



ISEL – INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA
ADEETC – ÁREA DEPARTAMENTAL DE ENGENHARIA DE
ELECTRÓNICA E TELECOMUNICAÇÕES E DE COMPUTADORES

LEIM

LICENCIATURA EM ENGENHARIA INFORMÁTICA E MULTIMÉDIA
UNIDADE CURRICULAR DE PROJETO

Aplicação ANDROID para utilizadores com dificuldade visuais



Érica Pereira (42356)

Orientador

Professor Pedro Fazenda

setembro, 2022

Resumo

O relatório descreve a pesquisa e o trabalho realizado para desenvolver uma aplicação para telemóveis ANDROID de um jogo de estratégia apto para utilizadores com deficiência visual.

Existe uma necessidade no mercado dos jogos para telemóveis de acessibilidade para pessoas com dificuldade visuais. Este trabalho tem com objetivo desenvolver uma aplicação demonstrativa de um jogo de estratégia. Foram realizados questionários para ajudar a entender quais as melhores formas de tornar os jogos de telemóvel mais acessíveis.

A ideia é tornar um jogo complexo num jogo mais acessível, explorando a hipótese de existir uma navegação pela aplicação e distribuição de botões mais intuitiva e/ou fácil de entender e lembrar.

Palavras-chave: Aplicação ANDROID; Jogo de Estratégia; Apto a pessoas com dificuldades visuais.

Abstract

The report describes the research and work carried out to develop an application for ANDROID mobile phones of a strategy game suitable for visually impaired users.

There is a need in the market for accessibility mobile games for people with visual impairments. This work aims to develop a demonstrative application of a strategy game. Questionnaires were carried out to help understand the best ways to make mobile games more accessible.

The idea is to turn a complex game into a more accessible game, exploring the possibility of having a navigation through the application and distribution of buttons more intuitive and/or easier to understand and remember.

Keywords: ANDROID application; Strategy Game; Suitable for people with visual difficulties.

Agradecimentos

A realização deste trabalho deve-se à contribuição do professor Pedro Fazenda pela direção e orientação integral do projeto, do professor Hélder Bastos pela direção provida em horário de aulas e discussões de tópicos significativos sobre o projeto, e pela ajuda em forma de apoio moral de Joana Torres Santos, Inês Raquel Faísca, Ana Sofia Coelho e, principalmente, Luís Alexandre Guimarães.

Índice

Resumo	2
Abstract	4
Agradecimentos	6
Índice	8
Lista de Figuras	10
1 Introdução	12
2 Trabalho Relacionado	14
3 Modelo Proposto	17
3.1 Requisitos	17
3.2 Fundamentos	19
3.3 Abordagem	20
4 Implementação do Modelo	25
5 Conclusões e Trabalho Futuro	32
A Casos de Utilização	34
B UML – Diagrama de Classes	36
C MEA – Modelo Entidade-Associação	37
D Questionário	38
E Respostas obtidas do Questionário	40
F Requisitos Funcionais e Não Funcionais	43

G Mock-Ups da aplicação de jogo	43
Bibliografia	48

Lista de Figuras

Figura 1 - Respostas da pergunta 3 do questionário online aplicado a 11 utilizadores.	14
Figura 2 - Telas do jogo AFK: Arena dos menus principais.....	15
Figura 3 - Casos de utilização: 5 casos de uso baseados na mecânica do jogo.....	18
Figura 4 - Divisão do ecrã em quadrantes intencionada para a navegação base da aplicação; divisão ao meio do ecrã excepcionalmente, durante batalhas entre dois personagens.	20
Figura 5 - Teoria das cores complementares.	22
Figura 6 - Cores complementares usadas nos ecrãs da aplicação.	22
Figura 7 - modelo MEA – Modelo Entidade-Associação simplificado.....	23
Figura 8 - modelo UML – Diagrama de Classes simplificado.....	24
Figura 9 - Ecrã de inicialização e criação/acesso à base de dados.	27
Figura 10 - Ecrã principal do menu, com o botão de recompensas; Ecrã das recompensas.	28
Figura 11 - Ecrãs de batalha, vitória e derrota.	29

Capítulo 1

Introdução

Este projeto tem como objetivo principal desenvolver acessibilidade numa aplicação de um jogo a pessoas com dificuldades visuais. Em segundo plano, o trabalho tem como objetivo desenvolver um jogo de estratégia (Mythical Battles) interessante e de jogabilidade adequada.

Mythical Battles é um jogo de estratégia, que consistem na recolha de recompensas acrescidas ao longo do tempo, com o intuito de adquirir e colecionar novos personagens e desbloquear os seus poderes especiais que agilizam no combate a inimigos. Documento sobre o jogo e o seu funcionamento em anexo ao relatório (GDD – Documento de Design de Jogo, nome do ficheiro: GDD – Mythical Battles.pdf).

Estão disponíveis um número considerado de jogos para computador ou consolas, como por exemplo o jogo Lost and Hound para computador e consola, mas, a nível de smartphones, em comparação existem disponíveis mais aplicações de auxílio quotidiano a pessoas cegas do que jogos divertidos, interessantes e relaxantes ao utilizador, de modo que a motivação do trabalho foi encontrar uma necessidade que ainda não tinha sido atingida.

Este relatório relata os modos de acessibilidade possíveis, as funcionalidades que a aplicação necessita para concretizar o jogo pretendido e a acessibilidade e jogabilidade desejadas, a realização do produto a fabricar e as ferramentas necessárias para tal e, por fim, os resultados obtidos do projeto.

O trabalho foi realizado em Flutter, na linguagem Dart.

Capítulo 2

Trabalho Relacionado

Existem vários jogos de telemóvel com os objetivos e mecânicas de jogo similar ao jogo de estratégia desenvolvido neste trabalho.

Este tipo de jogos são, na maioria, sobre a aquisição e evolução de personagens fictícias com diferentes estilos e tipos de ataque diferentes, que são posteriormente postos em combate contra IA's ou contra outros jogadores. Exemplos de jogos deste gênero são: AFK: Arena, Mythic Heroes, Dislyte e Super Auto Pets, entre outros.

Estes, e outros jogos disponíveis para telemóvel, são visualmente e funcionalmente complexos, onde, num só ecrã, é possível que se encontrem cerca de 15 ou mais botões com funcionalidades distintas (Figura 2), como é o caso do jogo AFK: Arena, e, dificilmente, com acessibilidade a pessoas cegas ou com baixa visão.

Do questionário aplicado online (Anexo D e E), 45,5% dos utilizadores gostaria de ter acesso a mais jogos, seguido de 36,4% de utilizadores que gostaria de jogar jogos, mas não encontra jogos bons e acessíveis (Figura 1).

3 - Gosta ou gostaria de jogar jogos no telemóvel?

11 respostas

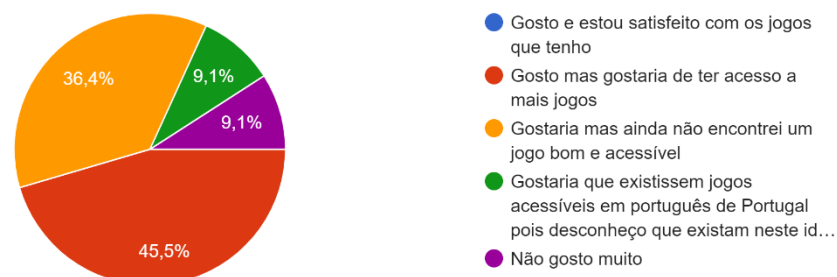


Figura 1 - Respostas da pergunta 3 do questionário online aplicado a 11 utilizadores.



Figura 2 - Telas do jogo AFK: Arena dos menus principais.

A Google Play Store, o marketplace de apps do Android, contém um filtro designado por blind-friendly que dá ênfase a jogos que tenham a população de pessoas cegas ou amblíopes em mente durante o desenvolvimento do jogo (de momento descontinuado).

Existem jogos que utilizam o som como mecanismo fundamental de jogo, mas o tipo de jogos mais frequentes para pessoas com dificuldades visuais são jogos de narrativa, à base de texto e que permitam a narração.

O intuito do jogo desenvolvido foi mostrar ser possível desenvolver jogos à base de turnos, semelhantes a jogos bastante difundidos e populares e que podem alcançar mais audiências, tendo a acessibilidade em mente.

Capítulo 3

Modelo Proposto

3.1 Requisitos

Para a análise de requisitos foram determinados os seguintes requisitos funcionais e respetivos requisitos não funcionais (Anexo F):

- Configurações: Alterar volume e velocidade do áudio, com necessidade de que o tempo de resposta seja mínimo, sendo desejável no máximo o tempo de 2 segundos; ativar ou desativar a vibração, o texto auditivo e os efeitos sonoros; vincular a conta do utilizador;
- Notificações Auditivas: Sinalizar todas as ações feitas na aplicação auditivamente com tempo de resposta máximo de 2 segundos – Anunciar mudanças de ecrãs, sinalizar toques com efeitos sonoros, sinalizar perda ou vitória de uma rodada em batalha, anunciar nome do botão tocado;
- Notificações Vibratórias: Sinalizar ações feitas a aplicação, com diferentes sequências possíveis para distinguir respostas – sinalizar toques com uma vibração leve, 2 vibrações ao recolher recompensas, 4 vibrações seguidas em sequência de batimento cardíaco em caso de vitória e 1 vibração prolongada em caso de derrota;

- **Discrições Auditivas:** Descrever auditivamente personagens e itens, através de texto de estatísticas de vida, força, Liga e tipo de ataque e características detalhadas; Descrever auditivamente ajudas, perguntas frequentes e tutoriais de como navegar pela aplicação e de como jogar - auditivamente com tempo de resposta máximo de 2 segundos;
- **Mecanismos de jogo:** Listar as personagens e itens adquiridos, recolher recompensas, iniciar a batalha e selecionar heróis de combate. Com usabilidade focada em toques de ecrã e caixas de diálogo.

