

Revista Científica

FACULDADE ATENAS- PARACATU-MG

Ano 2024, V.17, N.1



FACULDADE
ATENAS

www.atenas.edu.br
38 3672-3737

APLICATIVO DE SUPORTE AO RPG DE MESA

Álvaro Guimarães Franco Neto
Alice Sodré Dos Santos
Jardel Rodrigues Marques De Lima
Thais Dias dos Santos
Cristiano André Peixoto

RESUMO

O *Role-Playing Game* (RPG) é um gênero de jogo presencial cooperativo que permite que os jogadores interpretem personagens e criem universos fictícios, combinando elementos da literatura e do teatro de forma interativa, apresentando um potencial pedagógico, estimulando habilidades intelectuais e sociais, como trabalho em equipe, cooperação, criatividade e interatividade. Este trabalho de conclusão de curso tem como objetivo o desenvolvimento de uma aplicação móvel para o gerenciamento de fichas de personagens e campanhas de RPG, utilizando as tecnologias Flutter e Firebase. O desenvolvimento do aplicativo se baseou em uma revisão bibliográfica sobre o tema e em uma pesquisa exploratória no mercado de aplicações móveis para o gênero. O aplicativo desenvolvido foca no sistema *Dungeons and Dragons*, oferecendo uma interface em português, melhorando a experiência do jogo para os jogadores brasileiros.

Palavras-chave: Interpretação. Literatura. RPG. Aplicativo. Leitura. Jogador.

ABSTRACT

The Role-Playing Game (RPG) is a genre of cooperative face-to-face game that allows players to interpret characters and create fictional universes, combining elements of literature and theater interactively, presenting a pedagogical potential, stimulating intellectual and social skills, such as teamwork, cooperation, creativity, and interactivity. This final course work aims at the development of a mobile application for the management of character sheets and RPG campaigns, using Flutter and Firebase technologies. The development of the application was based on a bibliographic review on the theme and an exploratory research in the mobile applications market for the genre. The developed application focuses on the Dungeons and Dragons system, offering an interface in Portuguese, improving the game experience for Brazilian players.

Keywords: Interpretation. Literature. RPG. Application. Reading. Player.

1 INTRODUÇÃO

O *Role-Playing Game* (RPG) é um *game* onde os jogadores interpretam papéis de personagens que tem como objetivo a criação de ficção. No game os jogadores possuem a liberdade de construir seu próprio universo, trabalhando a criatividade, desenvolvimento e cultura, combinando elementos da literatura e do teatro. A narrativa e a interpretação desempenham papéis vitais nesse *game*, onde todos os jogadores interagem e interpretam como se fizessem parte desse mundo. Um dos jogadores assume a responsabilidade pelo desenvolvimento da trama, enredo e outros aspectos do universo criado, atuando como mediador e catalisador da interação entre todos os participantes da partida (CUPERTINO, 2008).

O jogador responsável pela criação da narrativa é comumente chamado de mestre, e se baseia em um livro conhecido como sistema, que fornece orientações e regras para a construção de um universo rico e detalhado. Existem diversos tipos de sistemas, cada um com suas próprias regras e modos de jogar, incluindo o famoso *Dungeons and Dragons*, um livro que descreve uma aventura medieval com magias, dragões e guerreiros, sendo considerado um dos responsáveis pela popularização do gênero, oferecendo um orbe rico em conteúdo que serve de base para a construção do mundo pelo mestre (PAVÃO, 2000). Para Araújo e Fonseca (2005), o RPG é um formato que vai além de um jogo, sendo uma forma interativa de literatura que tem a capacidade de agir como um agente pedagógico no ensino e aprendizado através da interpretação de cenários históricos e locais distantes, desenvolvendo diversos aspectos intelectuais e sociais dos jogadores com práticas de trabalho em equipe, cooperação, criatividade e interatividade.

Com o avanço da pandemia, muitos jogadores tiveram que se adaptar à impossibilidade de se reunir presencialmente e migraram para plataformas de videoconferência, como *Skype*, *Microsoft Teams* e *Discord*, para continuar suas sessões de forma remota através de aplicativos que possibilitam o compartilhamento de fichas e campanhas online, como *Roll20* e *D&D Beyond*. (SCRIVEN, 2021). No Brasil, a utilização desses aplicativos é dificultada pela falta de localização em português e preços altos, o que se torna um entrave para os jogadores brasileiros. Dessa forma, este trabalho tem como objetivo desenvolver um aplicativo localizado, em português, gratuito e de código aberto que contribua para a melhoria da experiência de jogo tanto virtual quanto presencial, estimulando o desenvolvimento criativo dos praticantes e a criação de uma comunidade cada vez mais rica e

envolvente, oferecendo suporte para a criação de histórias cada vez mais surpreendentes e influentes na cultura.

2 ANÁLISE DOS IMPACTOS DO RPG NA SOCIEDADE

O Role-playing game é um jogo baseado na interpretação de papéis pelos jogadores, que desbravam um universo fictício criado por um dos jogadores, comumente chamado de narrador ou de mestre, que cria o seu universo baseando-se ou não em um sistema existente de regras, sendo o jogador responsável por narrar a história e agir como um mediador daquela mesa, e os outros jogadores atuam como personagens inseridos naquela mesa (PAVÃO, 2000). Uma das principais características do RPG de mesa é ser um jogo feito com lápis, papel, livro e uma mesa ao redor dos amigos, o que proporciona uma experiência mais intimista, acolhedora e imersiva entre os jogadores, contando com diversos benefícios sociais.

Para Cupertino (2008), o RPG tem suas raízes em jogos de mesa criado por militares, que tinham como propósito a solução de problemas de guerra utilizando da imaginação, e assim, o gênero nasceu da junção desses jogos militares com elementos fantasiosos das obras de J.R.R Tolkien. O primeiro exemplar de RPG como conhecemos hoje foi criado em meados da década de 70 nos estados pela editora Tactical Studio Review, pelos escritores Gary Gygax e Dave Arneson (FRAGA, 2021). Na década de 1980, o RPG chegou no Brasil através de intercambistas americanos que trouxeram consigo os primeiros exemplares desses livros-jogos, sendo um sucesso entre os jovens de todo país devido as características que juntam videogames, narrativas e a criação de suas próprias histórias ficcionais, sendo rapidamente propagando pelo brasil através de cópias de xerox, mais conhecimento como “cultura de xerox”, tendo sua primeira produção nacional lançada em 1991 (FILHO, 2018).

2.1 RPG NO MERCADO GLOBAL

A expectativa para o mercado de RPG de mesa é positiva, sendo um modelo de negócios em crescimento no mundo todo com uma estimativa de crescimento de 11,84% até 2030. Surpreendentemente, mesmo no ano da pandemia, a editora do maior RPG do mundo registrou um aumento impressionante de 33% em suas receitas esse dado indica que, mesmo diante dos desafios, os jogadores continuaram a

desfrutar do RPG de mesa, encontrando uma maneira de se manterem conectados com seus amigos mesmo durante o isolamento social (WHITTEN, 2021).

2.2 SISTEMAS DE REGRA DO JOGO

Os jogadores têm à sua disposição uma variedade de opções quando se trata de escolher um livro de regras para iniciar a construção de sua própria história no RPG, alguns exemplos conhecidos são GURPS, D&D e The Storyteller. Esses livros desempenham um papel fundamental como manuais de instruções, fornecendo orientações e material de apoio, como mundos fantásticos, raças, seres mitológicos, itens mágicos e equipamentos que podem ser utilizados pelo mestre na criação da história (PAVÃO, 2000). Além disso, de acordo com Araújo e Fonseca (2005), o sistema de regras também desempenha um papel importante ao garantir a equidade entre os jogadores, evitando ações absurdas durante a tomada de decisões pelos personagens e proporcionando um conteúdo equilibrado.

