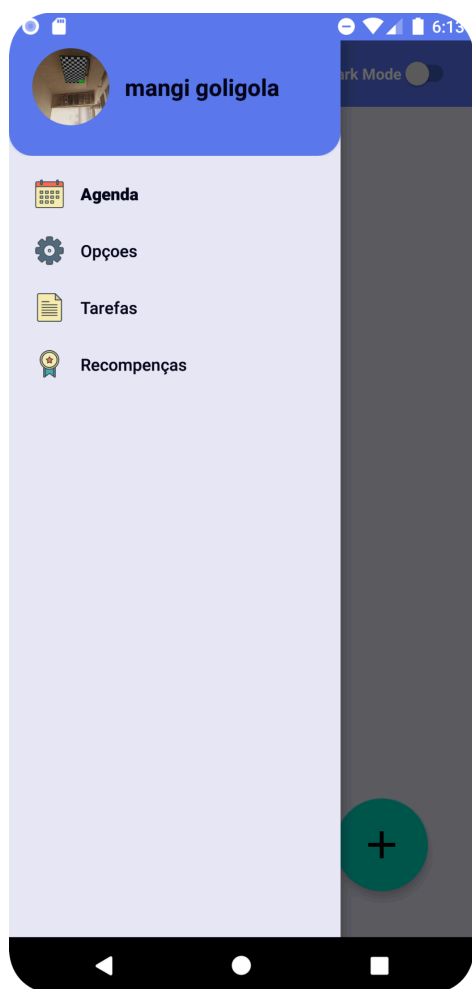
Fonte: Elaborado pelo Autor

3.4.3. Menu:

Figura 9 - Tela de menu

Através do menu é possível navegar entre as principais telas do aplicativo que são:

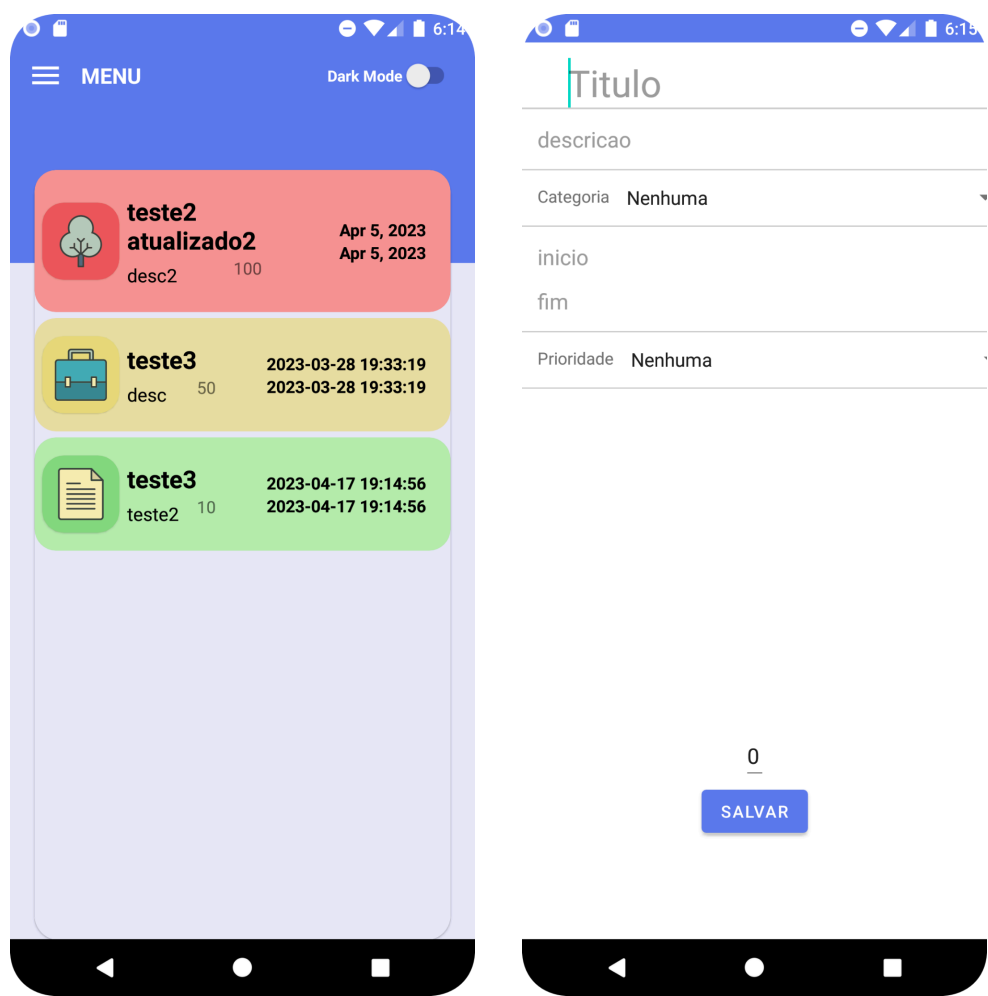
- Agenda
- Tarefas
- Recompensas



Fonte: Elaborado pelo Autor

3.4.4. Tarefas e Cadastro de tarefas:

Figura 10 - Tela de tarefas e cadastro de tarefas

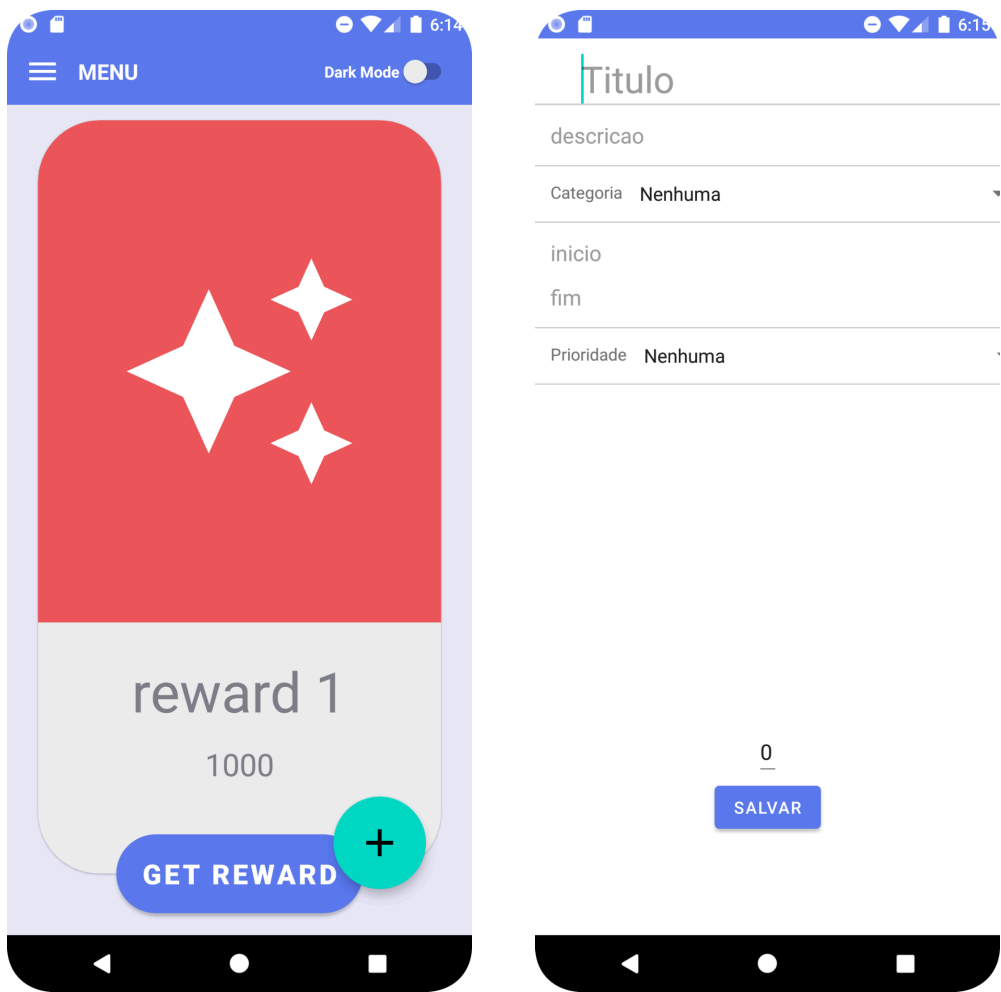


Fonte : Elaborado pelo Autor

Na imagem à esquerda, é exibido todas as tarefas criadas, incluindo informações como nome, descrição, data de início e término, pontuação quando completada, e nível de prioridade. Na imagem à direita, é possível ver a interface de cadastro de novas tarefas.

3.4.5. Recompensas e Cadastro de Recompensas:

Figura 11 - Tela de recompensas e Cadastro de Recompensas



Fonte: Elaborado pelo Autor

Na imagem à esquerda, são exibidas todas as recompensas, incluindo informações como nome e pontuação necessária para resgate. Na imagem à direita, é possível ver a interface de cadastro de novas recompensas.

4. RESULTADOS E DISCUSSÕES

Requisitos funcionais e não funcionais são tipos de requisitos usados na engenharia de software para descrever as necessidades e expectativas do sistema.

4.1. Métricas De Teste:

Ao realizar uma avaliação abrangente do desempenho do aplicativo, concentramos nossos esforços na análise crítica do tempo de carregamento e na resposta a ações do usuário. Essas métricas são indicativas diretas da experiência do usuário e, portanto, são cruciais para garantir a satisfação e a eficiência operacional do aplicativo.

4.2. Usabilidade:

Ao direcionar nosso foco para a usabilidade do aplicativo, buscamos a facilidade com que os usuários podem navegar entre as telas, a clareza das informações apresentadas e a simplicidade na execução das tarefas propostas. Esses aspectos são vitais para garantir que a experiência do usuário seja intuitiva e eficaz.

4.3. Compatibilidade:

A avaliação da usabilidade do aplicativo em diversos dispositivos móveis é uma etapa crucial para garantir uma experiência consistente. Este conjunto de métricas destaca os resultados obtidos ao testar a usabilidade em diferentes dispositivos, assegurando que o aplicativo funcione corretamente em variadas configurações.