Nos casos de utilização (Figura 3), a aplicação só tem um ator – o jogador.

Dos requisitos funcionais, foram extraídos no total 14 casos de uso (Anexo A): Equipar herói, aceder a informação detalhado do herói, listar todos os heróis, listas heróis adquiridos, listar itens adquiridos, recolher recompensas, selecionar heróis para combater, ver oponentes, iniciar batalha, começar rodada manualmente, ativar/desativar vibração, alterar volume efeitos sonoros, alterar volume áudio de voz e alterar velocidade do áudio de voz.

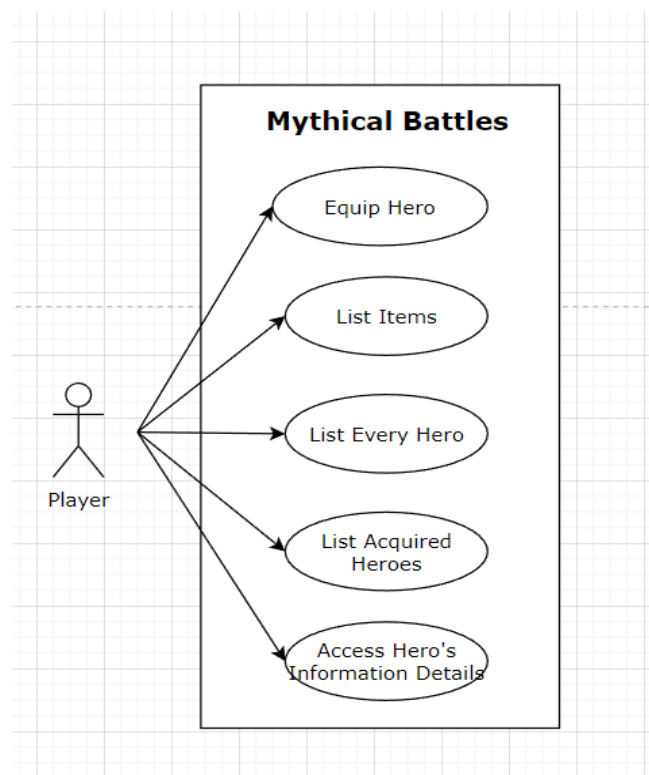


Figura 3 - Casos de utilização: 5 casos de uso baseados na mecânica do jogo.

3.2 Fundamentos

Atualmente, estima-se que 39 milhões de pessoas no mundo sejam pessoas cegas e que 246 milhões estejam a sofrer de perda moderada ou severa da visão. Em Portugal há 160 mil pessoas com problemas de cegueira. Destes, 12,5% são pessoas totalmente cegos e 87,5% são pessoas amblíopes (com baixa visão).

Este trabalho pretende desenvolver uma aplicação que facilite a jogabilidade e acessibilidade a pessoas com baixa visão e, principalmente, a pessoas totalmente cegas.

Posto isto, o foco de acessibilidade na aplicação é o áudio e a vibração, que são o meio de entregar informação atingindo os sentidos de audição e tato dos utilizadores.

Por outro lado, em segundo plano, a aplicação vai incluir animações, texto de tamanho médio a grande e cores contrastantes para facilitar utilizadores com baixa visão que consigam assimilar informação exposta desta forma.

3.3 Abordagem

Para ter em conta as respostas de utilizadores, foi aplicado um questionário online (Anexos D e E), que obteve, no seu total, 11 respostas de utilizadores.

Após análise, concluiu-se que as interações mais desejadas entre os participantes são toques e arrastos rápidos com o dedo, assim como, um sistema de reconhecimento de fala, seguido de arrastos sem largar e sinais de vibração.

Em questão de acessibilidade, a aplicação vai-se basear substancialmente no áudio e vibração para obter informação sobre o ecrã.

Porém, um dos focos da aplicação, no que toca a distinguir-se de aplicações já existentes, vai ser a disposição dos botões de informação e das funcionalidades:

- Os ecrãs vão ser divididos em quatro quadrantes, equivalentes a 4 botões (Figura 4), e, excecionalmente, caso seja necessário e/ou intuitivo, irá ser dividido ao meio.



Figura 4 - Divisão do ecrã em quadrantes intencionada para a navegação base da aplicação; divisão ao meio do ecrã excecionalmente, durante batalhas entre dois personagens.

Deste modo, tem-se expectante que o utilizador tenha mais facilidade em encontrar informações, funcionalidades e a navegar pela aplicação, de forma que, posteriormente, a aplicação se torne intuitiva e fácil de entender e relembrar.

Do ponto de vista da experiência do utilizador, sempre que o utilizador interage com uma aplicação, deve ser dado um feedback demonstrando sucesso, falha ou caso a sua conclusão esteja pendente.

Este feedback pode ser visual, sonoro ou tactual. Em questão de acessibilidade, se uma aplicação fornecer vários tipos de feedback em simultâneo, mais acessível ao utilizador e a todas as pessoas se torna.

Um exemplo da necessidade de feedback pode ser um botão que efetua uma ação em específico. Tal evento pode disparar uma função que modifique um valor em questão, mas, caso não haja feedback, visual, sonoro ou de outro tipo, o utilizador fica sem saber se a sua ação foi executada e pode tentar repetir a sua ação indefinidamente, desconhecendo o facto de que está a repetir a mesma ação com sucesso múltiplas vezes. Logo, um feedback é necessário para confirmar as ações dos utilizadores.

Feedback visuais necessitam que o utilizador esteja a prestar atenção.

Feedbacks sonoros podem ser considerados inoportunos ou até mesmo fatigantes caso cada interação com a aplicação seja acompanhada duma resposta sonora.

A vibração é um tipo de feedback mais sutil, mas mais básico. A vibração será um indicador tátil ao interagir com o ecrã, como por exemplo, para confirmar que o utilizador interagiu com um dos quadrantes da aplicação, e, com possíveis padrões de sequências de vibração, sinalizar diferentes resultados do jogo (vitórias ou derrotas) ou outras funcionalidades diferenciadas.

Posto isto, cada botão e ação disponíveis na aplicação terá um áudio descritivo ou efeito sonoro a indicar a funcionalidade ou ação em questão. Os áudios serão em português.

Em relação ao input visual, os quadrantes serão distintos pelo utilizador ambliópe potencialmente através da utilização de cores complementares (Figuras 5 e 6). Desta forma, os quadrantes de um ecrã terão, entre si, as cores que mais se contrastam.

Círculo Cromático

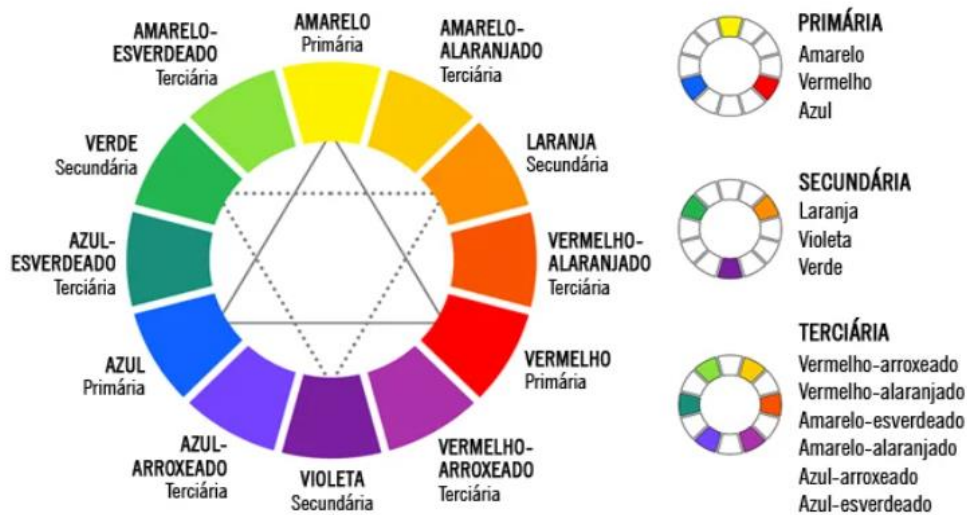


Figura 5 - Teoria das cores complementares.

You can also directly paste your color's HEX value...

#FFC864

Your picked color



HEX:
#FFC864

Red: 255
Green: 200
Blue: 100

Complementary color



HEX:
#7D7BA9

Red: 125
Green: 123
Blue: 169

You can also directly paste your color's HEX value...

#78E6FF

Your picked color



HEX:
#78E6FF

Red: 120
Green: 230
Blue: 255

Complementary color



HEX:
#FF835D

Red: 255
Green: 131
Blue: 93

Figura 6 - Cores complementares usadas nos ecrãs da aplicação.

O modelo MEA – Modelo Entidade-Associação (Figura 7 e Anexo C – completo) descreve:

- O jogador adquire 1 quantidade de recompensas e as recompensas são adquiridas por diversos jogadores. A quantidade depende do nível do jogador;
- O jogador joga um único capítulo por vez e cada capítulo pode ser jogado por vários jogadores. À medida que o jogador vai vencendo, vai alterando de capítulo;
- Cada herói é uma única personagem. Uma personagem contém informações inalteráveis. Um herói tem estatísticas que evoluem ao longo do jogo;
- O jogador tem vários heróis, sejam eles da mesma personagem ou não. Sendo assim, cada herói tem de ter um identificador, para diferenciar heróis com a mesma personagem pertencentes a um único utilizador;
- O caso das relíquias é igual ao dos heróis: Cada relíquia é um único item. O jogador tem várias relíquias, sejam eles do mesmo item ou não, por isso, cada relíquia tem de ter um identificador próprio;
- Um herói é equipado com uma única relíquia, ambos pertencentes a um jogador.