2.3 SISTEMA LIVRE

Segundo a *Open Source Initiative* (2018), o software de código aberto refere-se a um tipo de programa que o seu código fonte é exposto e aberto para que todos os usuários possam ler, modificar, copiar e distribuir para qualquer pessoa e finalidade, sendo principalmente um modelo de desenvolvimento colaborativo e público, portanto, a partir do código aberto, é possível permitir que a comunidade de RPG no mundo todo possa modificar o software e utilizar para a criação dos seus próprios sistemas de regras e mesas, sendo totalmente livre e aberto a customização.

3 ELABORAÇÃO DE ARQUITETURA DE SOFTWARE FUNDAMENTADA NO RPG *DUNGEONS AND DRAGONS*

O D&D (*Dungeons and Dragons*) é considerado o primeiro e mais popular RPG de mesa do mundo, criado por Gary Gygax e Dave Arneson na década de 70 que moldou diversos conceitos de fantasia da cultura popular e inspirou diversos jogos, séries e filmes ao longo desses anos. Esse sistema é ambientado em um mundo de fantasia medieval e os jogadores criam seus próprios personagens escolhendo entre diversas raças e classes, e participam de aventuras narradas pelo mestre, que controla os

elementos do cenário, os inimigos, os aliados e os desafios que enfrentam (PAVÃO, 2000).

Os livros principais apresentam conteúdo base diverso para criação de campanhas, bem como a explicação para criação de personagem, diversas classes, raças, itens lendários, magias, reinos e personagens históricos que o mestre pode utilizar para enriquecer a sua história. O livro discorre sobre a forma de se jogar, apresentando as regras para todos os jogadores, que são livres para narrar e tomar decisões do ponto de vista do seu personagem, mas o resultado destas é narrado pelo mestre, que pode exigir o uso de dados multifacetários como os da figura X para determinar o resultado consequente, refletindo em um forte aspecto de imprevisibilidade e aleatoriedade (COAST, 2014).

3.1 O PAPEL DOS JOGADORES

No sistema de D&D, os jogadores são encarregados de criar os seus próprios personagens, escrevendo suas características em uma folha de papel, como nome, equipamentos, atributos, personalidade e outros aspectos do personagem, como pode ser visto na figura 1. Com base nessa representação da ficha de papel, é possível observar quais campos o nosso aplicativo deve suprir para cumprir o requisito de permitir que os jogadores escrevam as suas fichas de personagem (COAST, 2014).

FIGURA 1 - Modelo de ficha de personagem

3.2 REQUISITOS DE SOFTWARE

Requisitos de software são as especificações que definem o que um software deve fazer e como deve fazer. Eles são importantes para garantir que o software atenda às necessidades e expectativas dos clientes e usuários, bem como para facilitar o desenvolvimento, teste e manutenção do software (UML, 2005). Os requisitos funcionais são caracterizados como especialmente como as regras de negócio do nosso software, são as partes que devem cumprir exatamente como o essencial para que o sistema funcione de maneira adequada.

QUADRO 1 - Requisitos funcionais do aplicativo

Requisito	Descrição
1	Permitir que os usuários se cadastrem com e-mail, senha e nome de usuário.
2	Permitir que os usuários acessem suas próprias contas com endereço de e-mail e senha.
3	Permitir que o usuário crie uma ficha de personagem.
4	Permitir que o usuário customize sua ficha de personagem.
5	Permitir que o usuário adicione equipamentos da ficha.
6	Permitir que o usuário remova equipamentos da ficha.
7	Permitir que o usuário adicione magias da ficha.
8	Permitir que o usuário remova magias da ficha.
9	Permitir que o usuário remova uma ficha.
10	Permitir que o usuário crie uma campanha.
11	Permitir que o usuário adicione notas a uma campanha.
12	Permitir que o usuário convide outros usuários a uma campanha.
13	Permitir que o usuário aceite um convite para uma campanha.
14	Permitir que o usuário negue um convite para uma campanha.
15	Não permitir que o usuário convide outro usuário se já há algum convite pendente.
16	Permitir que o usuário delete uma campanha.

Fonte: de autoria própria (2023)

Os requisitos não funcionais são implementações não visíveis aos usuários, mas são importantes de um ponto de vista macro. São os requisitos que não são funcionalidades do aplicativo, mas servem como base para completar tais funcionalidades.

QUADRO 2 - Requisitos não funcionais do aplicativo

Requisito	Descrição
1	O aplicativo deve ser instalável no sistema operacional Android
2	O aplicativo deve ser acessível pelo navegador em dispositivos móveis
3	O aplicativo deve ter uma UI/UX de fácil utilização
4	O aplicativo deve atualizar o conteúdo em tempo real
5	O usuário deve ser capaz de acessar através de diferentes dispositivos ao mesmo tempo.
6	O aplicativo deve estar em português.
7	Deve ser possível modificar, criar e deletar fichas mesmo sem internet.

Fonte: de autoria própria (2023)

3.3 ARQUITETURA DE SOFTWARE

A arquitetura é um conceito abstrato que define o planejamento inicial e a tomada de decisões sobre as estruturas mais básicas, características, componentes, relacionamentos e a forma com que o sistema se comunica com outros sistemas de forma geral, com o objetivo de reduzir o esforço na construção e manutenções de sistemas (MARTIN, 2019). Sendo um aspecto importante para garantirmos a manutenibilidade e existência futura do nosso projeto mesmo com adição ou remoção de novas funcionalidades. A arquitetura de software pode ser representada por meio de diagramas UML que auxiliam na construção do software.

3.4 UML

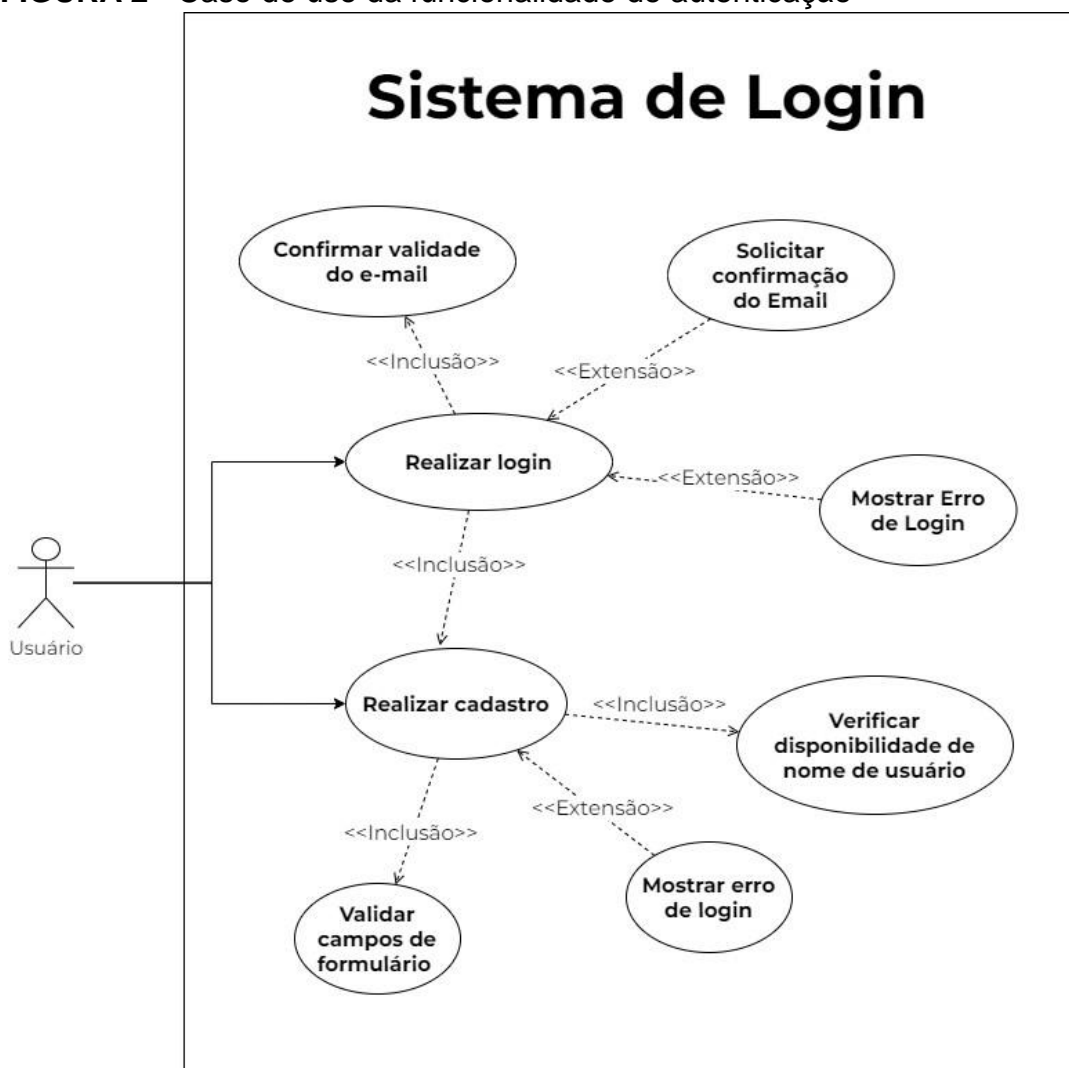
A UML é uma linguagem de notação que define padrões e especificações para a representação visual da arquitetura e design de software, sendo uma ferramenta essencial para lidar com a complexidade de aplicações em larga escala, permitindo que se modele os aspectos principais e essenciais do software sem se preocupar com os detalhes de implementação, com a capacidade de representar o software antes mesmo dele ser construído e portanto, sendo uma ferramenta agnóstica às metodologias de desenvolvimento (UML, 2005).