4.3.1. Navegação Confiável em Diferentes Tamanhos de Tela:

O aplicativo demonstrou uma navegação eficiente, ajustando-se adequadamente a telas menores, como smartphones compactos, e otimizando o espaço em dispositivos de tela maior, como tablets.

4.3.2. Desempenho em Condições de Rede Variadas:

Em condições de conectividade limitada, o aplicativo manteve funcionalidades essenciais, destacando sua resiliência. Adicionalmente, adaptou-se eficazmente a diferentes velocidades de conexão, garantindo uma experiência consistente.

4.4. Tela Final do Aplicativo

4.4.1. Tela de Login:

Figura 12 - Tela de Login



Não houveram mudanças da tela de login protótipo para a tela de login do APP final.

Fonte: Elaborado pelo Autor

4.4.2. Calendário:

Figura 13 - Calendário



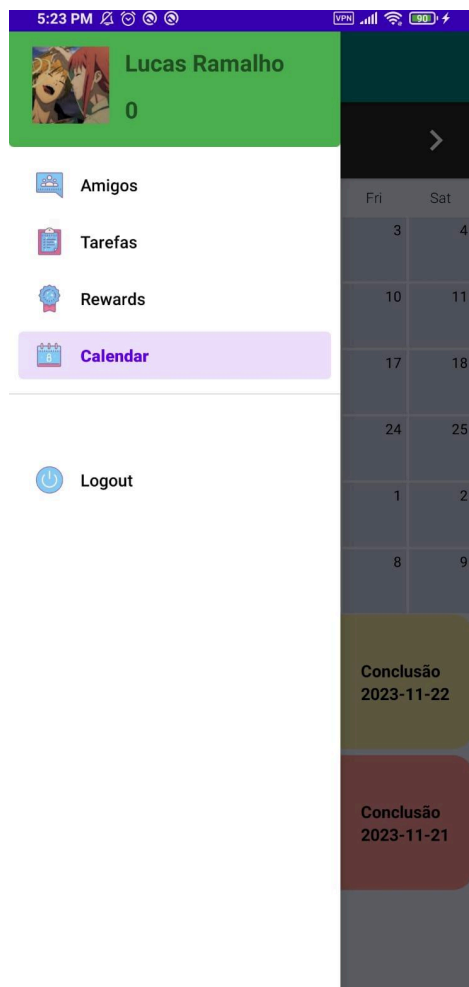
Na tela de calendário, agora é possível visualizar as suas tarefas nos dias de término com as respectivas cores de relevância, assim, deixando mais fácil a visualização do fim da atividade.

Também é possível ver as respectivas atividades daquele mês logo abaixo do calendário.

Fonte: Elaborado pelo Autor

4.4.3. Menu:

Figura 14 - Menu



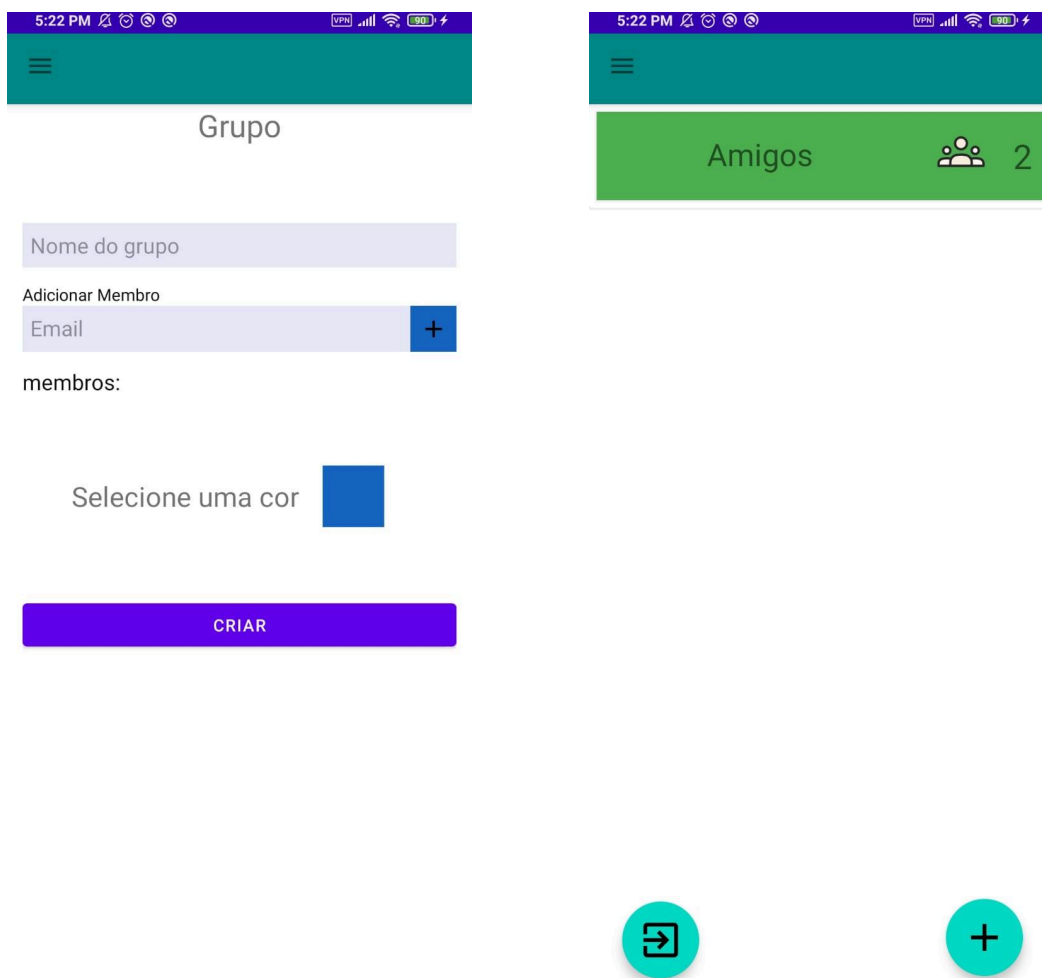
Através do menu é possível navegar entre as principais telas do aplicativo que são:

- Amigos
- Agenda
- Tarefas
- Recompensas
- LogOut

E foram adicionadas agora ao menu a pontuação que o usuário possui para o resgate das recompensas e um botão para Logout.

4.4.4. Grupos:

Figura 15- Grupos



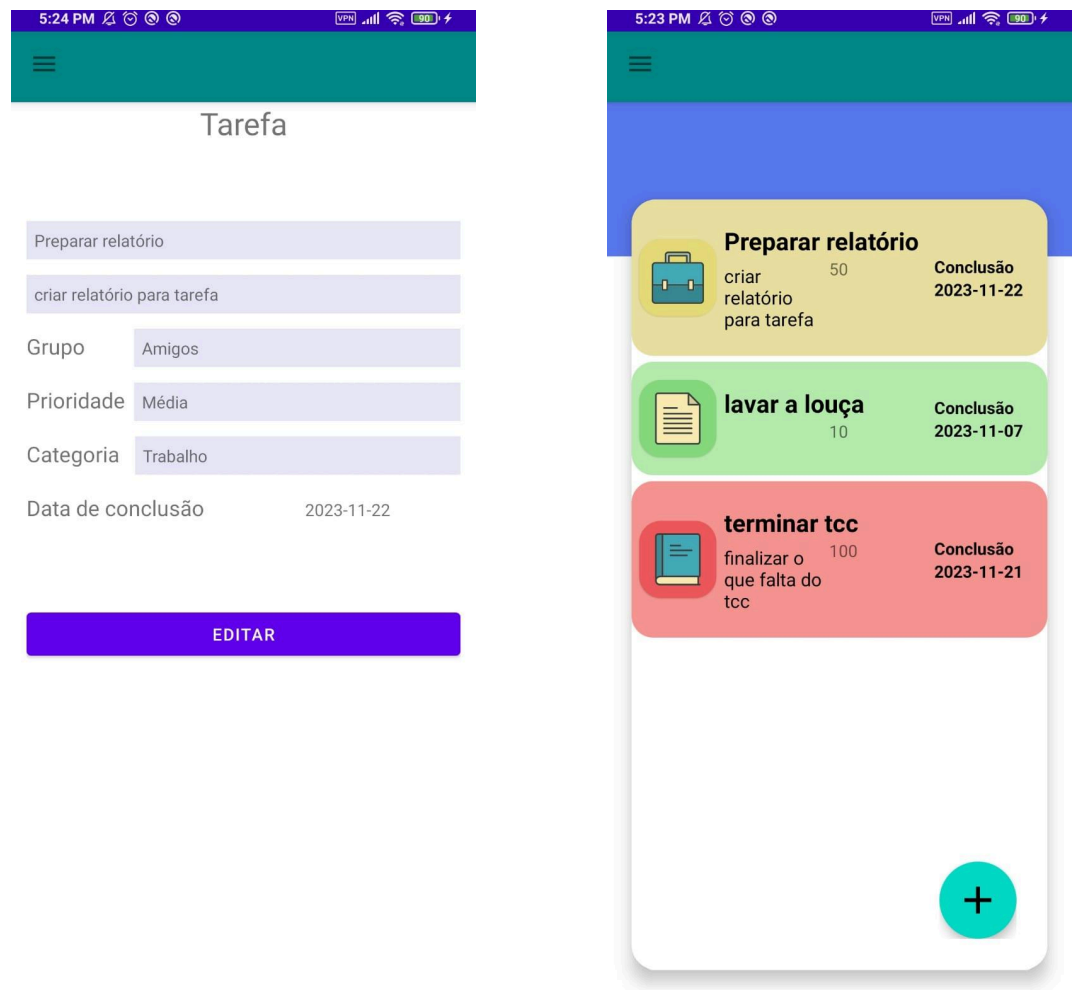
Fonte: Elaborado pelo Autor

Na imagem à Direita, são exibidos todos os grupos já criados, incluindo informações como nome, e a quantidade de pessoas presentes nesses grupos. Na imagem à esquerda, é possível ver a interface de cadastro de novos grupos.