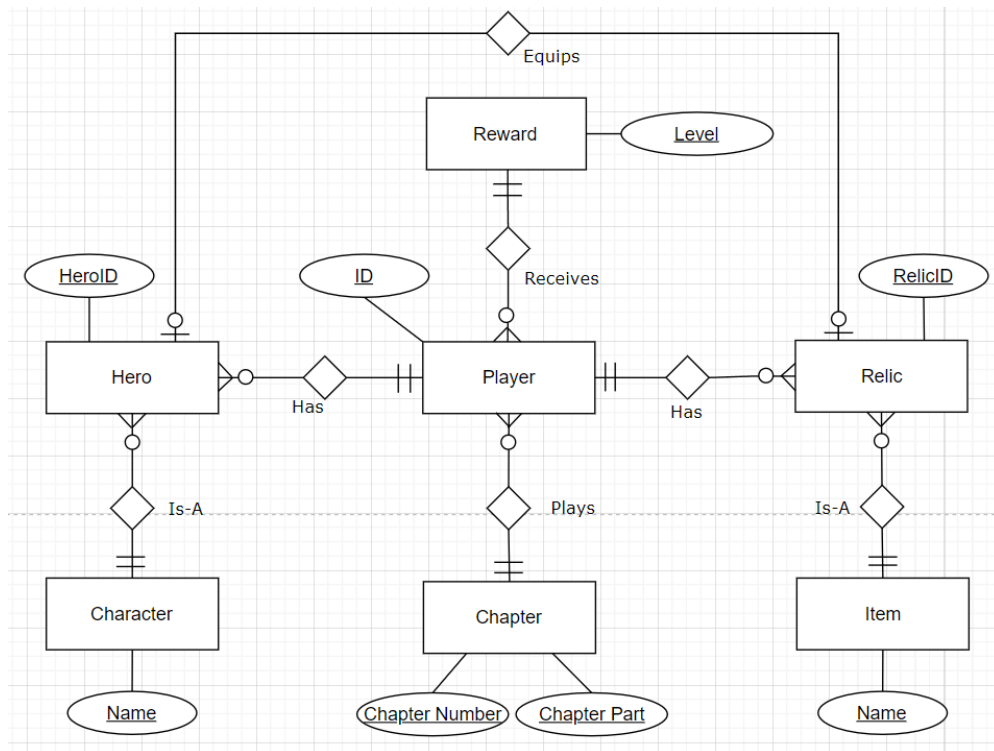


Figura 7 - modelo MEA – Modelo Entidade-Associação simplificado.

O modelo UML – Diagrama de Classes (Figura 8 e Anexo B – completo) dita que a relíquia e o herói fazem parte da classe do utilizador, existindo sem este. No entanto, a relíquia e o herói não existem sem a classe Item e a classe Personagem, respetivamente.

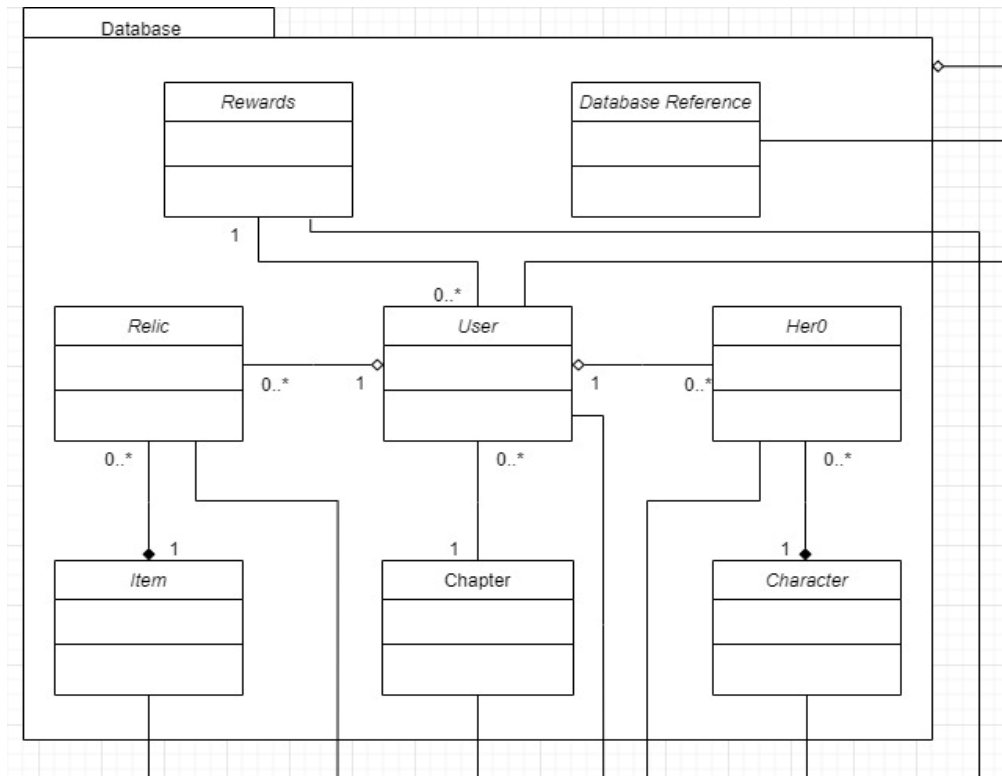


Figura 8 - modelo UML – Diagrama de Classes simplificado.

Capítulo 4

Implementação do Modelo

Para a implementação do modelo, aplicação utiliza uma base de dados SQL para o armazenamento de dados – SQLite.

- **SQLite**

Foi escolhida uma base de dados SQL em vez de uma NoSQL devido à quantidade de restrições impostas pelo modelo do jogo, sendo preferível impor essas restrições a nível da base de dados do que implementá-las no código da aplicação. No contexto do jogo, os dados estão fortemente relacionados entre si.

No contexto da aplicação, a este nível não é necessário armazenar uma quantidade de dados que justifique usar bases de dados mais robustas com autenticação, muitos tipos de dados e que suportem muitas ligações em simultâneo.

Para além disso, dispensa da necessidade de ter um servidor a correr na rede para efetuar a interação com os dados armazenados, a base de dados está embutida na aplicação e não necessita de um servidor nem configuração ou administração.

A aplicação desenvolvida utiliza a framework de desenvolvimento Flutter, através da linguagem de programação open source Dart, disponibilizada pela Google.

- **Flutter vs Android Studio**

Apesar de providenciar menos funcionalidades que programar para Android de forma nativa em Java ou Kotlin, é multiplataforma, isto é, uma aplicação desenvolvida em Flutter para além de ser correr em Android, também é compatível com dispositivos móveis iOS, desktops e para a web.

Em relação à vibração, embora o Flutter não tenha tanta diversidade como o Android nativamente, através do uso de pausas é possível criar padrões de feedback para variados cenários como uma vibração simples ou uma vibração constante ou uma vibração periódica, como por exemplo, um batimento cardíaco.

- **Flutter vs React Native**

Com o intuito de explorar outras linguagens de programação foi feita uma escolha entre React Native e Flutter. Após ter sido feito um estudo comparativo entre as duas, concluiu-se que o Flutter seria mais indicado por ter alto performance e ser mais adequado para aplicações centradas em UI's, as quais estão fortemente presentes na aplicação desenvolvida.

O uso de React Native é recomendado para projetos mais complexos e extensos devido à grande quantidade de bibliotecas disponíveis, o que justificaria a perda de performance devido ao facto de utilizar JavaScript como linguagem de programação.

O Flutter também fornece uma funcionalidade de “Hot Reload” que permite fazer modificações ao código mais rapidamente, sendo ideal para experimentações com interfaces gráficas.

A implementação do trabalho, em Flutter, consiste na construção de Widget's para cada ecrã.

- **Inicialização**

A aplicação inicia com um ecrã de inicialização (Figura 9), onde cria a base de dados, se inexistente, ou abre a base de dados. Após ter acesso à base de dados, identifica o utilizador através de reconhecimento do identificador do dispositivo, que passa a ser o identificador do utilizador na base de dados.



Figura 9 - Ecrã de inicialização e criação/acesso à base de dados.

- **Áudio**

Para o som e áudio da aplicação, utilizou-se a biblioteca audioPlayer, onde, nos ecrãs das definições, é possível alterar o volume e a velocidade dos sons.

Para manter a aplicação rápida e menos caótica, sempre que um áudio é inicializado enquanto um outro áudio está a ser tocado, este último é parado e começa o novo áudio, de modo a não haver sobreposição de áudios. Também foi criada uma função de sequência de áudios, para áudios com descrições de estatísticas e informações detalhadas.

- **Vibração**

Para a vibração, utilizou-se o HapticFeedback, sendo que a biblioteca Vibration, que, em teoria, seria mais completa e com mais modos de vibração, nem sempre funciona com qualquer ANDROID, tendo complicações com as permissões.

- **Acúmulo de recompensas**

Foi implementado o sistema de acumulação de recompensas (Figura 10). O utilizador assim que instala a aplicação no telemóvel e recebe a primeira recompensa, o contador começa a contar o tempo e a acumular recompensas, até o utilizador quiser coletar recompensas novamente.

Dependendo do nível do utilizador, o acúmulo de recompensas pára a um determinado tempo. O utilizador, mesmo passado um dia inteiro, coleta a quantidade de recompensas que foram acumuladas até ter parado automaticamente.

As recompensas recebidas são apresentadas numa caixa de diálogo.