3.5 CASOS DE USO

A UML é uma linguagem de notação que define padrões e especificações para a representação visual da arquitetura e design de software, sendo uma ferramenta essencial para lidar com a complexidade de aplicações em larga escala, permitindo que se modele os aspectos principais e essenciais do software sem se preocupar com os detalhes de implementação, com a capacidade de representar o software antes mesmo dele ser construído e portanto, sendo uma ferramenta agnóstica às metodologias de desenvolvimento (UML, 2005).

A Figura 2 mostra o diagrama UML que representa os requisitos funcionais de autenticação do aplicativo, exemplificando os processos necessários para se cadastrar e entrar na aplicação. O diagrama utiliza os símbolos de ator, caso de uso, associação e extensão para ilustrar as interações entre o usuário e o sistema.

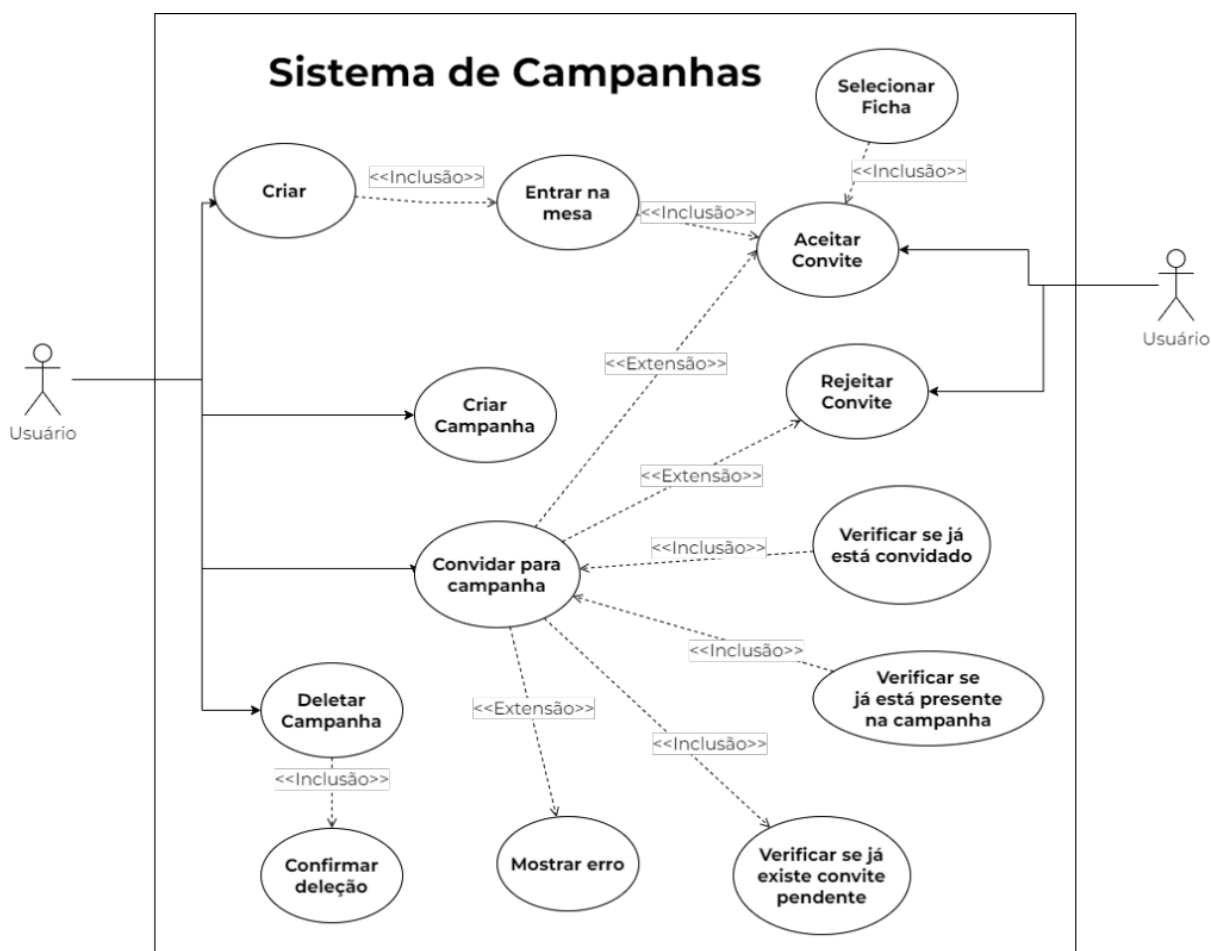
FIGURA 2 - Caso de uso da funcionalidade de autenticação



Fonte: elaborado pelo autor (2023).

A Figura 3 mostra o diagrama que representa o fluxo que os usuários percorrem ao criar campanhas e convites:

FIGURA 3 - Caso de uso da funcionalidade campanhas



Fonte: elaborado pelo autor (2023).

