**DES - Documento de Engenharia de Software**

Versão: 0.1

24 de Fevereiro de 2017

**Projeto Dungeons & Dragons**

<NomeAutor>

<NomeAutor>

<NomeAutor>

EC205 - AulaLab 4 - Documento Engenharia de Software.docx

Tabela de Revisões

| Versão | Principais Autores | Descrição da Versão | Data de Término | Aprovação e data | |
| --- | --- | --- | --- | --- | --- |
| V0.1 | <NomeAutor>  <NomeAutor>  <NomeAutor> | Introdução, Escopo e Descrição de Funcionamento. | xx/xx/xxxx | **Daniel Izario** | xx/xx/xxxx |
| V0.2 | <NomeAutor>  <NomeAutor>  <NomeAutor> | Requisitos Funcionais, Diagrama de Casos de uso, Fluxo de Eventos e Requisitos Não Funcionais. | xx/xx/xxxx | **Daniel Izario** | xx/xx/xxxx |
| ... | ... | ... | ... | **...** | ... |

Índice

Tabela de Revisões 2

Índice 3

1. Lista de Figuras 4

2. Lista de Tabelas 5

3. Introdução 6

3.1 Definições, Acrônimos e Abreviaturas 6

4. Visão geral 7

4.1 Introdução 7

4.2 Escopo 7

4.3 Descrição de funcionamento 7

5. Especificação de Requisitos 9

5.1 Requisitos Funcionais 9

5.1.1 Req.1 - Efetuar o cadastro dos clientes de acesso 9

5.1.2 Req.2 - Exibir o relatório de backups 9

5.2 Diagrama de Casos de Uso 10

5.2.1 Descrição dos Atores 10

5.2.2 Descrição dos Casos de Uso 10

5.3 Fluxos de Eventos de Casos de Uso 11

5.3.1 Login do Administrador 11

5.4 Requisitos Não-Funcionais 12

5.4.1 Req.91 - Utilizar Windows como sistema operacional 12

5.4.2 Requisitos de Desempenho 12

5.4.2.1 Req.92 - O tempo da geração de relatório não deve exceder 1 segundo. 12

6. Projeto de Dados 13

6.1 Modelo Entidade-Relacionamento 13

7. Projeto Lógico 14

7.1 Diagrama de Classes 14

7.2 Diagrama de Sequência 14

7.3 Diagrama de Pacotes 15

7.4 Diagrama de Atividade 15

8. Anexos 16

8.1 Storyboarding 16

8.2 Estrutura Analítica do Projeto - EAP 16

8.3 Cronograma de Atividades 17

9. Bibliografias de Texto 18

10. Bibliografia de Imagens 19

# Lista de Figuras

**Figura 1 -** Exemplo da implementação. 7

**Figura 2 -** Diagrama do <Projeto>. 8

**Figura 3 -** Fluxograma do <Projeto>. 8

**Figura 4 -** Diagrama de casos de uso. 10

**Figura 5 -** Modelo Entidade-Relacionamento. 13

**Figura 6 -** Diagrama de classes. 14

**Figura 7 -** Diagrama de sequência. 14

**Figura 8 -** Diagrama de Pacotes. 15

**Figura 9 -** Diagrama de Atividades. 15

**Figura 10 -** Telas do Software. 16

**Figura 11 -** EAP. 15

**Figura 12 -** Cronograma. 16

# Lista de Tabelas

**Tabela 01 -** Requisito Req.1. 9

**Tabela 02 -** Requisito Req.2. 9

# Introdução

## Definições, Acrônimos e Abreviaturas

**Backup -** Nomenclatura que significa a cópia de dados para um meio seguro onde possa ser restaurado.

**UML -** Sigla que representa a padronização utilizada para a linguagem de modelagem.

**Software Cliente** **-** Software que inicia uma comunicação com um Servidor de dados.

**Rede TCP/IP** **-** Meio que permite a transmissão de dados entre computadores.

**Banco de dados Remoto -** Servidor de acesso externo para armazenamento de dados.

# Visão geral

## Introdução

Dungeons & Dragons é um RPG (*Role Playing Game*) de fantasia medieval de mesa, no qual os jogadores se reúnem e interpretam personagens de diversas classes, orientados pelo *Dungeon Master*, que é o responsável por guiar os jogadores e criar as aventuras que serão jogadas. Diferente dos conhecidos RPGs virtuais (como os famosos World of Warcraft, Skyrim e Final Fantasy), que utilizam computadores para tornar o jogo possível, Dungeons & Dragons faz o uso de fichas de personagem, dados e diversos livros de regras (os principais são: Livro do Jogador, Livro dos Monstros e Livro do Mestre), sendo um dos pioneiros neste estilo, inspirando, até hoje, jogos dessa geração. Então, com o objetivo de otimizar e facilitar a dinâmica do jogo, a equipe se propõe a criar um aplicativo para Android que possibilite a digitalização das ferramentas utilizadas pelo RPG, como as fichas, dados e principais regras. O jogo pode ser visto na Figura 1.



