

UNIVERSIDADE FEDERAL DE SANTA CATARINA

CENTRO TECNOLÓGICO

DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA

POKÉMON STADIUM

ESPECIFICAÇÃO DE REQUISITOS DE SOFTWARE

VERSÃO 1.0.0

FELIPE DI BERNARDI S THIAGO

FILIPE VOGES

HIAGO MEDEIROS MENDES

FLORIANÓPOLIS

2016

Sumário

[1. Introdução](#_30j0zll)

[1.1. Objetivo do desenvolvimento](#_1fob9te)

[1.2. Definições e abreviaturas](#_3znysh7)

[1.3. Referências](#_2et92p0)

[1.3.1. Regras do jogo:](#_tyjcwt)

[2.Visão geral do sistema](#_3dy6vkm)

[2.1. Arquitetura do sistema](#_1t3h5sf)

[2.2. Arquitetura da aplicação](#_4d34og8)

[2.3. Premissas de desenvolvimento](#_2s8eyo1)

[2.3.1. Interface gráfica](#_17dp8vu)

[2.3.2. Implementação](#_3rdcrjn)

[3. Requisitos da aplicação](#_26in1rg)

[3.1. Requisitos funcionais](#_lnxbz9)

[3.1.1. Iniciar preparação da partida](#_35nkun2)

[3.1.2. Selecionar Pokémon](#_1ksv4uv)

[3.1.3. Atribuir nível ao Pokémon](#_44sinio)

[3.1.4. Evoluir Pokémon](#_2jxsxqh)

[3.1.5. Calcular atributos](#_z337ya)

[3.1.6. Selecionar Move set](#_3j2qqm3)

[3.1.7. Iniciar partida](#_1y810tw)

[3.1.8. Verificar turno: novo Pokémon em batalha](#_4i7ojhp)

[3.1.9. Atacar](#_2xcytpi)

[3.1.10. Substituir Pokémon](#_1ci93xb)

[3.1.11. Verificar estado dos Pokémon](#_3whwml4)

[3.2. Requisitos não funcionais](#_2bn6wsx)

[3.2.1. Linguagem de Programação](#_qsh70q)

[3.2.2. Servidor de Jogos e Framework de rede](#_3as4poj)

[3.2.3. Especificação de projeto](#_1pxezwc)

[3.2.4. Interface gráfica para usuário](#_49x2ik5)

# 1. Introdução

## 1.1. Objetivo do desenvolvimento

O objetivo deste desenvolvimento é a criação de um jogo baseado em turnos para dois jogadores, que através do acesso a um servidor, pode ser jogado de qualquer lugar, com o intuito promover entretenimento e interação entre os usuários.

## 1.2. Definições e abreviaturas

Pokémon: Criaturas que são utilizadas nas batalhas entre jogadores.

Move set: Conjunto de ataques que um Pokémon possui.

HP: Pontos de vida do Pokémon.

## 1.3. Referências

### 1.3.1. Regras do jogo:

Na base do jogo existe uma lista com os Pokémon possíveis para escolha;

Os jogadores selecionam, cada um, três Pokémon, que terão nível e atributos aleatórios, e escolhem seu move set de acordo com o nível que foi atribuído ao Pokémon;

Alguns Pokémon possuem um estágio de evolução, que possui como requisito um nível mínimo (exemplo: para um determinado Pokémon evoluir ele precisa estar, no mínimo, no nível 16);

Os jogadores se enfrentam através desses Pokémon, um contra um;

A batalha acontece através de turnos, em seus respectivos turnos os jogadores escolhem uma ação, que pode ou não ser realizada pelo Pokémon, e passam para o turno do adversário;

Cada jogador escolhe um dos seus Pokémon para batalhar; durante seu turno, o jogador pode escolher atacar o Pokémon do adversário ou substituir seu Pokémon por um Pokémon que não esteja em estado de desmaio; a defesa não é uma ação direta, pode ser feita por ataques passivos que aumentam seu atributo de defesa, logo reduzindo o dano recebido ou por ataques que criam uma barreira que bloqueia o ataque do Pokémon adversário;

Quando um Pokémon perde todos os seus pontos de vida (HP) ele fica em estado de desmaio e obrigatoriamente deve ser substituído por um Pokémon que não esteja no estado de desmaio;

Um Pokémon desmaiado fica fora de combate durante o resto da partida corrente;

Caso o jogador não possua mais Pokémon em estado diferente de desmaio, ele perde o jogo.