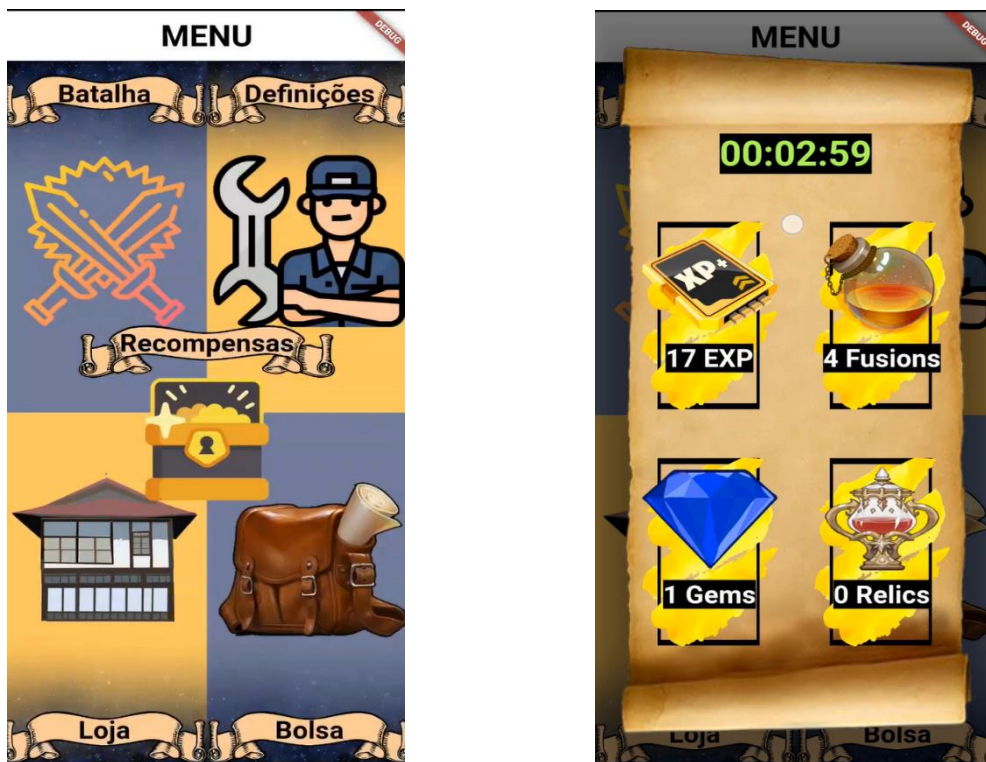


Figura 10 - Ecrã principal do menu, com o botão de recompensas; Ecrã das recompensas.

- **Jogo de batalha**

Desenvolveu-se o jogo de batalha (Figura 11): antes da batalha o utilizador pode ver os oponentes e seleccionar heróis para combater de uma lista só com os heróis que o utilizador já adquiriu.

Após seleccionar 4 heróis, o jogador pode começar a jogar.

O jogo tem 4 botões (jogar, jogar automaticamente, ver estatísticas do herói, ver estatísticas do oponente), mas visualmente o ecrã está dividido ao meio, para diferenciar o herói do oponente.

O verdadeiro jogo encontra-se na preparação dos heróis, colocação dos heróis por ordem e equipamento das relíquias certas, tendo em conta os adversários. O jogo em si consta em demonstrar os resultados das escolhas feitas, carregando no botão de jogar a rodada.

No final, o utilizador é redirecionado para o ecrã onde mostra o resultado da batalha, onde o jogador, ou acaba vitorioso e recebe recompensas ou acaba em derrota.

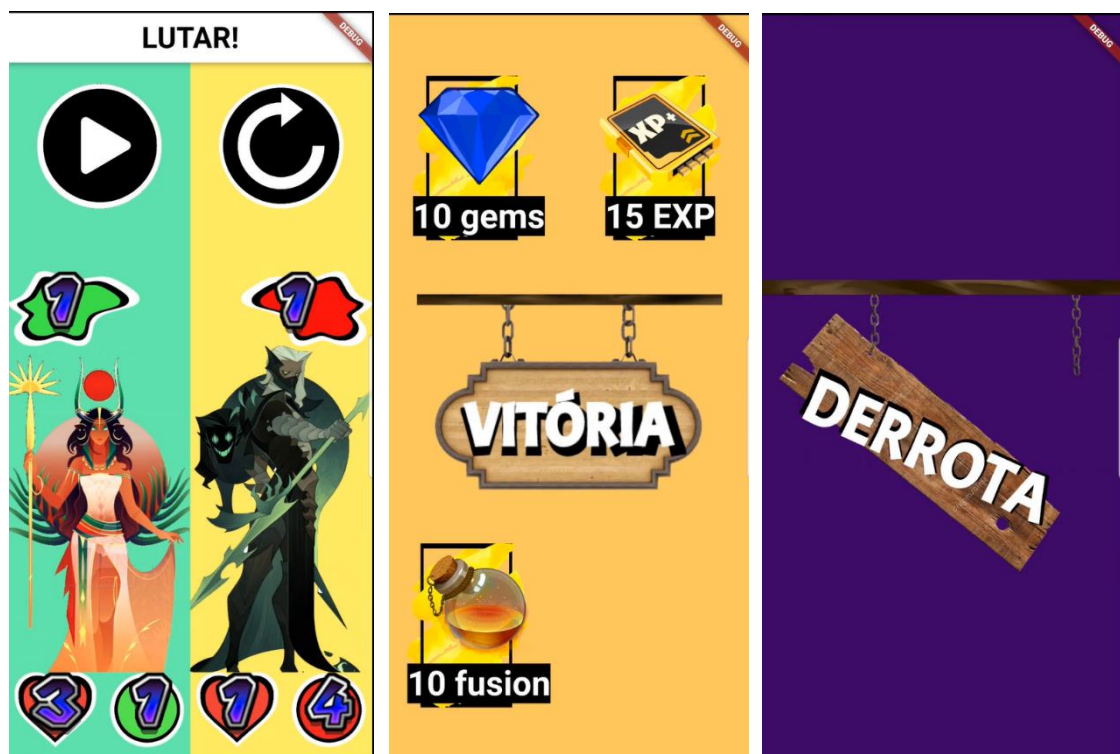


Figura 11 - Ecrãs de batalha, vitória e derrota.

- **Navegação**

Para navegar pela aplicação, implementaram-se toques singulares para declarar auditivamente o nome do botão tocado e dois toques para ativar a funcionalidade do botão. Para recolher recompensas, o utilizador tem a opção de arrastar com o dedo para cima, e, para sair das recompensas, novamente os dois toques ou, arrastar para baixo.

Ao navegar para um outro ecrã, se o utilizador quiser retornar ao ecrã anterior, essa função é ativada arrastando o dedo para a direita.

Para começar a jogar o jogo de batalha, pode-se toques duas vezes no botão com a funcionalidade para começar, ou, na página “Batalha” ou na página “Selecionar Heróis”, o jogo também começa se se arrastar com o dedo para a esquerda.

Para passar pelas rondas do jogo de batalha, também se pode carregar duas vezes no botão com essa funcionalidade ou arrastar com o dedo para a esquerda.

- **Animação**

Para além das cores complementares e texto em letras grandes, foram também implementadas algumas animações na aplicação dos heróis.

No ecrã do herói, onde exhibe a informação detalhada do herói, a imagem do deste no centro executa uma animação repetitiva de translação para cima e para baixo, a uma velocidade moderada, do tipo idle.

No jogo de batalha, os heróis e os oponentes surgem nos ecrãs com uma animação de escalamento do tamanho 0 ao tamanho equivalente a metade do ecrã. Caso um herói ou oponente perca a vida total, este executa uma animação de translação de cima para baixo duas vezes rapidamente, a sinalar o recebimento do dano causado pelo adversário.

- **Equipar relíquia**

O utilizador pode equipar um dos seus heróis com uma das suas relíquias. Esta função atribui mais pontuação ao herói nas batalhas, de acordo com os valores que a relíquia dá. Caso a relíquia seja compatível com o herói que a equipou, nas batalhas, o poder especial desse herói é ativado e executado.

A restrição de que uma dada relíquia só pode pertencer a um único herói e um herói só pode equipar uma única relíquia foi concretizada, sendo atribuição de uma relíquia já equipada num herói removida e transitada para o novo herói pretendido.

Capítulo 6

Conclusões e Trabalho Futuro

Neste trabalho foram englobadas táticas de transmitir informação através de sentidos tais como o tato e a audição.

Desenvolveu-se um jogo de estratégia em Flutter, com base em batalhas contra inimigos similares, mas com atributos diferentes. Implementou-se um sistema de acúmulo de recompensas cronometradas.

Criação de uma base de dados local em sqlite, com transações e atualizações de informação sobre o utilizador e os seus heróis e as suas relíquias.

Foi tido em conta a utilização por parte de jogadores cegos e jogadores amblíopes (com baixa visão), através de efeitos sonoros e áudios descritivos e intuitivos, com padrões de vibração, e, para utilizadores com baixa visão, incluiu-se cores contrastantes, texto em letras grandes e animações em heróis.