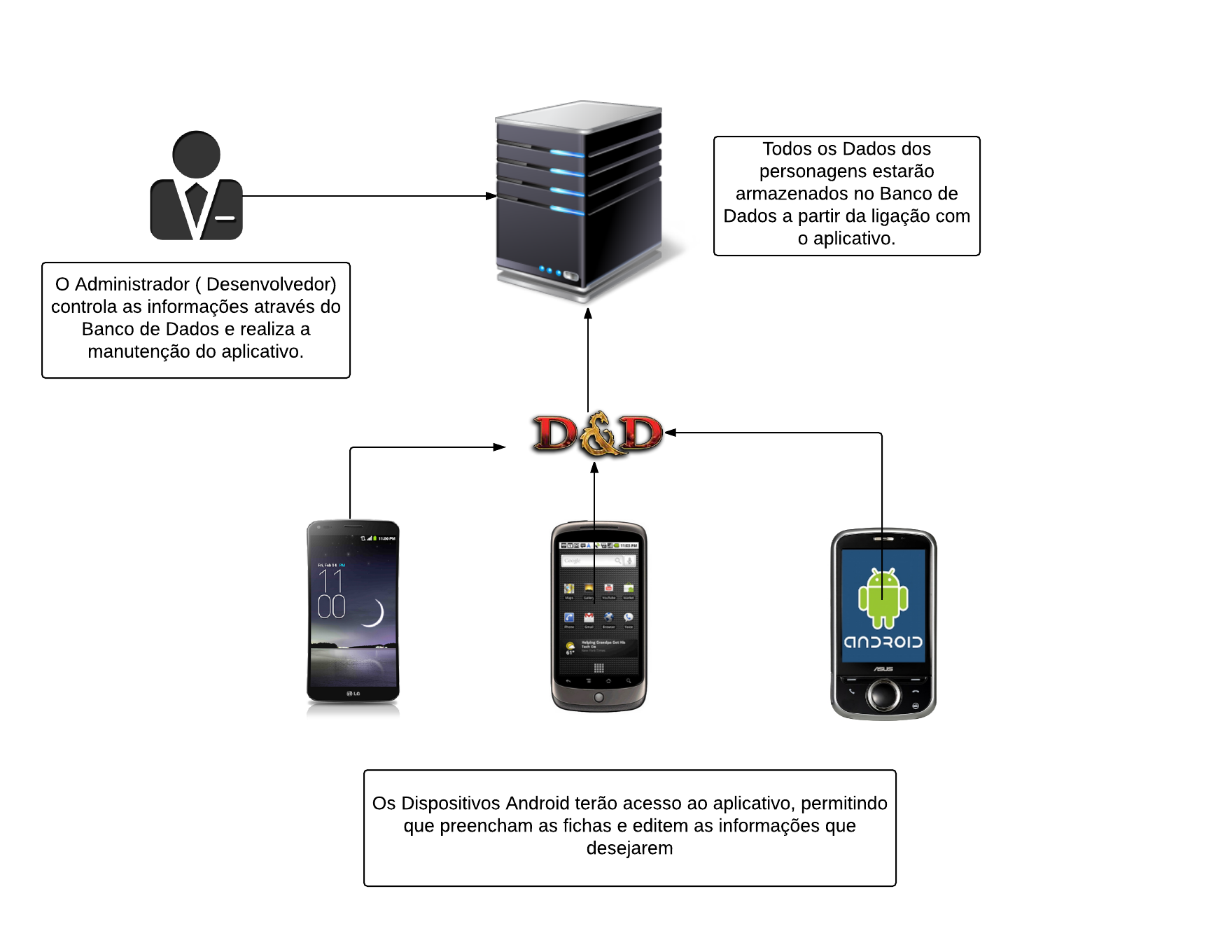
**Figura** 1 – Pessoas jogando Dungeons & Dragons

## Escopo

Este projeto consiste em desenvolver um aplicativo para a plataforma Android que facilite o processo de jogo do Dungeons & Dragons (versão 5.0), possibilitando a criação e edição de fichas de personagem, jogar os diversos tipos de dados utilizados pelo sistema do jogo e consultar regras principais e mais importantes.