# 2.Visão geral do sistema

## 2.1. Premissas de desenvolvimento

### 2.1.1. Interface gráfica

O programa deve apresentar uma interface gráfica bidimensional.

### 2.1.2. Implementação

O programa deve ser implementado em Java, sendo executado em qualquer plataforma que disponha de uma máquina virtual Java (JVM).

### 2.1.3. NetGamesServer

O programa deve ser distribuído utilizando NetGamesServer.

# 3. Requisitos da aplicação

## 3.1. Requisitos funcionais

### 3.1.1. Iniciar preparação da partida

Operação: Iniciação da preparação da partida.

Descrição: Verifica se há dois jogadores prontos para iniciar uma partida e, caso haja, inicia a seleção de Pokémon para ambos.

Lê: Estado dos jogadores.

Muda: Estado dos jogadores; estado da partida.

Envia: Alerta avisando jogadores do início da seleção de Pokémon.

Resulta: Partida em estado inicial, jogadores alocados, porém sem nenhum Pokémon.

### 3.1.2. Selecionar Pokémon

Operação: Seleção de Pokémon

Descrição: Apresenta uma lista de Pokémon para o jogador; jogador seleciona 3 Pokémon.

Lê: Lista de todos os Pokémon disponíveis na aplicação.

Muda: Lista de Pokémon do jogador; estado do jogador.

Envia:

Resulta: Partida em estado inicial, jogadores alocados, com 3 Pokémon em seu estado inicial.

### 3.1.3. Atribuir nível ao Pokémon

Operação: Atribuição de nível ao Pokémon.

Descrição: Atribui um nível aleatório a cada Pokémon selecionado.

Lê: Lista de Pokémon do jogador.

Muda: Nível do Pokémon.

Envia: Alerta do nível de cada Pokémon que o jogador selecionou.

Resulta: Partida em estado inicial, jogadores alocados, com 3 Pokémon com níveis definidos.

### 3.1.4. Evoluir Pokémon

Operação: Evolução de Pokémon baseado em seu nível.

Descrição: Evolui o Pokémon, se e somente se o nível mínimo for atingido para aquela evolução e o jogador desejar evoluir o Pokémon. O Pokémon que foi evoluído é retirado da lista de Pokémon do jogador e é adicionado o correspondente a sua evolução.

Lê: Nível do Pokémon.

Muda: Pokémon na lista de Pokémon do jogador.

Envia: Alerta de evolução de Pokémon, com o nome do Pokémon para qual ele evoluiu.

Resulta: Partida em estado inicial, jogadores alocados, com 3 Pokémon com níveis definidos.

### 3.1.5. Calcular atributos

Operação: Cálculo dos valores dos atributos do Pokémon.

Descrição: Com base nos atributos base e no nível do Pokémon, calcula seus atributos finais.

Lê: Atributos e nível do Pokémon.

Muda: Atributos do Pokémon.

Envia:

Resulta: Partida em estado inicial, jogadores alocados, com 3 Pokémon com níveis e atributos definidos.

### 3.1.6. Selecionar Move set

Operação: Seleção de move set do Pokémon.

Descrição: É gerado uma lista de ataques possíveis, com base no nível do Pokémon, e o jogador seleciona 4 dentre esses; muda o estado do jogador caso ele já possua os 3 Pokémon com move set selecionados.

Lê: Lista de Pokémon do jogador; Lista base de ataques.