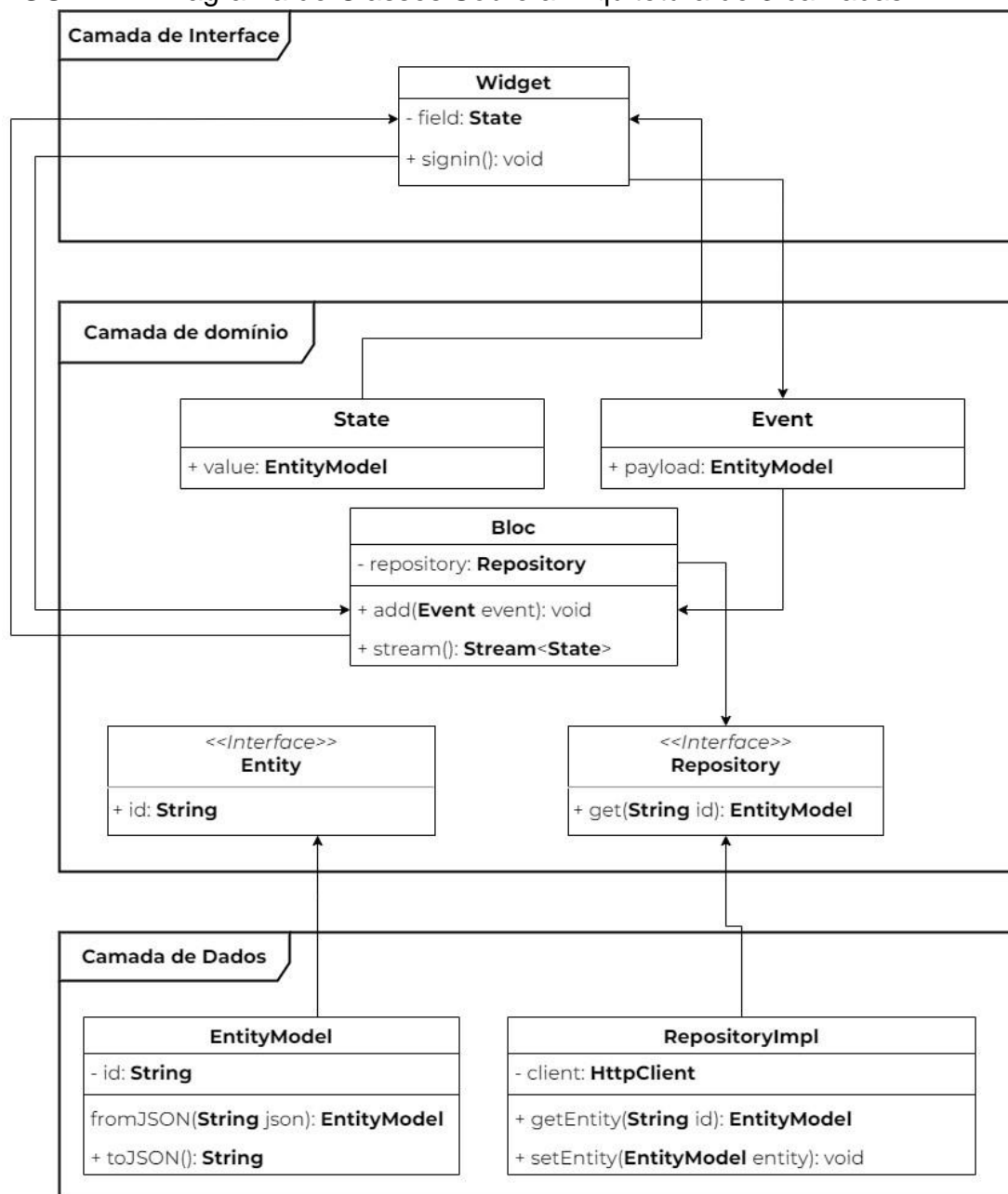
3.6 DIAGRAMA DE CLASSES

Os Diagramas de classe também fazem parte da UML, e tem o propósito muito importante em um software orientado a objeto, pois permite mapear o relacionamento entre as diversas classes do sistema operacional e planejar uma arquitetura de antemão (UML, 2005).

Para criar um sistema com baixo acoplamento, Martin (2019) sugere criar classes do sistema com a divisão em três camadas bem definidas de domínio, recursos e apresentação, desacoplando as regras de negócio, a busca pelos dados no banco de dados e a apresentação desses dados na interface. Como pode ser visto na figura 2, a camada de domínio é responsável pelas regras de negócio da aplicação, sendo independente das outras camadas, a camada de recursos e serialização é responsável

pela busca e escrita dos dados nos servidores, e por fim, a camada de interface ou visualização é onde o usuário interage com o aplicativo e os dados são exibidos.

FIGURA 4 - Diagrama de Classes Sobre a Arquitetura de 3 camadas



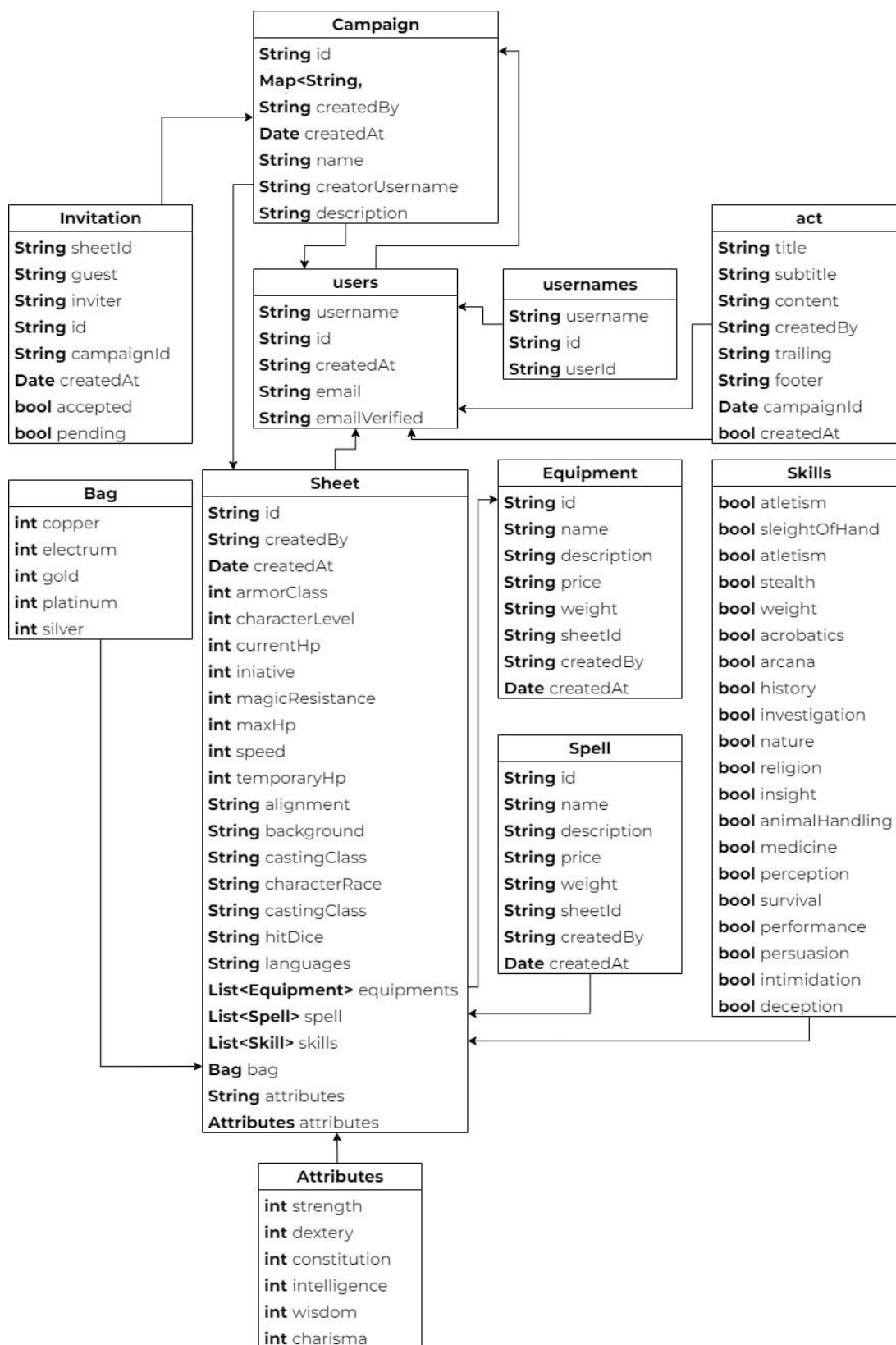
Fonte: elaborado pelo autor (2023).

3.7 MODELAGEM DOS DADOS

De acordo com Sikha Bagui e Richard Walsh (2003), a modelagem de dados é um processo de representação das entidades, que podem ser qualquer objeto físico ou lógico, geralmente denominado com um substantivo, e os seus relacionamentos com outras entidades, que são representadas pela utilização de um verbo que interliga essas duas entidades. A Figura 3 ilustra um diagrama UML que mostra os modelos de

dados da aplicação proposta, que contém os dados da ficha do jogador, os dados necessários para se criar uma campanha e as entidades auxiliares.