É importante mencionar que é possível fazer parte de vários grupos, e as tarefas de cada grupo só podem ser criadas pelo fundador do grupo.

4.4.5. Tarefas e Cadastro das Tarefas:

Figura 16 - Tarefas e Cadastro de Tarefas



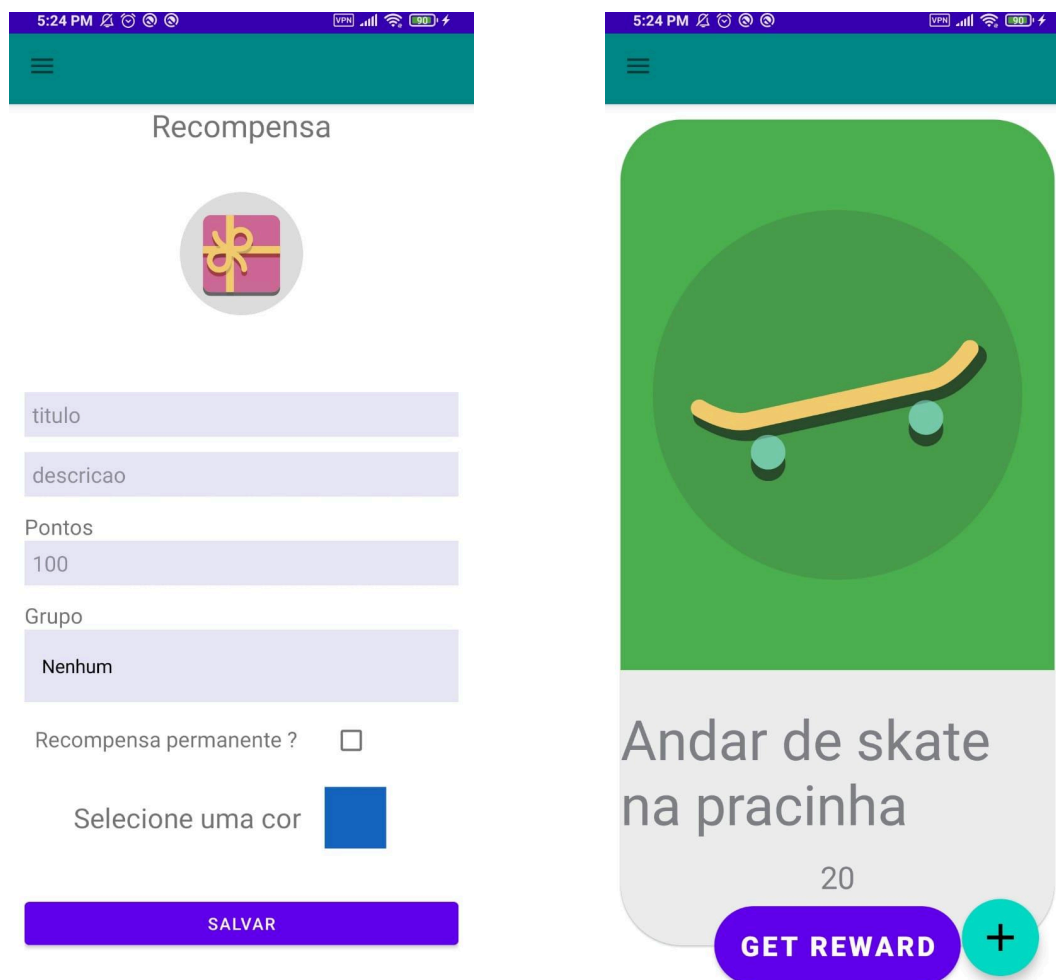
Fonte: Elaborado pelo Autor

Na imagem à Direita, é exibido todas as tarefas criadas, incluindo informações como nome, descrição, data de início e término, pontuação quando completada, e nível de prioridade. Na imagem à direita, é possível ver a interface de cadastro de novas tarefas.

As tarefas que aparecem à direita são correspondentes ao grupo selecionado.