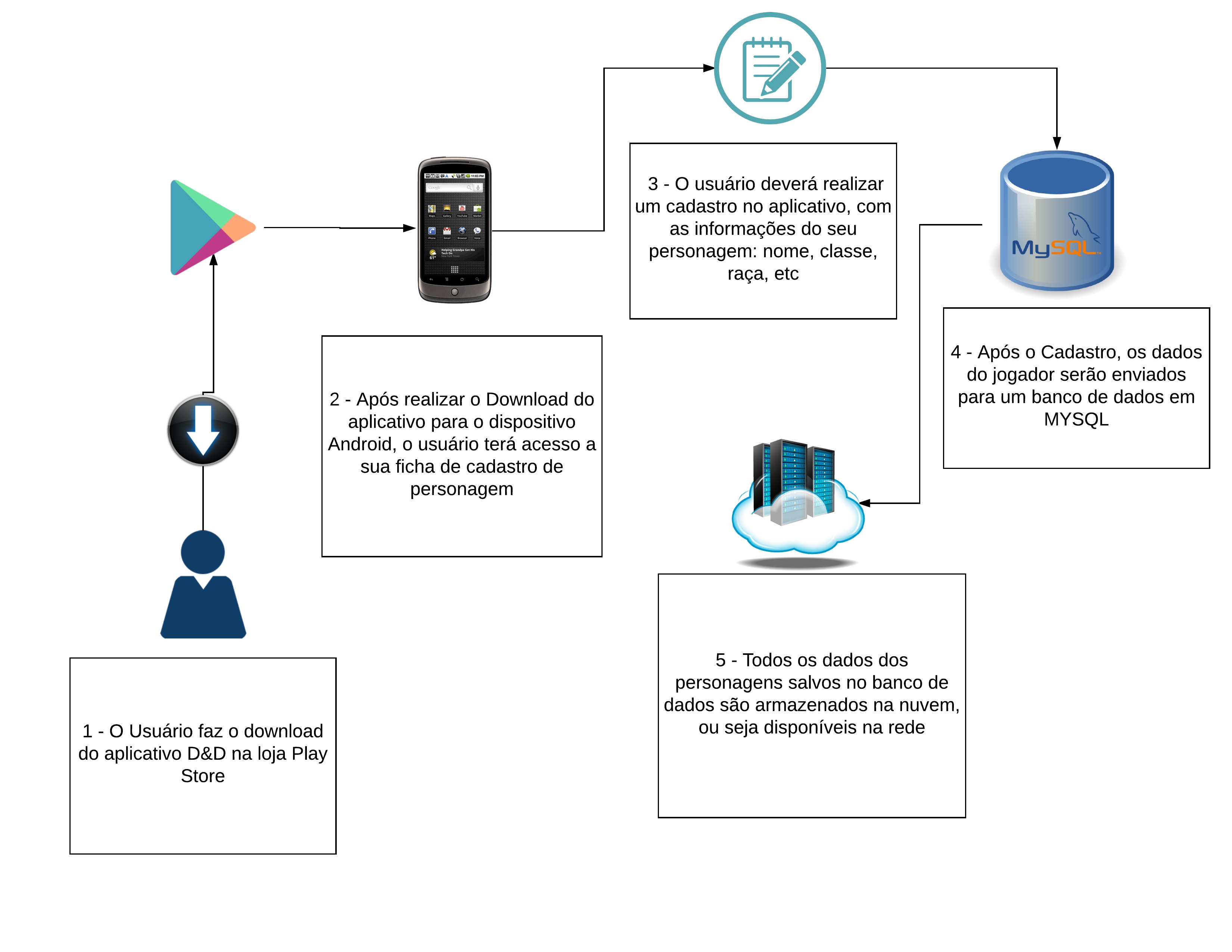
Na função de criação de fichas dos jogadores, o jogador terá que preencher um formulário com os principais dados do seu personagem, como seu nome, inventário, classes, nível, pontos de vida, magias disponíveis, atributos e todas as outras características utilizadas pela ficha de personagem. Todas as fichas serão armazenadas em um banco de dados, possibilitando que qualquer ficha criada seja selecionada em uma lista e editada, exibida ou excluída do aplicativo. Essas características podem ser atualizadas a qualquer momento, permitindo que o personagem tenha seus pontos de experiência, habilidades, inventário e os outros dados modificados ao decorrer da aventura.

O aplicativo também terá uma função para realizar a jogada dos diversos dados utilizados pelo RPG. Os dados existentes no jogo são: 4 faces (D4), 6 faces (D6), 8 faces (D8), 10 faces (D10), 12 faces (D12) e 20 faces (D20). Eles serão implementados no aplicativo mostrando um número aleatório limitado pelo dado utilizado pelo usuário.

As regras dos jogos utilizadas mais frequentemente serão apresentadas no aplicativo, facilitando o acesso dos jogadores durante o jogo para evitar que a constante consulta aos livros de regras seja necessária.