FIGURA 5 - Entidades da aplicação e seus relacionamentos



Fonte: elaborado pelo autor (2023).

4 DESENVOLVIMENTO DE UMA APLICAÇÃO

4.1 BANCO DE DADOS

Um banco de dados é uma coleção de dados correlacionados que pode ser armazenada em um arquivo ou em um cluster dedicado, no formato estruturado ou semiestruturado. O acesso de leitura e escrita do banco é feito através de um software chamado DBMS, que é responsável pela comunicação do banco de dados com os usuários e aplicativos externos (CARVALHO, 2015) .

4.2 FIREBASE

O Firebase é uma plataforma de BaaS (*backend-as-service*) criado em 2012 e adquirido pela google 2014 que fornece uma solução em infraestrutura que facilita construir, manter e aprimorar aplicações de maneira escalável, oferecendo vários produtos e soluções para cada etapa do desenvolvimento de uma aplicação, como banco de dados, armazenamento e autenticação, sendo compatível com diversas linguagens e softwares do mercado como Javascript, Java, Unity, Flutter (KUMAR, 2018).

4.3 LINGUAGEM DE PROGRAMAÇÃO

As linguagens de programação são um conjunto de símbolos que nos permitem dar instruções para o computador, sendo organizadas pelo seu nível de abstração e complexidade, que determina o grau de proximidade com a linguagem humana ou com a linguagem de máquina. A linguagem de máquina é a forma mais primitiva de comunicação, realizada por meio de números binários ou hexadecimais que dão instruções diretas em código de máquina, um nível acima estão as linguagens de baixo nível que são mais fáceis de se utilizar, pois permitem uma comunicação mais natural com a máquina a partir de instruções de formato mnemônico, além destas, existem as linguagens de alto nível, que são construídas em cima da linguagem humana que abstraem diversos procedimentos complexos de um computador, tornando o desenvolvimento mais agradável (MANZANO e OLIVEIRA, 2019).

4.4 DART

O Dart é uma linguagem de programação de código aberto e multiplataforma de

propósito geral que nasceu na conferência de GOTO em 2011 e foi lançada oficialmente em 2013, desenhada pelos desenvolvedores Lars Bak e Kasperr Lund e desenvolvida pela Google com objetivo de interfaces para usuários sendo inicialmente uma linguagem que iria substituir o Javascript nos navegadores (BAK, 2011). Uma das características mais marcantes da linguagem é a sua extrema versatilidade, sendo uma linguagem que pode ser tanto compilada quanto interpretada (DART, 2023).

4.5 FLUTTER

À medida que o desenvolvimento de aplicações se tornou mais complexo, surgiu a necessidade de reutilização de código a partir da utilização de frameworks que abstraem e implementam aspectos comuns do software, permitindo a criação de software escalável e de fácil manutenção, possibilitando se concentrar nas funcionalidades específicas do sistema.

O Flutter é um framework de código aberto anunciado pela Google em 2015 e lançado oficialmente em 2018 com suporte para as plataformas Android e IOS, permitindo o desenvolvimento de aplicações híbridas com a compilação para código nativo destas plataformas, possibilitando a utilização de uma base de código única para a criação de diferentes binários de aplicativo. O framework adota uma arquitetura baseada em widgets e na construção de interfaces baseadas em uma hierarquia, o que torna o desenvolvimento flexível e customizável, e inclui uma biblioteca de widgets prontos implementados a partir do design system Material UI (GOOGLE, 2018).

4.6 MATERIAL UI

Para Couldwell (2019), um *design System* é uma abordagem que visa padronizar as decisões de design, usando conjuntos de componentes reutilizáveis que promovem uma linguagem de design comum coesa e consistente para a criação de um ou mais produtos, criando uma experiência uniforme para os usuários através da padronização dos elementos interativos e de interface, focando na documentação das decisões e prevenindo a necessidade de reinventar a roda na construção de novos produtos.

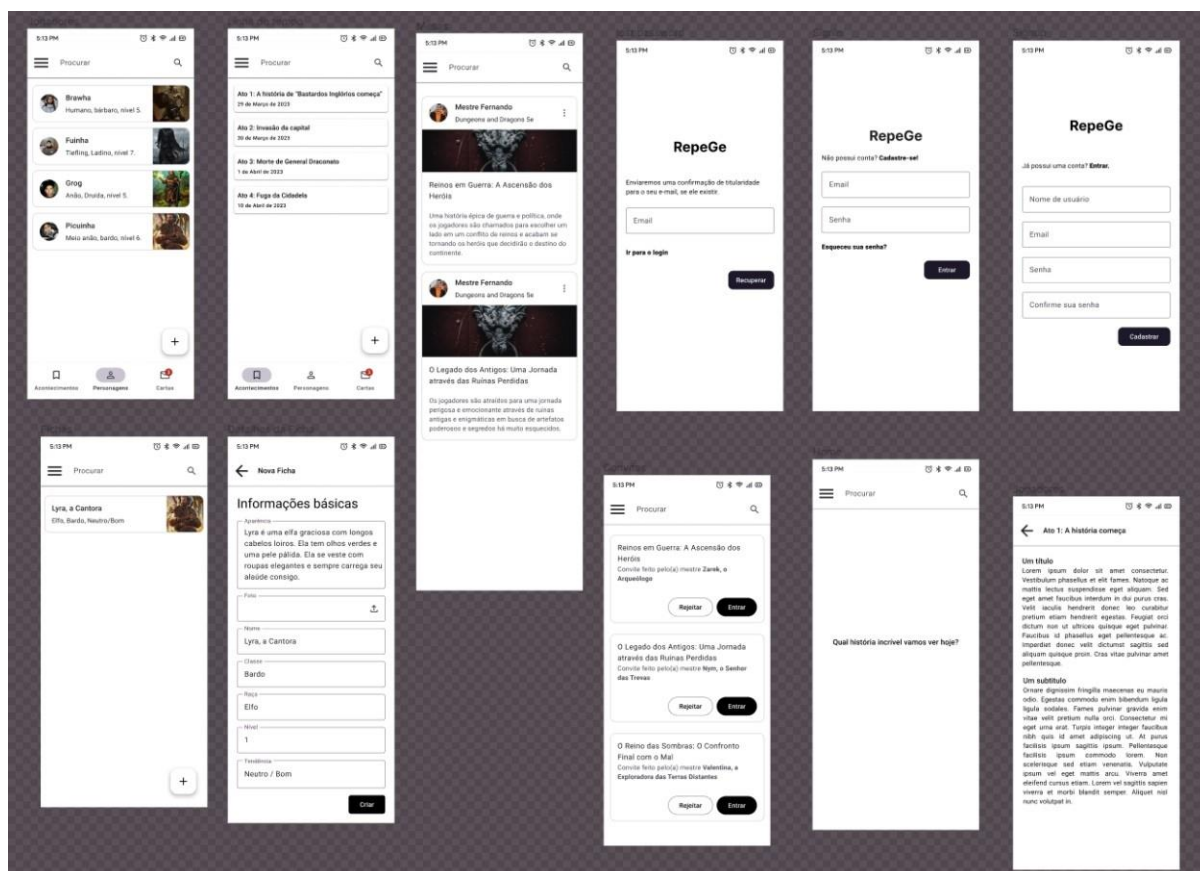
Originalmente chamado de *Quantum Paper*, o material UI é um *design system* desenvolvido pela Google em 2014, que se inspira nas propriedades físicas do papel e da tinta, empregando grades e sombras para criar uma experiência de usuário atraente e intuitiva (SPRADLIN, 2014).

4.7 PROTOTIPAÇÃO

De acordo com Arnowitz, Arent e Berger (2006), A prototipação é uma técnica de desenvolvimento de software que visa criar modelos ou softwares de baixo custo para validar ideias e corrigir aspectos importantes antes da implementação de um produto real, para Morgan (2006), a prototipação não somente é uma forma de validação, mas uma tentativa de representar os aspectos do software como se fosse construído com o atual conhecimento da equipe de desenvolvimento, garantindo que os usuários finais possam dar avaliações que melhorem o software antes do lançamento da versão final.