4.4.6. Recompensas e Cadastro das recompensas:

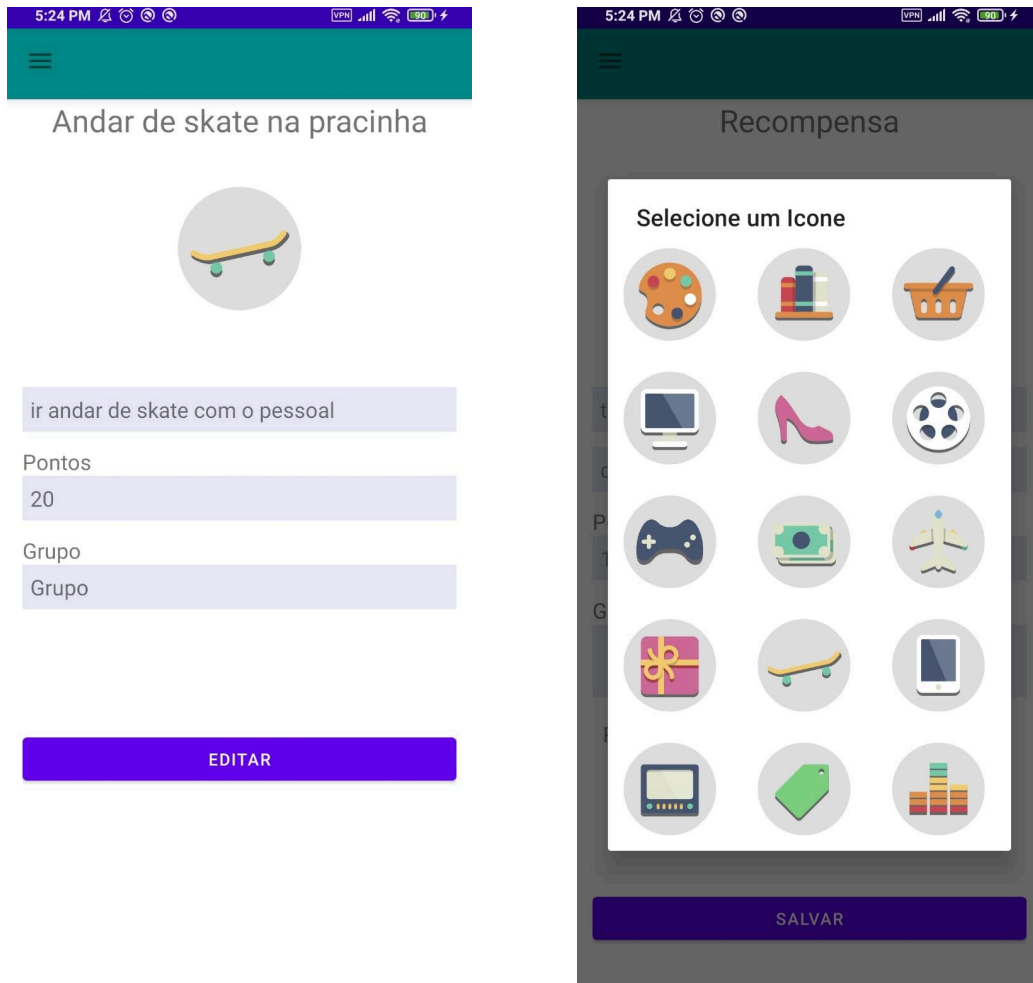
Figura 17 - Recompensa e Cadastro das Recompensas



Fonte: Elaborado pelo Autor

Na imagem à Direita, são exibidas todas as recompensas em um sistema de carrossel, incluindo informações como nome e pontuação necessária para resgate. Na imagem à esquerda, é possível ver a interface de cadastro de novas recompensas.

Figura 18 - Edição das Recompensas



Fonte: Elaborado pelo Autor

Na imagem à esquerda, é possível ver a interface para edição de recompensas já criadas, a direita são os ícones disponíveis para personalização dessas recompensas.

4.5. CÓDIGO

4.5.1. Resgate de recompensas:

Figura 19 -Resgate de Recompensas

```
124 @
125 void confirmReward(Reward reward, User user, int position){
126
127     AlertDialog.Builder builder = new AlertDialog.Builder(context);
128     builder.setTitle("Resgatar recompensa ?");
129     builder.setMessage("Tem certeza que deseja adquirir a recompensa " + reward.getTitle() + " ?");
130     builder.setPositiveButton("Sim", (dialogInterface, i) -> {
131
132         dbRefReward = FirebaseDatabase.getInstance().getReference( path: "rewards").child(reward.getId());
133
134         Map<String, Integer> uPoints = new HashMap<>();
135         uPoints = user.getPoints();
136         Log.d( tag: "USER POINTS", msg: "points =" + uPoints);
137
138         if(reward.getGroupID() != null){
139             uPoints.put(reward.getGroupID(), <v> uPoints.get(reward.getGroupID()) - reward.getPoints());
140         } else {
141             uPoints.put( <k> "default", <v> uPoints.get("default") - reward.getPoints());
142         }
143
144         dbRefUser.child(sPref.getString( <e> "uID", <s> null)).child( pathString: "points").setValue(uPoints);
145
146         if(reward.getIs_permanent() == 0) {
147             dbRefReward.removeValue();
148             this.notifyItemRemoved(position);
149         }
150
151         ((MainActivity) context).sendEmail(reward.getEmail(), user.getEmail(), reward.getTitle(), type: 1, <date> false);
152     });
153     builder.setNegativeButton( text: "Não", (dialogInterface, i) -> {
154
155     });
156     builder.create().show();
157 }
158 }
```

Fonte: Elaborado pelo Autor

A imagem acima representa o trecho de código do resgate de recompensas. Este trecho abrange um fluxo essencial no processo de resgate de recompensas. Ele inicia exibindo um pop-up de confirmação para garantir que o usuário deseja realmente resgatar a recompensa selecionada. Em seguida, verifica se a recompensa faz parte de algum grupo e, caso positivo, confirma se o usuário possui pontos suficientes no grupo para resgatá-la. Se a recompensa não pertencer a um grupo, ela é considerada pessoal, e o código verifica se o usuário possui pontos pessoais suficientes para o resgate.