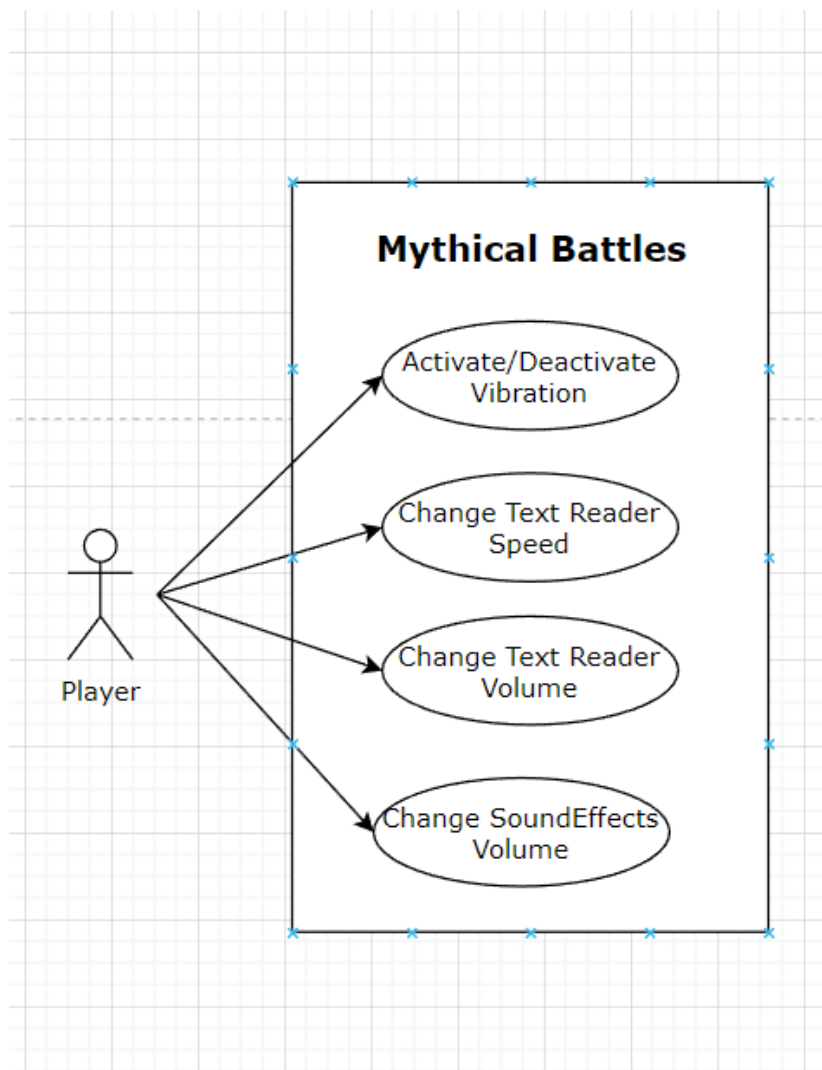
Para trabalho futuro, tem-se como pretensões:

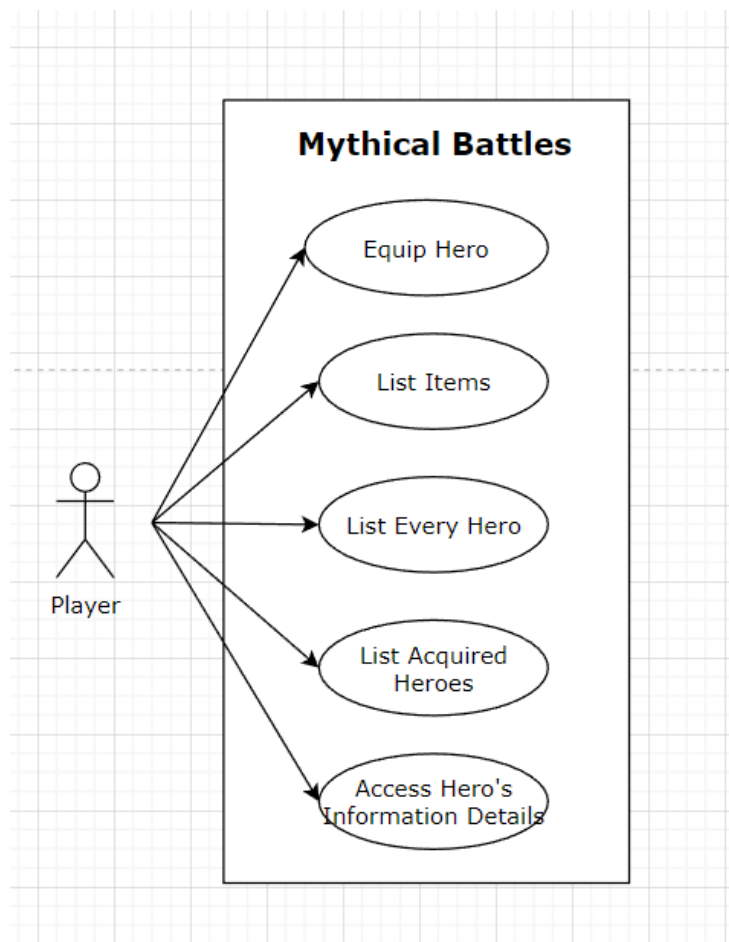
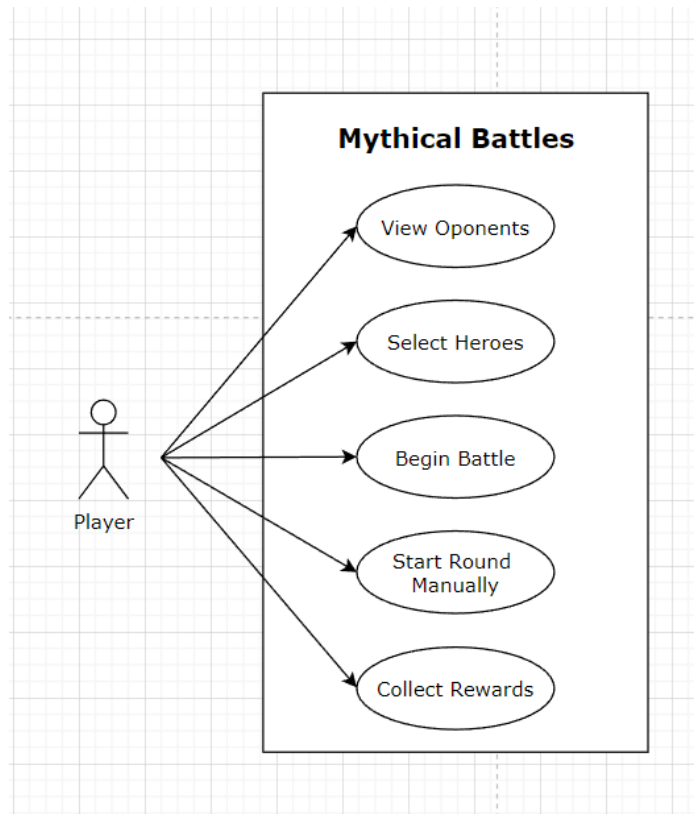
- Tutoriais interativos sobre a aplicação, funcionalidades e de como jogar. Uma passagem básica pela aplicação direcionado o utilizador, recetível a respostas tácteis pelo utilizador a ações direcionadas pelo tutorial;
- Implementar a Loja do jogo, e trocar recompensas por pacotes de heróis, pacotes de relíquias e outros itens;
- Melhorar os áudios, as imagens e a projeção dos títulos e botões de modo a ficar mais compreensível e nítido;
- Acrescentar a funcionalidade de pesquisa por reconhecimento de voz;
- Vincular a conta do utilizador;

- Completar atualizações de informação da base de dados, como por exemplo, atualizar o nível do utilizador e as suas recompensas adquiridas, assim como, avançar para o próximo capítulo;
- Tornar a base de dados numa base de dados Serverful e realizar batalhas entre jogadores e os seus heróis;
- Desenvolver a funcionalidade de fundir heróis ou fundir relíquias e aumentar o seu nível.