O Figma é uma ferramenta de manipulação de interfaces e *wireframes* criada para desenvolvedores e designers que fornece recursos de desenho, navegação e prototipação de interfaces de aplicativos para diferentes tipos de tamanhos de telas, além de permitir o rápido compartilhamento de uma versão interativa do aplicativo para os usuários finais (STAIANO, 2022).

FIGURA 6 - Conceitos iniciais para o aplicativo no Figma



Fonte: elaborado pelo autor (2023).

4.8 ANDROID

O Android é um sistema operacional de código aberto, baseado no núcleo Linux, desenvolvido pela Google e destinado a dispositivos móveis, como smartphones e tablets com telas sensíveis ao toque, no qual os usuários interagem com o SO por meio de gestos como tocar, deslizar, pinçar e pressionar a interface do dispositivo, sendo capaz de oferecer suporte a outros tipos de entradas como voz, teclado e mouse. A primeira versão estável do sistema Android foi lançada em 2008 pela Open Handset Alliance, consórcio que define padrões para a criação de uma plataforma de dispositivos móveis gratuita e aberta (HAASE, 2022).

Segundo Reads (2022), em uma pesquisa feita pela StockApps, foi constatado que o Android é o sistema operacional móvel mais utilizado no mundo, representando cerca de 70% do mercado mundial de dispositivos móveis, o que favorece a criação de uma versão dedicada a plataforma para a utilização do aplicativo por meio de dispositivos móveis.

4.9 WEB

De acordo com Almeida (2016), a *World Wide Web*, popularmente conhecida como web, é um sistema de informação global que permite o acesso a diversos tipos de conteúdo, como textos, imagens, vídeos, áudios e dados, funcionando por meio de um sistema de endereços chamados de URLs, que identificam os websites e recursos disponíveis na rede, que são acessados por meio de navegadores, possibilitando a interação dos usuários com esses recursos por meio de formulários, links, botões e outros elementos. Os websites são construídos em cima de três linguagens com propósitos distintos: O HTML é a linguagem de marcação que organiza o conteúdo em elementos hierárquicos e semânticos; A apresentação e aparência do conteúdo é definido pela linguagem CSS, que personaliza os elementos na tela; por fim, o comportamento e interatividade é definido pela linguagem de programação Javascript, que adiciona dinamismo e a interação com o usuário.

5 CONSIDERAÇÕES FINAIS

O presente trabalho teve como principal objetivo o desenvolvimento de um aplicativo destinado ao gerenciamento de campanhas e fichas para jogadores de RPG de mesa, explorando a aplicabilidade do RPG em diversos contextos. Para alcançar

esse propósito, realizou-se uma pesquisa bibliográfica sobre o gênero RPG, que proporcionou uma base sólida para a realização deste projeto, sendo possível concluir que o aplicativo desenvolvido atende de forma eficaz à sua proposta, cumprindo com os objetivos específicos. A partir do aplicativo, os usuários podem inserir e editar as suas informações da ficha do jogador, incluindo a possibilidade de acessar seus dados de diferentes dispositivos, graças ao armazenamento em nuvem, que permite uma maior flexibilidade e organização, reduzindo a necessidade de utilização de lápis e papel durante o jogo. Além disso, os jogadores podem ingressar em grupos que permitem o compartilhamento em tempo real das suas informações com os outros participantes, o que auxilia no gerenciamento do jogo remotamente.

O sistema incorpora uma funcionalidade de autenticação que possibilita a utilização consistente por parte dos usuários em várias plataformas diferentes, implementando uma interface simples e intuitiva, que permite uma experiência consistente, sendo acessível e gratuito.

É importante observar que o aplicativo apresenta margem para aprimoramentos em vários aspectos, visando à sua aplicabilidade a diferentes tipos de sistemas, e melhoria da experiência do usuário.

5.1 TRABALHOS FUTUROS

Há margem para melhoria no aplicativo, como a adição de novas funcionalidades que melhorariam a experiência de uso dos jogadores, como a possibilidade de conversar em tempo real com os participantes de uma campanha, tornando mais fluída a narração dos acontecimentos e a interação dos jogadores, e a possibilidade de anexar arquivos de imagem e vídeos nas fichas e nas campanhas, permitindo ilustrar os personagens, os cenários, os objetos e eventos do jogo.

REFERÊNCIAS

ARAÚJO, L. E. D.; FONSECA, M. R. D. **R.P.G no ensino de literatura infanto juvenil nas séries iniciais**. Centro Universitário de Brasília. Brasília, p. 85. 2005.

BAGUI, S.; EARP, R. **Database Design Using Entity-Relationship Diagrams**. [S.l.]: Auerbach Publications, 2003.

BAK, L. Opening Keynote: Dart, a new programming language for structured web programming. **Google For Developers**, 2011. Disponível em: <<https://developers.googleblog.com/2011/10/dart-language-for-structured-web.html>>. Acesso em: 13 outubro 2023.

- CARVALHO, V. **MySQL: Comece com o principal banco de dados open source do mercado.** [S.l.]: Casa do Código, 2015. 165 p.
- COAST, W. O. T. **Dungeons & Dragons - Player's Handbook.** 5. ed. [S.l.]: Wizards of the Coast, v. I, 2014.
- COULDWELL, A. **Laying the foundations.** [S.l.]: [s.n.], 2019.
- CUPERTINO, E. R. **Vamos jogar RPG? Diálogos com a literatura, o leitor e autoria.** Universidade de São Paulo. São Paulo, p. 132. 2008.
- DART. Dart. **Dart Programming Language,** 2023. Disponível em: <<https://dart.dev/>>. Acesso em: 13 Outubro 2023.
- FILHO, C. B. P. Uma análise da história dos RPGs (RolePlaying Games) de mesa brasileiros. **Proceedings of SBGames,** Balneário Camboriú, 29 Outubro 2018. 1-4.
- FONSECA, J. J. S. D. **METODOLOGIA DA PESQUISA CIENTÍFICA.** Universidade Federal do Ceará. Ceará, p. 127. 2002.
- FRAGA, A. M. **Gênese do mercado de livros de RPG no Brasil: as traduções dos livrosjogos de Steve Jackson e Ian Livingstone pela editora Marques Saraiva.** Universidade de São Paulo. São Paulo, p. 18. 2021.
- GOOGLE. Ready for Production Apps: Flutter Beta 3. **Google Developers,** 2018. Disponível em: <<https://developers.googleblog.com/2018/05/ready-for-production-apps-flutter-beta-3.html>>. Acesso em: 24 outubro 2023.
- HAASE, C. **Androids: The Team That Built the Android Operating System.** [S.l.]: No Starch Press, 2022.
- JONATHAN ARNOWITZ, M. A. N. B. **Effective Prototyping for Software Makers (Interactive Technologies).** [S.l.]: Morgan Kaufmann, 2006.
- KUMAR, A. **Mastering Firebase for Android Development.** Birmingham: Packt, 2018.
- MANZANO, J. A. N. G.; OLIVEIRA, J. F. D. **Algoritmos: Lógica Para Desenvolvimento de Programação de Computadores.** 29. ed. Editora ÉRICA: Érica, v. I, 2019.
- MARTIN, R. C. **Arquitetura limpa: o guia do artesão para estrutura e design de software.**
Tradução de Samantha Batista. Rio de Janeiro: Alta Books, v. 1, 2019. 432 p. ISBN 97885-508-1600-5. Acesso em: 12 Outubro 2023.
- MOLINARI, W. **Desconstruindo a Web.** São Paulo: Casa do Código, 2016.
- NOVA, S. P. D. C. C. et al. **Trabalho de Conclusão de Curso - Uma abordagem leve, Divertida e Prática.** São Paulo: Saraiva, 2020. Acesso em: 11 Abril 2023.
- OPEN SOURCE INITIATIVE. History of Open Source Initiative. **Open Source Initiative,** 2018. Disponível em: <<https://opensource.org/history/>>. Acesso em: 13 Outubro 2023.