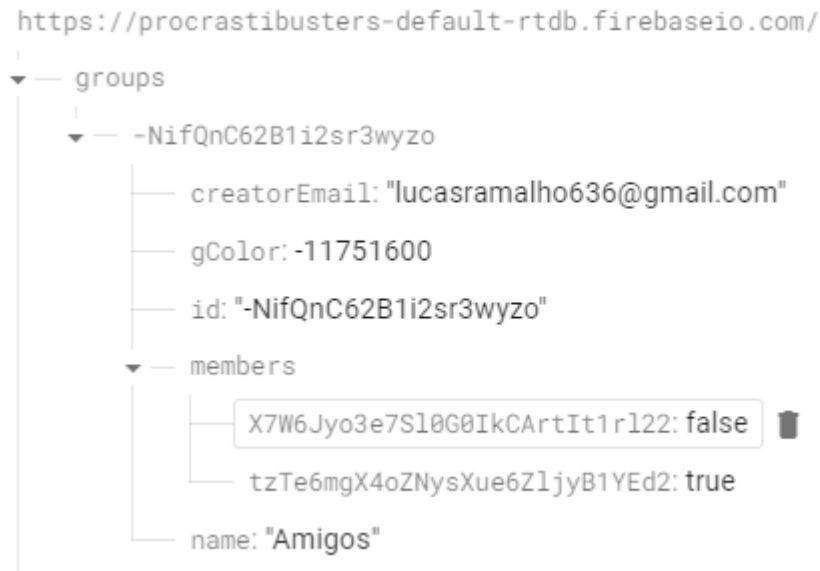
Após a confirmação dos pontos necessários, o código procede ao resgate da recompensa, considerando se a recompensa é permanente ou não. Se permanente, a recompensa não é removida do banco de dados, caso contrário, ela é excluída, e a lista de recompensas do usuário é atualizada.

Esse trecho destaca a lógica intrincada por trás do processo de resgate de recompensas, garantindo que todos os aspectos, desde a disponibilidade de pontos até a permanência da recompensa, sejam gerenciados de maneira eficaz no aplicativo.

4.6. Banco de dados

4.6.1. Grupos:

Figura 20 - Grupos DB



Fonte: Elaborado pelo Autor

A tabela "Grupos" no banco de dados do TaskMaster é crucial para promover interação e colaboração entre usuários. Ela permite que os usuários se agrupam com base em interesses comuns, compartilhando tarefas e recompensas. Este trecho de código específico estabelece critérios claros para resgatar recompensas para cada indivíduo dentro do grupo. Em resumo, essa tabela não apenas organiza usuários, mas cria uma atmosfera colaborativa que enriquece a experiência do TaskMaster.

4.6.2. Tarefas:

Figura 21 - Tarefas DB



Fonte: Elaborado pelo Autor

A importância desta tabela reside na sua capacidade de organizar e registrar as tarefas dos usuários, proporcionando uma visão clara e acessível de suas responsabilidades. Ao armazenar dados cruciais, como prazos e estados de conclusão, a tabela "Tarefas" viabiliza a funcionalidade principal do aplicativo: o gerenciamento eficiente do dia a dia.

Além disso, a tabela é projetada para suportar operações comuns de gerenciamento de tarefas, como adição, atualização e remoção de tarefas. Essa estrutura garante que a interface do usuário permaneça sincronizada com as atividades em andamento, oferecendo uma experiência de gerenciamento de tarefas mais fluida.

Em resumo, a tabela "Tarefas" é a espinha dorsal do sistema, proporcionando a base estrutural para que os usuários possam visualizar, organizar e completar suas tarefas de maneira eficiente dentro do TaskMaster.

5. CONSIDERAÇÕES FINAIS

Ao longo deste trabalho, exploramos os desafios enfrentados pelos usuários na gestão de suas responsabilidades diárias e, conseqüentemente, a importância de ferramentas eficientes para lidar com essas demandas. O aplicativo desenvolvido atendeu com sucesso a essas necessidades, proporcionando uma interface intuitiva, recursos robustos e uma experiência de usuário aprimorada.

Em conclusão, o desenvolvimento e implementação bem-sucedidos do aplicativo gerenciador de tarefas proporcionaram uma notável otimização no fluxo de trabalho do usuário. A capacidade de organizar e monitorar tarefas de maneira eficiente contribui significativamente para o aumento da produtividade e a gestão eficaz do tempo.

Como sugestão para o próximo passo deste projeto, recomenda-se a implementação de uma funcionalidade de geolocalização, integrando o Google Maps ao aplicativo. Esta adição permitiria que os usuários associassem um local específico a cada tarefa cadastrada, fornecendo estimativas precisas de tempo de deslocamento. Ao cadastrar uma tarefa e indicar um endereço, o aplicativo poderia calcular o tempo necessário para o usuário se deslocar até o local da tarefa, oferecendo uma visão ainda mais completa e prática do gerenciamento de suas atividades diárias.

Essa integração não apenas melhoraria a eficiência na execução de tarefas, mas também proporcionaria uma experiência mais personalizada e adaptada ao estilo de vida do usuário, aumentando assim a utilidade e a relevância do aplicativo no contexto da vida cotidiana.