Apêndice A

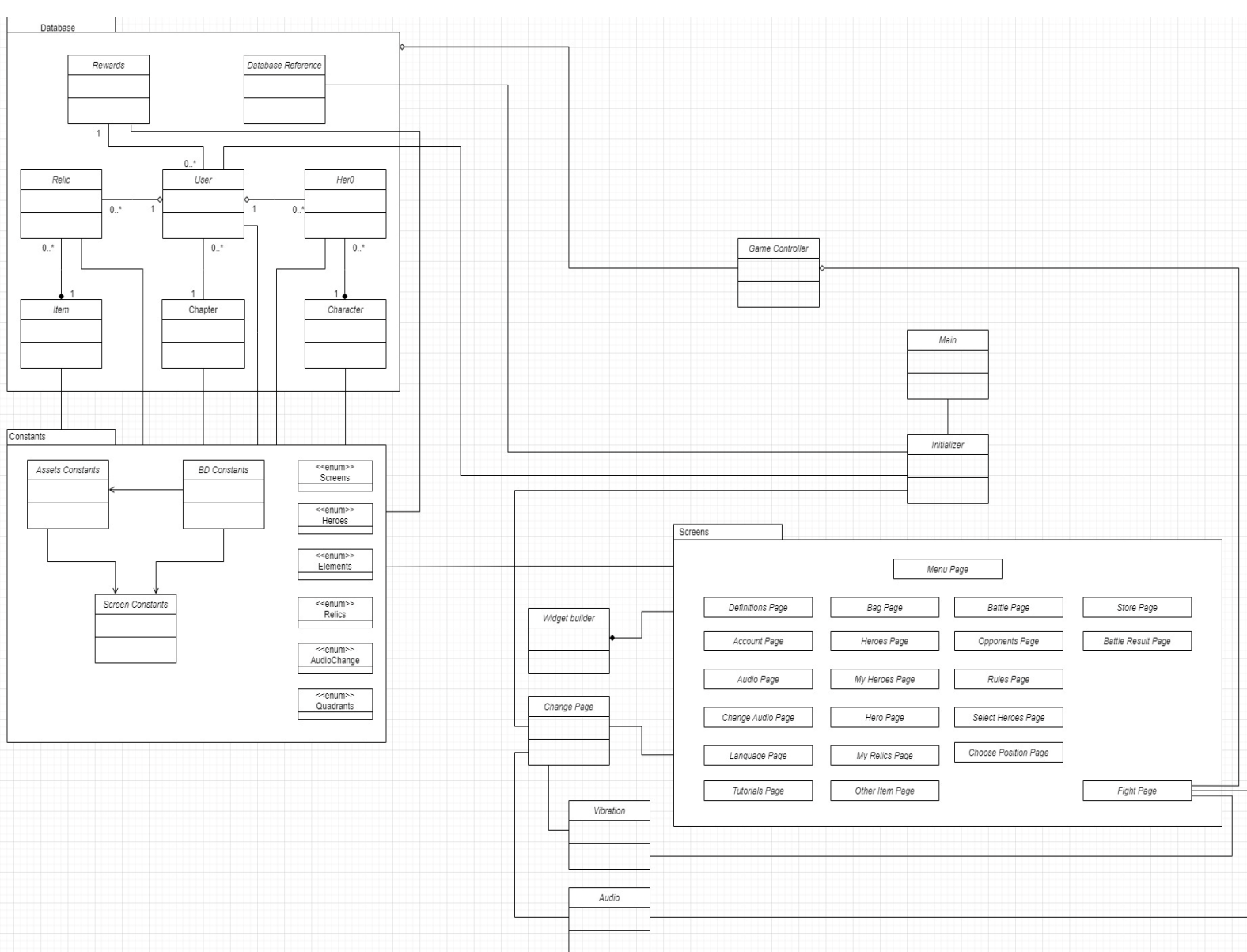
Casos de Utilização





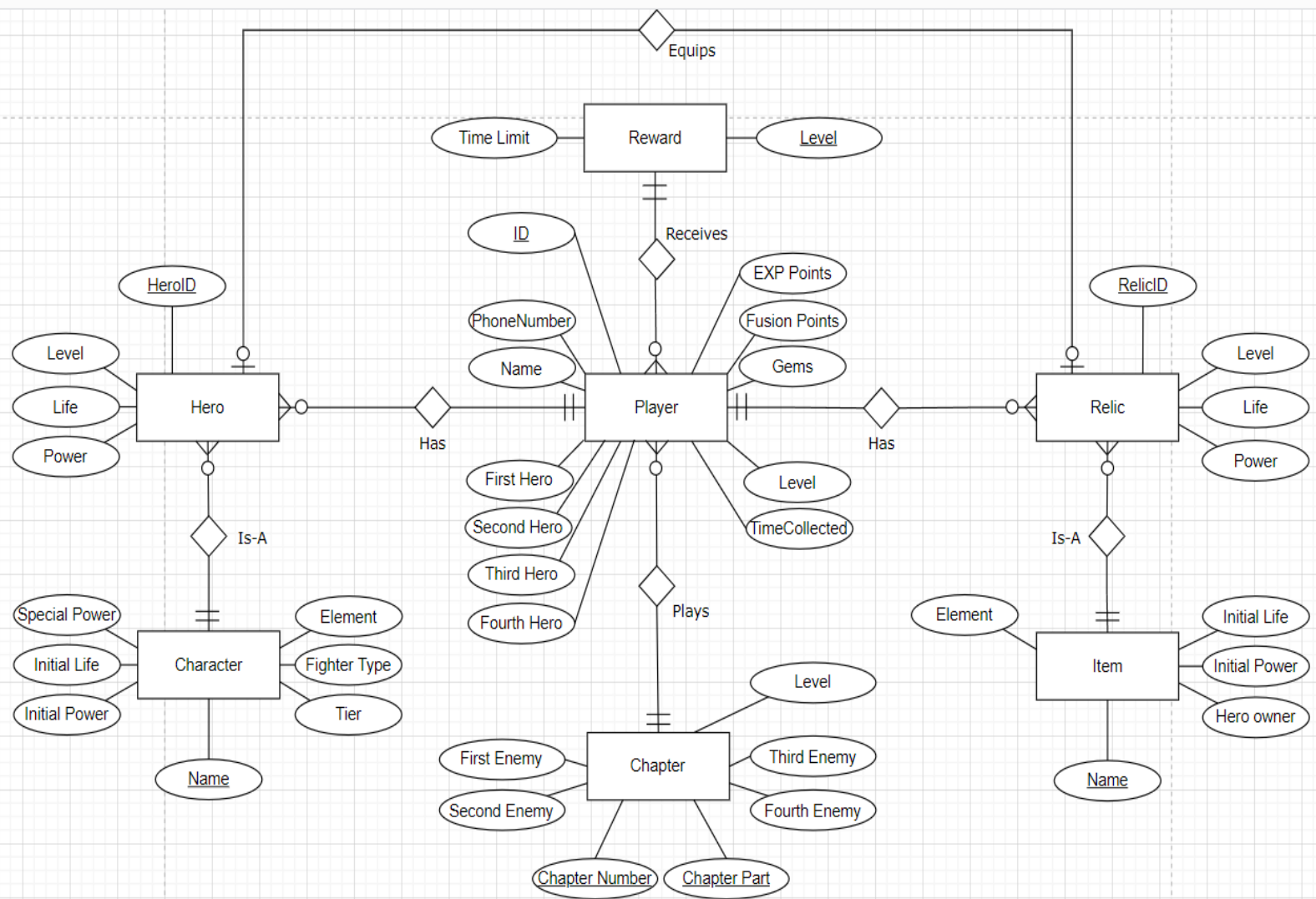
Apêndice B

UML – Diagrama de Classes



Apêndice C

MEA – Modelo Entidade-Associação



Apêndice D

Questionário

<https://forms.gle/Re22UuQFx6wJEopAA>

Perguntas:

1 - De que forma costuma utilizar as aplicações que tem no telemóvel?

Pode seleccionar mais do que 1 resposta.

- ☐ Não sou eu que mexo
- ☐ Uso um teclado braille ou outro auxiliar externo
- ☐ Toques com o dedo
- ☐ Faço arrastos rápidos com o dedo
- ☐ Arrasto o dedo pelo ecrã sem soltar
- ☐ Tenho alguém que me auxilia quando necessário
- ☐ Uso reconhecimento de fala, como por exemplo: a Siri
- ☐ Sinto as vibrações do telemóvel
- ☐ Outra: _____

2 - Na sua opinião, qual é a melhor maneira ou maneiras de navegar/utilizar uma aplicação?

A sua resposta _____

3 - Gosta ou gostaria de jogar jogos no telemóvel?

- ☐ Gosto e estou satisfeito com os jogos que tenho
- ☐ Gosto mas gostaria de ter acesso a mais jogos
- ☐ Gostaria mas ainda não encontrei um jogo bom e acessível
- ☐ Outra: _____

4 - Se gostaria de jogar pela primeira vez ou gostaria de jogar mais, mas não o faz, é porque ...

Pode seleccionar mais do que 1 resposta.

- ☐ Nunca pesquisei e desconheço aplicações de jogos
- ☐ Não encontro jogos acessíveis para mim
- ☐ Tentei uns jogos mas eram muito difíceis de jogar/mexer
- ☐ Outra: _____

5 - Quais os jogos que gosta ou gostaria de jogar?

A sua resposta

6 - Qual a maior dificuldade que encontra ou do que menos gosta nos jogos que joga ou que tentou jogar?

Por exemplo, em relação a acessibilidade, modo de jogo, áudio, ou algo em falta, etc.

A sua resposta

7 - Quer, ainda, indicar algo que gostaria que se refletisse no jogo que tenciono desenvolver? (por exemplo: tipo de jogo, personagens, áudio, ...)

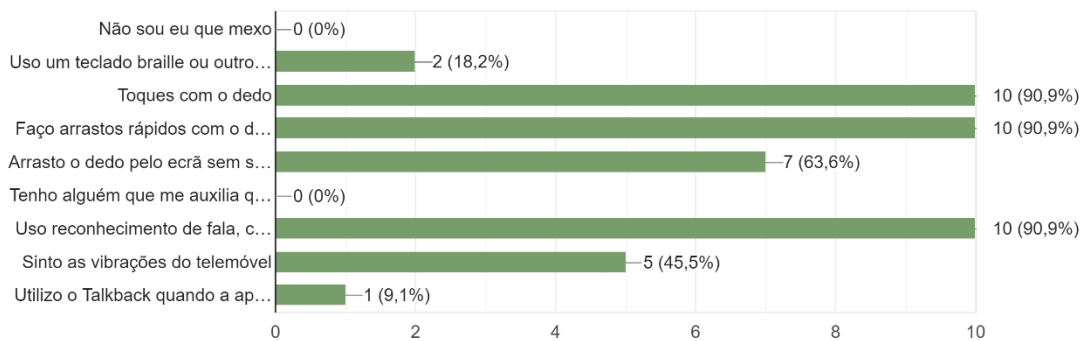
A sua resposta

Apêndice E

Respostas obtidas do Questionário

1 - De que forma costuma utilizar as aplicações que tem no telemóvel?

11 respostas



2 - Na sua opinião, qual é a melhor maneira ou maneiras de navegar/utilizar uma aplicação?

8 respostas

Através de gestos básicos de deslizamento para a os lados ou para cima e para baixo com a vibração ou a voza auxiliar a localização dos itens

arrastos para navegar e toque duplo para ativar

Usando auxílio da siri

Se estivermos a navegar nessa aplicação pela primeira vez, se acharmos necessário podemos sempre pedir auxílio a alguém. Se dominarmos o telemóvel podemos ir à descoberta por nós mesmos.

Varrendo com o dedo ou explorando dependendo da situação.

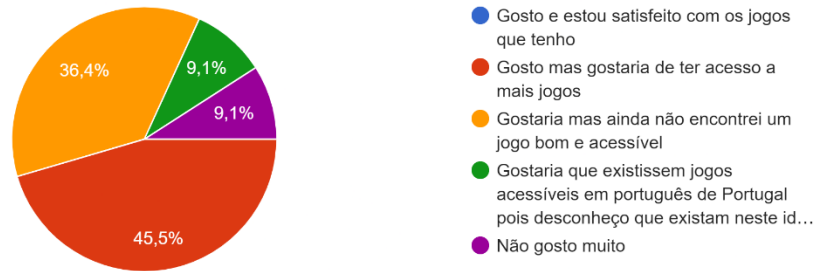
navegação por varrimento e 2 toques.