## Descrição de funcionamento

O aplicativo será programado em Java, utilizando a IDE específica do Android, o Android Studio. Ela apresentará na barra de ferramentas as três funções principais do aplicativo: regras, fichas de personagem e dados. Cada funcionalidade terá uma classe que irá definir seus principais métodos e atributos (alguns herdados das classes já existentes no Android, como, por exemplo, Activity e Fragment). Para que as fichas possam ser salvas, será utilizado banco de dados (MySQL) que possibilita que os dados sejam armazenados no dispositivo, possibilitando a edição, remoção, criação e exclusão de qualquer personagem criado pelo usuário.



....

....

.... e segue o fluxograma abaixo:

**Figura** 1 **-** Fluxograma do <Projeto>.

# Especiicação de Requisitos

## Requisitos Funcionais

### Req.1 - Efetuar o cadastro dos clientes de acesso

|  |  |
| --- | --- |
| **Detalhamento** | Cada cliente deve possuir os seguintes dados:  - Nome;  - IP.  ... |
| **Observação** | O CPF deve ser validado para efetuar o cadastro do cliente. Deve somente haver o cadastro, caso todas as informações citadas acima forem preenchidas, ou seja, não deve haver ausência de informação em nenhum campo do cadastro. |
| **Prioridade** | Alta. |

**Tabela 0**1 **-** Requisito Req.1.

### Req.1 - Exibir o relatório de backups

|  |  |
| --- | --- |
| **Detalhamento** | O sistema deve prover meios de exibir um relatório de backups contento as seguintes informações:  - IP;  - Quantidade de backups desde o início do cadastro;  - Data do último backup. |
| **Observação** | O relatório será exibido somente para os usuários cadastrados no sistema. |
| **Prioridade** | Alta. |

**Tabela 0**1 **-** Requisito Req.2.

## Diagrama de Casos de Uso



**Figura** 2 **-** Diagrama de casos de uso.

### Descrição dos Atores

**A1 - Administrador**

O Administrador tem acesso à as funcionalidades de Manter Viagens, Reservar Viagem, Manter Clientes, Manter Funcionários.

### Descrição dos Casos de Uso

**CaU1 - Manter Clientes**

Este caso de uso tem como objetivo manipular os dados dos clientes no banco de dados. Ela é composta pelas funcionalidades de cadastrar, listar, editar e excluir clientes. Somente o Administrador tem acesso a este caso de uso.

## Fluxos de Eventos de Casos de Uso

### Login do Administrador

|  |  |  |
| --- | --- | --- |
| **Nome da Use Case** | Login do **Administrador**. | |
| **Descrição** | Responsável pelo Login no software. | |
| **Requisitos associados** | Login. | |
| **Pré-condições** | Sistema tem que estar ligado. | |
| Possuir um Login. | |
| **Pós-condições** | Login certo. | |
| Login errado. | |
| **Atores** | **Administrador** e **Banco de Dados**. | |
| **Fluxo Principal** | | |
| **Ações** **Realizadas** | | **Ações Recebidas** |
| 1 - O **Administrador** deseja fazer o Login. | | 2 - O sistema solicita o Login. |
| 3 - O **Administrador** digita o Login. | | 4 - O sistema verifica o Login. |
| 5 - Se Login estiver certo, entra no software. |
| **Fluxo Alternativo** | | |
| **Ações** **Realizadas** | | **Ações Recebidas** |
| 1 - Sistema apresenta mensagem que o Login está errado. | | 3 - Usuário deseja tentar novamente. |
| 2 - Sistema pergunta se deseja efetuar Login novamente ou cancelar. | |
| 4 - Sistema solicita o Login. | | 5 - Usuário solicita cancelar operação. |
| 6 - Caso de uso encerrado. | |

**Tabela 3 -** Fluxo de evento principal < Login do Administrador >.

## Requisitos Não-Funcionais

### Req.91 - Utilizar Windows como sistema operacional

Será utilizada uma distribuição Windows ...

### Requisitos de Desempenho

#### Req.91 - O tempo da geração de relatório não deve exceder 1 segundo.

Este requisito deve ser validado somente se o sistema estiver ocioso e possua um número de registro menor que um milhão de clientes.

# Projeto de Dados

## Modelo Entidade-Relacionamento



**Figura** 3 **-** Modelo Entidade-Relacionamento.

# Projeto Lógico

## Diagrama de Classes



**Figura** 4 **-** Diagrama de Classes.

## Diagrama de Sequência



**Figura** 5 **-** Diagrama de Sequência.

## Diagrama de Pacotes



**Figura** 6 **-** Diagrama de Pacotes.

## Diagrama de Atividade



**Figura** 7 **-** Diagrama de Atividades.

# Anexos

## Storyboarding



**Figura** 8 **-** Telas do Software.

## Estrutura Analítica do Projeto - EAP



**Figura 11 -** EAP.

## Cronograma de Atividades



**Figura 12 -** Cronograma.

# Bibliografias de Texto

# Bibliografia de Imagens