5.1. REFERÊNCIAS

- Cirillo, F. (2006). **The Pomodoro Technique**. Cirillo Consulting. Disponível em: <http://friend.ucsd.edu/reasonableexpectations/downloads/Cirillo%20--%20Pomodoro%20Technique.pdf>. Acesso em: 05 Fev. 2023.
- Covey, S. R. (1989). **The 7 Habits of Highly Effective People**. Free Press. Available at: https://en.wikipedia.org/wiki/The_7_Habits_of_Highly_Effective_People. Acesso em: 05 Fev. 2023.
- Deterding, S., Dixon, D., Khaled, R., & Nacke, L. (2011). **From Game Design Elements to Gamefulness: Defining "Gamification"**. In Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments (pp. 9-15). Available at: <https://pdfs.semanticscholar.org/1768/99ac1f3d4ab0b836b5f62eae511cb79a09b3.pdf>. Acesso em: 07 Fev. 2023.
- Lakein, A. (1973). **How to Get Control of Your Time and Your Life**. New American Library. Disponível em: <https://archive.org/details/howtogetcontrol000lake>. Acesso em: 15 Fev. 2023.
- Shao, L., Liu, Z., & Liu, Y. (2016). **The Impact of Gamification on Cognitive Learning: A Comparative Experiment**. In **International Conference on Human-Computer Interaction** (pp. 302-309). Disponível em :<https://link.springer.com/article/10.1007/s10648-019-09498-w>. Accessed on: 02 Mar. 2023.
- Zichermann, G., & Cunningham, C. (2011). **Gamification by Design: Implementing Game Mechanics in Web and Mobile Apps**. O'Reilly Media. Disponível em: https://books.google.com/books/about/Gamification_by_Design.html?id=Hw9X1miVMMwC. Acesso em: 25 Mar . 2023.
- “**How to generate Class Diagram (UML) on Android Studio (IntelliJ Idea)**”. Available at:<https://stackoverflow.com/questions/17123384/how-to-generate-class-diagram-uml-on-android-studio-intellij-idea> . Accessed on: 10 Out. 2023.
- Lucidchart. “**UML Class Diagram Tutorial**”. Disponível em:<https://www.lucidchart.com/pages/uml-class-diagram> . Acesso em: 5 Out. 2023.
- Visual Paradigm. “**Creating Sequence Diagrams with a Use Case-Driven Approach: A Comprehensive Guide**”. Disponível em:<https://guides.visual-paradigm.com/creating-sequence-diagrams-with-a-use-case-driven-approach-a-comprehensive-guide/> . Acesso em: 09 Mar. 2023.
- “**draw.io**”. Disponível em:<https://www.drawio.com/> . Acesso em: 18 Mar. 2023.
- Android Developers. “**Android API reference**”. Disponível em:<https://developer.android.com/reference/> . Acesso em: 07 Abr. 2023.
- Firebase. “**Firestore API Reference**”. Disponível em:<https://firebase.google.com/docs/reference/> . Acesso em: 20 Abr. 2023.

Parsec. “**Technical Reference – Parsec**”. Disponível em: <https://support.parsec.app/hc/en-us/categories/360000336271-Technical-Reference> . Acesso em: 03 Nov. 2023.

DEVMEDIA. **Orientações básicas na elaboração de um diagrama de classes**. Disponível em: <https://www.devmedia.com.br/orientacoes-basicas-na-elaboracao-de-um-diagrama-de-classes/37224>. Acesso em: 02 Ago. 2023. Artigo 2

UFPR. **Writing a Technical Paper or Brief**. Disponível em: <http://ftp.demec.ufpr.br/CFD/orientacao/Writing%20a%20Technical%20Paper%20or%20Brief.htm>. Acesso em: 01 Jan. 2023.

UNITPAC. **Guia para apresentação de trabalhos Acadêmicos e TCC**. Disponível em: <https://assets.unitpac.com.br/arquivos/MANUAL-TCC-ITPAC-2017B.pdf>. Acesso em: 05 Mar. 2023.

UNIR. **NORMA ABNT NBR BRASILEIRA 14724**. Disponível em: <https://ppgham.unir.br/uploads/27093751/ABNT/NBR14724%20-%20Trabalhos%20academicos.pdf>. Acesso em: 09 Out. 2023.

Pavlov, I. P. (1953). **Distorções de Pavlov: ciência soviética e psicologia entre 1948 e 1953**. Disponível em: pepsic.bvsalud.org/scielo.php?script=sci_arttext&pid=S1519-549X2019000100005. Acesso em: 09 dez. 2023.

Alabau, I. **Ivan Pavlov: biografia e teoria do condicionamento clássico**. Psicologia-Online. Disponível em: <https://br.psicologia-online.com/ivan-pavlov-biografia-e-teoria-do-condicionamento-classico-239.html>. Acesso em: 09 dez. 2023.

eBiografia. **Biografia de Ivan Pavlov**. Disponível em: https://www.ebiografia.com/ivan_pavlov/. Acesso em: 09 dez. 2023.