Varrer com o dedo para a direita/esquerda, como se fosse sacodir uma migalha.

usando o boi souber neste caso o Google

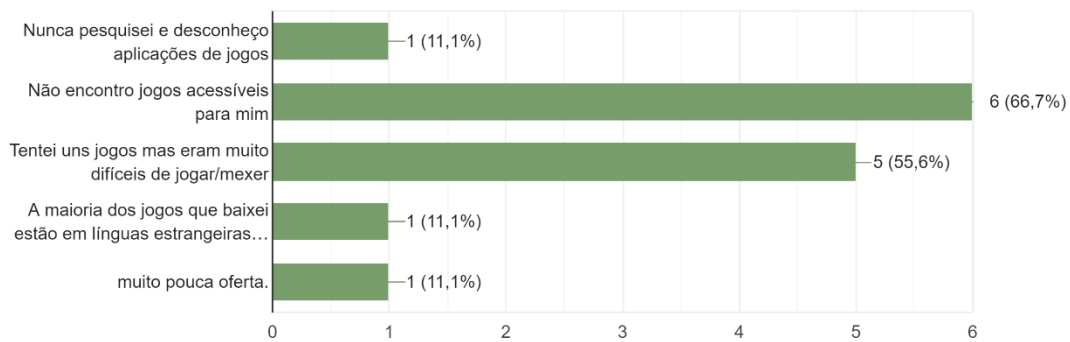
3 - Gosta ou gostaria de jogar jogos no telemóvel?

11 respostas



4 - Se gostaria de jogar pela primeira vez ou gostaria de jogar mais, mas não o faz, é porque ...

9 respostas



5 - Quais os jogos que gosta ou gostaria de jogar?

7 respostas

Estratégia, aventura. tabuleiro como xadrez e damas, cultura geral, rpg, quebra-cabeças, lógica e arcade

gostaria de ver jogos de fps, estratégia, como o soundrts ou algo parecido com o top speed

Jogos de cartas

Jogos de gerenciamento de equipes, "elifoot".

Gostaria de jogar jogos de RPG, jogos de ação.
Jogo perguntados pelo aplicativo e por um pelo site.

carros, futebol e luta.

futebol e estratégia'

6 - Qual a maior dificuldade que encontra ou do que menos gosta nos jogos que joga ou que tentou jogar?

8 respostas

A música de fundo ou os sons num tom muito alto confundem-me; as vozes gravadas serem robotizadas ou demasiado roufenhas; demasiados menus ou travamentos devido a conflitos de compatibilidade; por estarem em inglês; por serem demasiado fáceis e repetitivos; por serem apenas demonstrações e exigirem pagamento ou registo; por só estarem disponíveis para sistemas operativos como Windows e Linux

Uma personalização de gestos, por exemplo, você poder definir que ação realizar quando arrastar um dedo da esquerda pra direita, por exemplo

Não são muito acessíveis

Falta de acessibilidade em jogos que sirvam tanto para pessoas cegas quanto para videntes.

Pouca acessibilidade

Na maioria dos jogos, falta de acessibilidade.
Não sou de acessibilidade com leitores, mais acessibilidade de áudio descrição.

Falta de acessibilidade da própria app.

falta de acessibilidade

7 - Quer, ainda, indicar algo que gostaria que se refletisse no jogo que tenciono desenvolver? (por exemplo: tipo de jogo, personagens, áudio, ...)

5 respostas

Desafios com escolha múltipla, possibilidade de utilizar a voz nativa do telemóvel em vez de vozes gravadas; diferentes personagens com poderes distintos; música de fundo opcional ou com efeitos relaxantes; adaptação de jogos acessíveis de plataforma Windows para Androide, como, por exemplo, Audio Disco, Freedom Millionaire e O Segredo do Mosteiro

A biblioteca de áudio bass, tem uma ótima qualidade para jogos. Fica a sugestão.

Melhorar o leitor de ecrã nos jogos

Jogo jogo de RPG, aonde a gente pode fazer escolha com os determinados personagens.
Algumas telas ou fases.
Principalmente acessibilidade com leitores e por que não mal de descrição do jogo.
No começo do jogo, seria interessante também os atalhos ou dicas de como jogar.