PAVÃO, A. **A AVENTURA DA LEITURA E DA ESCRITA ENTRE MESTRES DE ROLEPLAYING GAMES (RPG)**. Associação Nacional de Pós-Graduação e Pesquisa em Educação. Caxambu, p. 1-16. 2000.

READS, E. M. Android loses 8% of its global OS market share in five years. **StockApps**, 2022. Disponível em: <<https://stockapps.com/blog/android-loses-8-of-its-global-os-market-share-infive-years/>>. Acesso em: 2023 Outubro 24.

RODRIGUES, M. P. D. O. **O RPG DE MESA COMO LABORATÓRIO FICCIONAL NA LITERATURA CONTEMPORÂNEA E O ESTATUTO DO NARRADOR**. PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS. GOIÂNIA, p. 169. 2021.

SCRIVEN, P. From Tabletop to Screen: Playing Dungeons and Dragons during COVID-19. **Societies**, Australia, 9 Outubro 2021.

SPRADLIN, L. Exclusive: Quantum Paper And Google's Upcoming Effort To Make Consistent

UI Simple. **Android Police**, 2014. Disponível em: SPRADLIN, L. Exclusive: Quantum Paper And Google's Upcoming Effort To Make Consistent UI Simple. **Android Police**, 2014. Disponível em:

<<https://www.androidpolice.com/2014/06/11/exclusive-quantum-paper-and-googlesupcoming-effort-to-make-consistent-ui-simple/>>. Acesso em: 2023 Outubro 24.

STAIANO, F. **Designing and Prototyping Interfaces with Figma**: Learn essential UX/UI design principles by creating interactive prototypes for mobile, tablet, and desktop. [S.l.]: Packt Publishing, 2022.

TIERNEY, P. Dungeons & Dragons: What's Better, A Paper Or Digital Character Sheet? **THEGAMER**, 2020. Disponível em: <<https://www.thegamer.com/dungeons-dragons-papervs-digital-character-sheet/>>. Acesso em: 31 Outubro 2023.

UML. UML. **What is UML | Unified Modeling Language**, 2005. Disponível em: <<https://www.uml.org/what-is-uml.htm>>. Acesso em: 18 Outubro 2023.

WHITTEN, S. Dungeons & Dragons had its biggest year ever as Covid forced the game off tables and onto the web. **Consumer News and Business Channel**, 2021. Disponível em: <<https://www.cnbc.com/2021/03/13/dungeons-dragons-had-its-biggest-year-despite-the-coronavirus.html>>. Acesso em: 22 Março 2023.

GLOSSÁRIO

Executáveis: são arquivos que contêm instruções para serem executadas por um sistema operacional ou por um programa específico

Framework: Um framework é uma ferramenta que facilita o desenvolvimento de software, fornecendo uma estrutura pré-definida de códigos, componentes, bibliotecas e padrões que podem ser reutilizados e adaptados pelo programador.

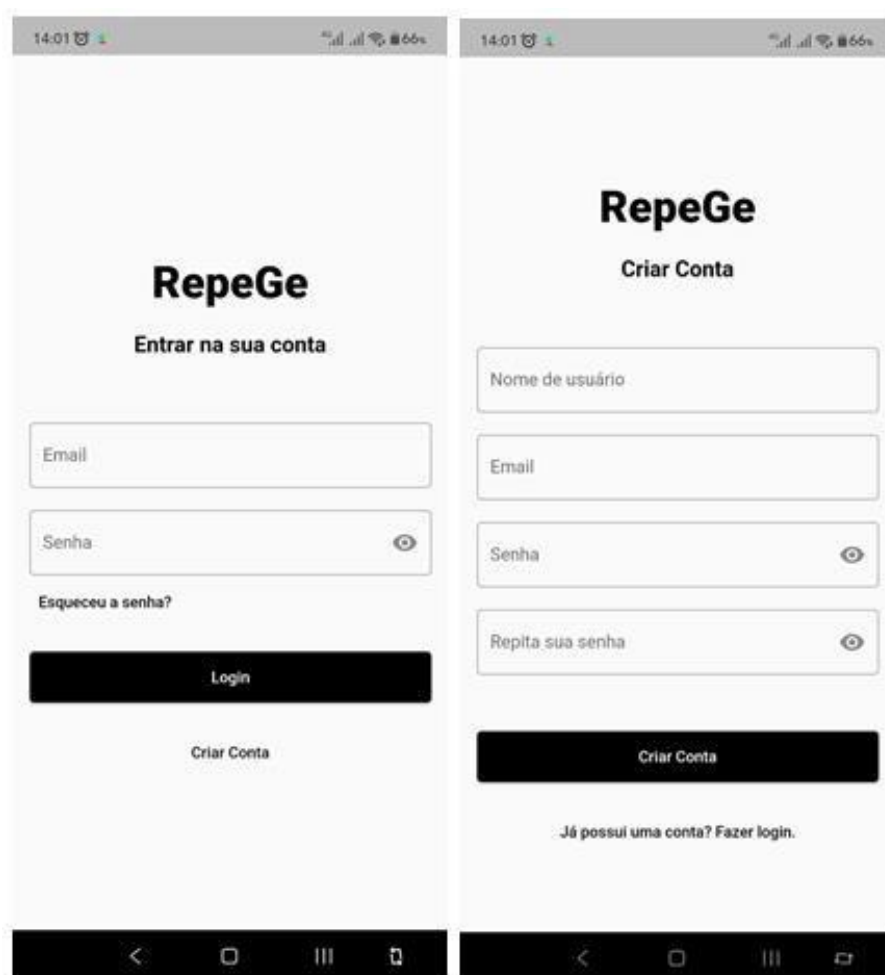
Linguagem Compilada: Linguagem compilada é uma linguagem de programação em que o código fonte é traduzido para uma linguagem de baixo nível, como linguagem de montagem ou código de máquina, antes da execução do programa.

Linguagem de interpretada: Linguagem interpretada é uma linguagem de programação em que o código fonte não é traduzido diretamente para uma linguagem de baixo nível, mas sim executado por um programa chamado interpretador, que lê e executa cada comando em tempo real.

SQL: é uma linguagem de domínio específico usada para gerenciar dados armazenados em um sistema de gerenciamento de banco de dados relacional ou em um sistema de gerenciamento de fluxo de dados relacional.