Muda: Move set do Pokémon; estado do jogador.

Envia: Mensagem alertando que o jogador está pronto para iniciar a partida.

Resulta: Partida em estado inicial, jogadores alocados, com 3 Pokémon com níveis, atributos e move set definidos.

### 3.1.7. Iniciar partida

Operação: Inicia a partida.

Descrição: Inicia a partida se ambos os jogadores estiverem no estado pronto para iniciar partida; os jogadores selecionam um de seus 3 Pokémon para iniciar a partida; é verificado qual Pokémon possui o maior valor do atributo velocidade e o primeiro turno é do jogador que possui esse Pokémon.

Lê: Estados dos jogadores alocados na partida.

Muda: Estado da partida para iniciada; Pokémon em batalha.

Envia: Alerta de início de partida.

Resulta: Partida em andamento, jogadores alocados, Pokémon em combate alocados.

### 3.1.8. Verificar turno: novo Pokémon em batalha

Operação: Verificação e atribuição de turno.

Descrição: Verifica e atribui o turno a um dos dois jogadores quando há um novo Pokémon em batalha, verificando o atributo velocidade dos dois Pokémon em batalha.

Lê: Atributos dos Pokémon em batalha.

Muda: Qual jogador está no turno de jogo.

Envia: Mensagem para o jogador que está em seu turno de jogo.

Resulta: Partida em andamento, jogadores alocados, Pokémon em combate alocados e turno definido.

### 3.1.9. Atacar

Operação: Atacar o Pokémon adversário.

Descrição: Seleciona um ataque da lista de ataques; diminui o HP do Pokémon que foi atacado baseado em seu tipo e o tipo do ataque; pode alterar o estado dos Pokémon caso o ataque permita essa possibilidade; muda o turno para o jogador adversário.

Lê: Dados do Pokémon atacante, dados do Pokémon adversário.

Muda: HP do Pokémon adversário, estado do Pokémon adversário, turno da partida.

Envia: Atualização do estado de ambos os Pokémon.

Resulta: Partida em andamento, jogadores alocados, Pokémon em combate alocados.

### 3.1.10. Substituir Pokémon

Operação: Substituição de Pokémon em batalha.

Descrição: Faz a troca do Pokémon em batalha do jogador por outro Pokémon em sua lista que não possua o estado desmaio.

Lê: Pokémon em batalha, Lista de Pokémon do jogador.

Muda: Pokémon em batalha.

Envia:

Resulta: Partida em andamento, jogadores alocados, Pokémon em combate alocados.

### 3.1.11. Verificar estado dos Pokémon

Operação: Verificar estados dos Pokémon.

Descrição: Verifica, ao início de cada turno, o estado de ambos Pokémon em batalha. Em caso de Pokémon em estado de desmaio, o jogador é obrigado a substituir o Pokémon. Se algum dos jogadores possuir 3 Pokémon em estado de desmaio, a partida é encerrada.

Lê: Dados de ambos Pokémon; lista de Pokémon dos jogadores.

Muda: Pokémon em batalha, se for o caso.

Envia: Mensagem avisando o jogar que ele é obrigado a substituir um Pokémon com estado de desmaio.

Resulta: Partida ou em andamento ou encerrada.

## 3.2. Requisitos não funcionais

### 3.2.1. Linguagem de Programação

A linguagem de programação utilizada no desenvolvimento do software será Java.

### 3.2.2. Servidor de Jogos e Framework de rede

O servidor de jogos a ser utilizado no projeto é o servidor multijogadores NetGamesNRT e o framework NetGamesNRTFramework, o qual possibilita abstração dos detalhes da rede para o desenvolvedor

### 3.2.3. Especificação de projeto

O software deve apresentar especificações de projeto baseada em UML, segunda versão.

### 3.2.4. Interface gráfica para usuário

O programa deve ter interface gráfica única, partilhada pelos usuários durante a partida. A interface gráfica será desenvolvida em Java-Swing.