carros

Apêndice F

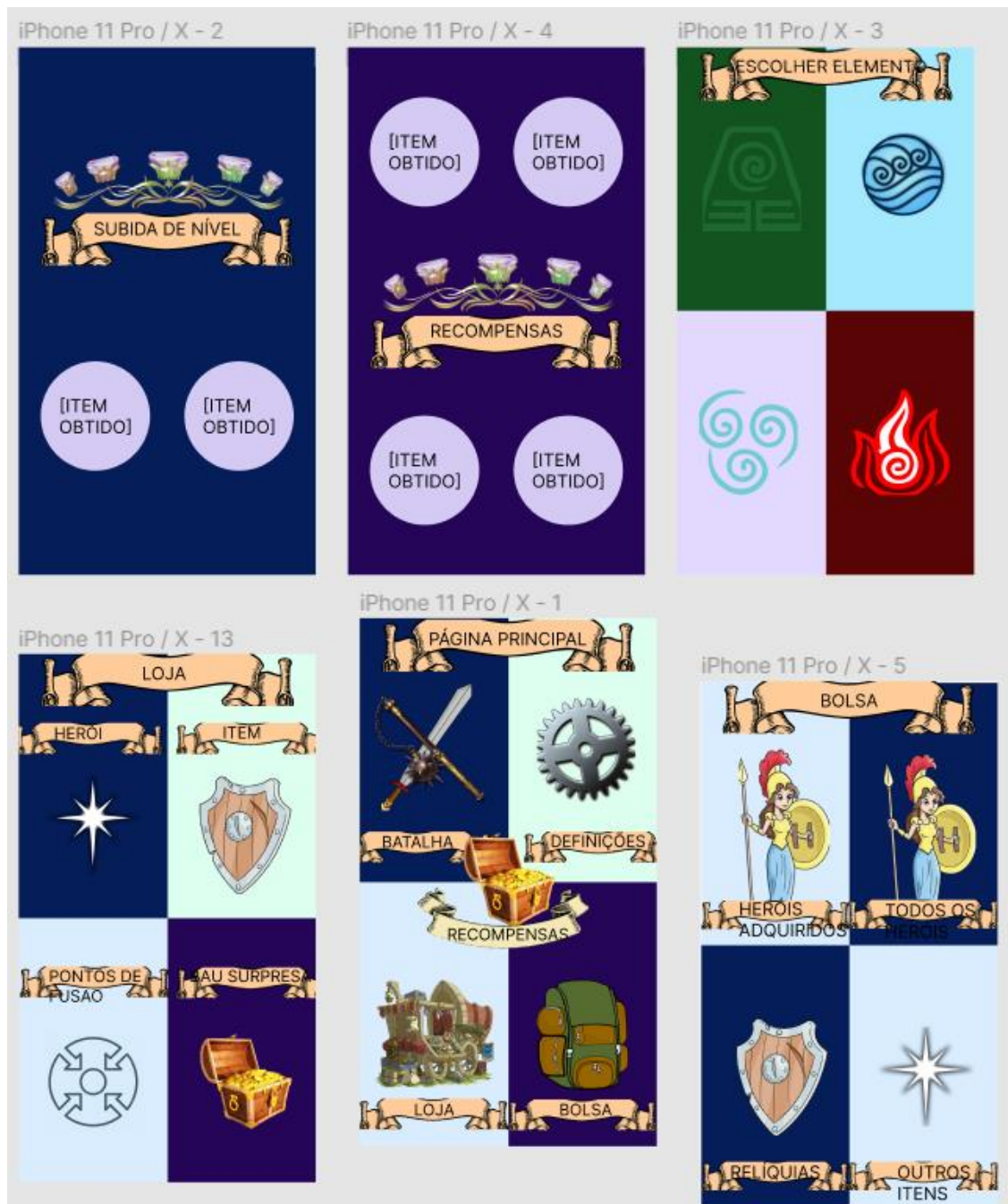
Requisitos Funcionais e Não Funcionais

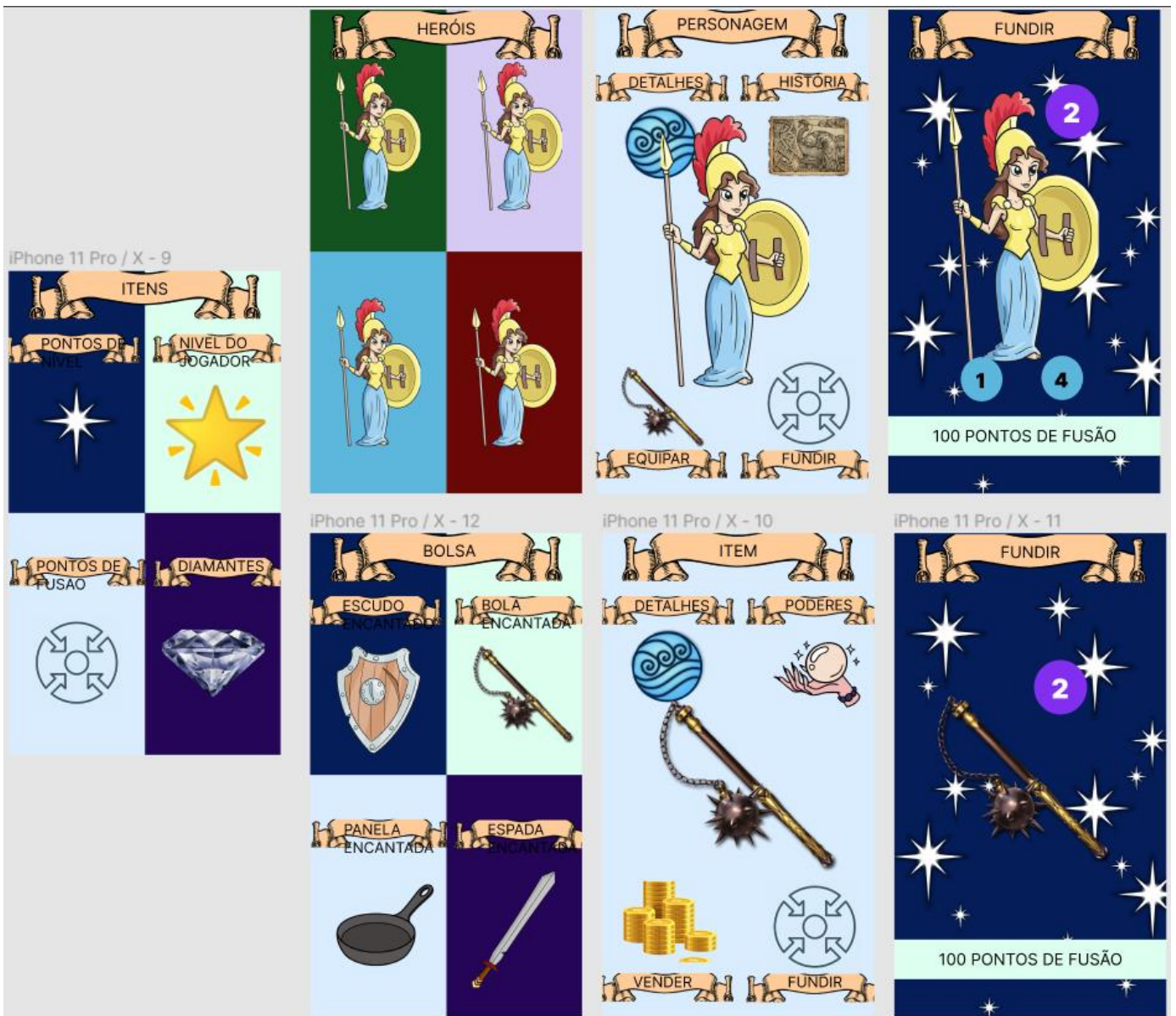
Requisito	Função (Descrição do requisito)	Categoria (Visível, Invisível, Adorno)	Atributo (Identificação do atributo)	Detalhe / Restrição Fronteira (Identificação dos detalhes e/ou valores do atributo)	Categoria (Obrigatório, Desejável)
R1.1	Vincular utilizador	Visível	Interação homem-máquina	Detalhe - Botões e caixas de diálogo	Obrigatório
			Usabilidade	Detalhe - Navegação pela aplicação por toque de ecrã	Obrigatório
			Tempo de resposta	Detalhe - curto, adequado	Obrigatório
			Plataforma	Detalhe - ANDROID	Obrigatório
R1.2	Autenticar utilizador	Invisível	Tempo de resposta	Restrição Fronteira - Máximo de 2 segundos	Desejável
R1.3	Ativar interação para utilizadores cegos ou amblíopes	Visível	Interação homem-máquina Tempo de resposta	Detalhe - Botões e caixas de diálogo Restrição Fronteira - Máximo de 2 segundos	Obrigatório Desejável
R1.4	Desativar interação para utilizadores cegos ou amblíopes	Visível			
R1.5	Alterar o volume do áudio leitor de texto	Visível			
R1.6	Alterar a velocidade do áudio do leitor de texto	Visível			
R1.7	Alterar o volume do áudio dos efeitos sonoros	Visível			
R2.1	Listar personagens adquiridas	Visível	Interação homem-máquina	Detalhe - Toque de ecrã	Obrigatório
R2.2	Listar itens adquiridos	Visível			
R2.3	Atribuir itens a uma personagem	Visível			
R2.4	Atualizar lista de personagens adquiridas	Invisível	Tempo de resposta	Detalhe - curto, adequado	Obrigatório
R2.5	Atualizar lista de itens adquiridos	Invisível			
R2.6	Atualizar nível do jogador	Invisível			
R2.7	Atualizar estatísticas da personagem	Invisível			

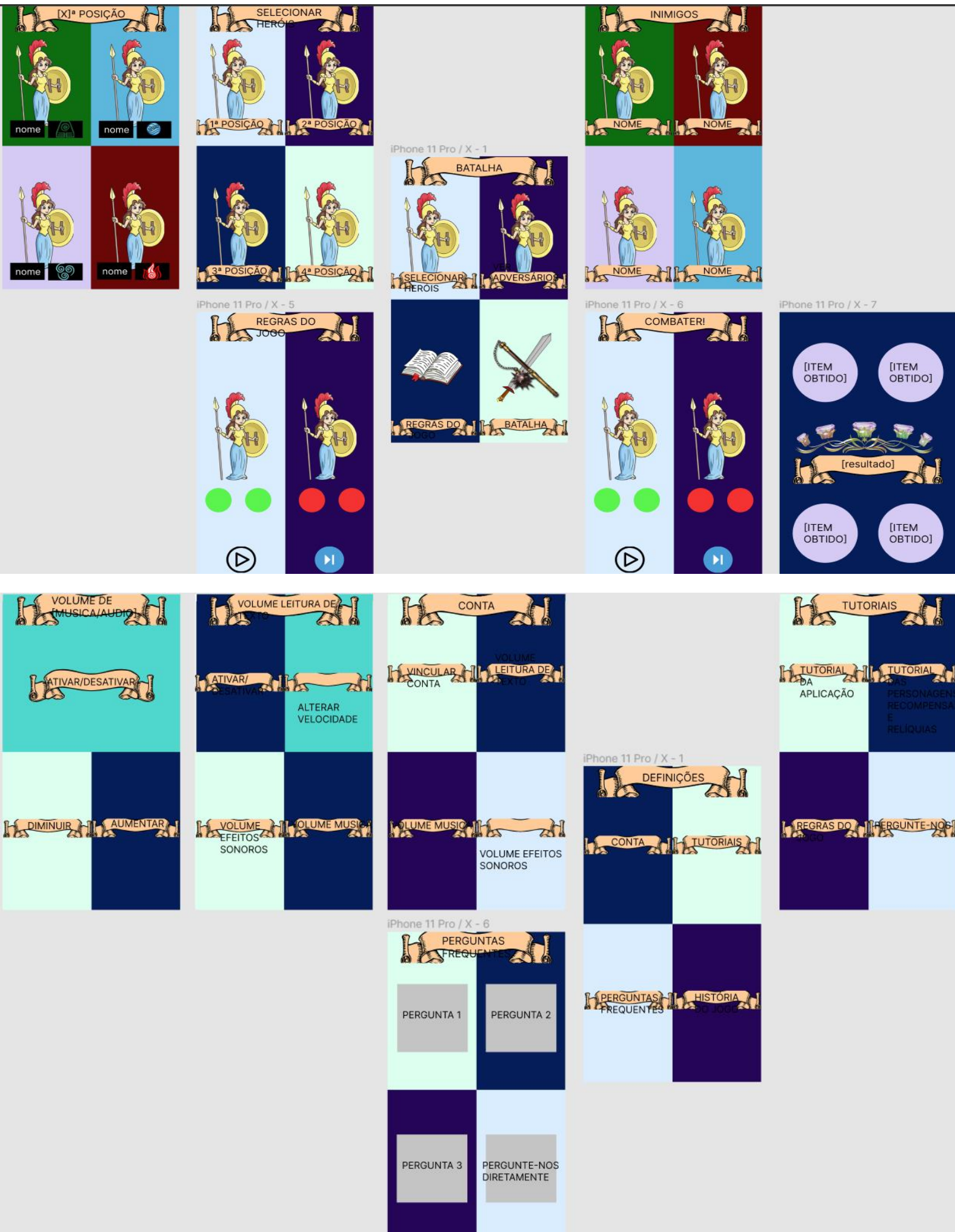
R3.1	tutorial auditivo das funcionalidades	Visível	Usabilidade Tempo de resposta	Detalhe - Descrições e notificações auditivas Detalhe - curto, adequado	Obrigatório Obrigatório
R3.2	tutorial auditivo dos jogos	Visível			
R3.3	Descrever auditivamente as personagens e itens	Visível			
R3.4	Descrever auditivamente as ações do utilizador na aplicação	Visível			
R4.1	Informar auditivamente qual a página aberta	Visível	Usabilidade Tempo de resposta	Detalhe - Descrições e notificações auditivas Restrição Fronteira - Máximo de 2 segundos	Obrigatório Desejável
R4.2	Sinalizar/Informar auditivamente o resultado do jogo	Visível			
R4.3	Sinalizar/Informar auditivamente os acontecimentos do jogo	Visível			
R4.4	Sinalizar/Informar auditivamente a subida de nível do jogador	Visível			
R4.5	Sinalizar/Informar auditivamente do aumento da defesa ou ofensa do personagem	Visível			
R4.6	Sinalizar/Informar auditivamente a subida de nível do personagem	Visível			
R5.1	Sinalizar/Informar através da vibração o resultado do jogo	Visível	Usabilidade Tempo de resposta	Detalhe - Descrições e notificações com vibração Restrição Fronteira - Máximo de 2 segundos	Obrigatório Desejável
R5.2	Sinalizar/Informar através da vibração os acontecimentos do jogo	Visível			
R5.3	Sinalizar/Informar através da vibração a subida de nível do jogador	Visível			
R5.4	Sinalizar/Informar através da vibração a subida de nível do personagem	Visível			
R6.1	Recolher recompensas	Visível	Usabilidade	Detalhe - Toque de ecrã	Obrigatório
R6.2	Cronometrar o aumento das recompensas	Invisível			
R6.3	Iniciar o jogo	Visível	Tempo de resposta	Restrição Fronteira - Máximo de 2 segundos	Obrigatório

Apêndice G

Mock-Ups da Aplicação de Jogo







Bibliografia

<https://flutter.dev/>

<https://www.interviewbit.com/blog/flutter-vs-android-studio/>

<https://hackr.io/blog/react-native-vs-flutter>

<https://docs.flutter.dev/development/tools/hot-reload>

<https://www.mongodb.com/nosql-explained/nosql-vs-sql>

<https://www.ibm.com/cloud/blog/sql-vs-nosql>

<https://www.educba.com/mysql-vs-sqlite/>

<https://www.ultraleap.com/company/news/blog/what-is-haptic-feedback/>

https://www.reddit.com/r/Blind/comments/eca4mv/a_list_of_blind_friendly_android_games/

<https://totenart.pt/blog/noticias/o-que-e-a-teoria-das-cores/>