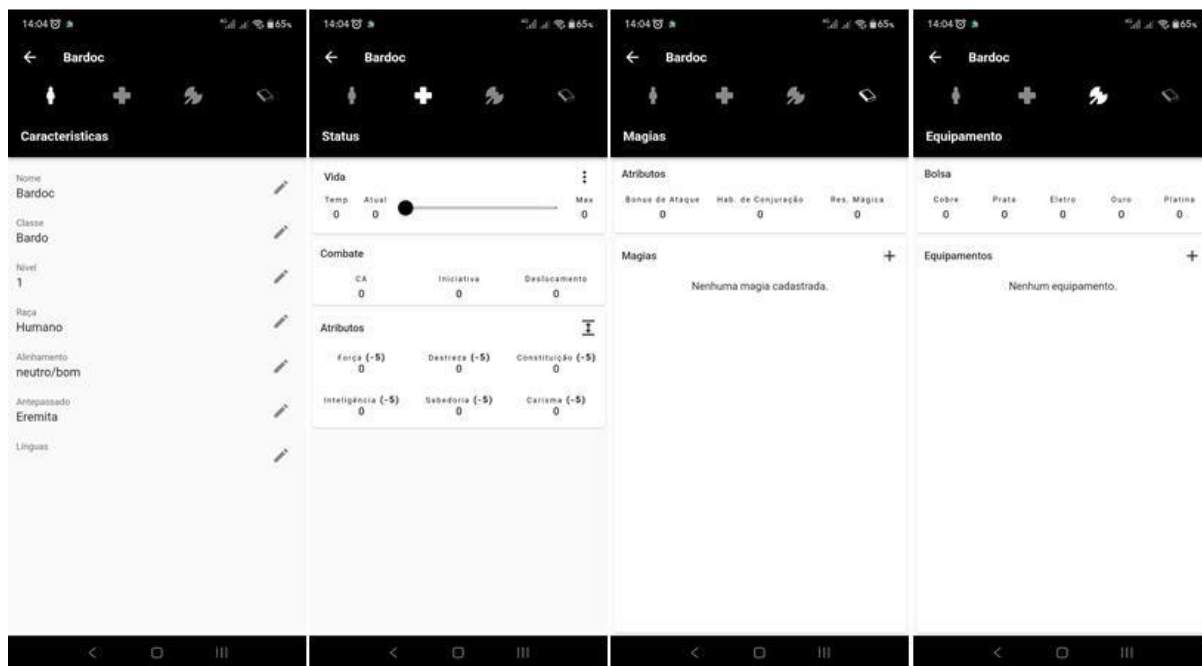
APÊNDICE A – INTERFACES DO APLICATIVO

FIGURA 7 - Telas de autenticação



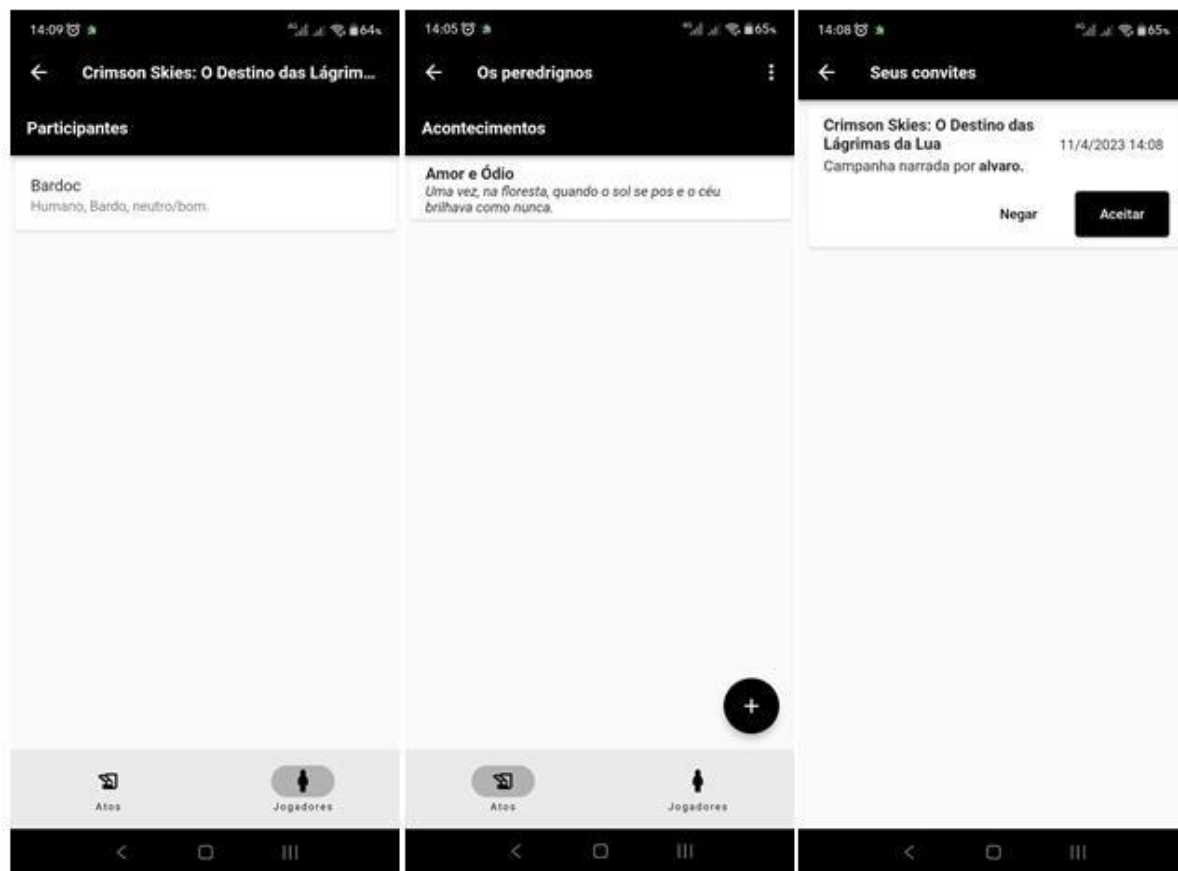
Fonte: elaborado pelo autor (2023).

FIGURA 8 - Telas de ficha de personagem



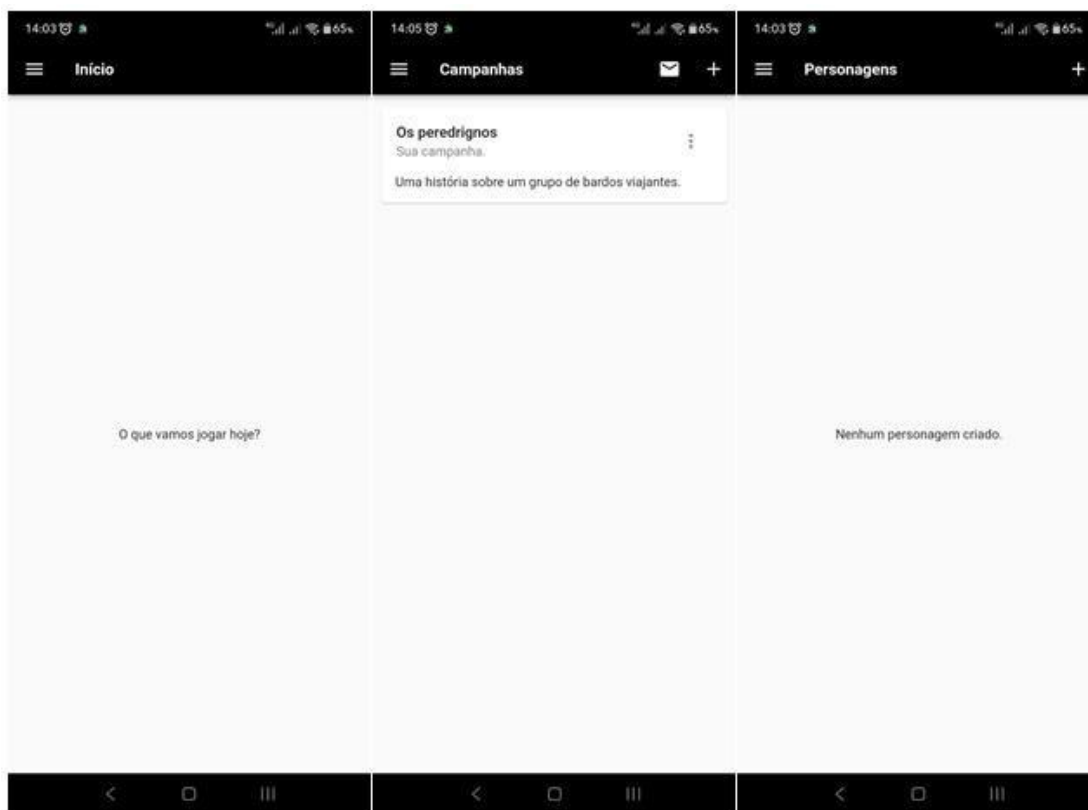
Fonte: elaborado pelo autor (2023).

FIGURA 9 - Telas de gerenciamento de campanhas



Fonte: elaborado pelo autor (2023).

FIGURA 10 - Telas de criação de personagem e campanha



Fonte: elaborado pelo autor (2023).