

# **Especificação de Requisitos**

**Projeto: Boxe SIN**

**Versão 1.0**

**João Mafra**

**João Paulo Martins**

**Pedro Henrique Mognon**

Universidade Federal de Santa Catarina  
Centro Tecnológico  
Departamento de Informática e Estatística  
Sistemas de Informação  
INE5608 – Análise e Projeto de Sistemas

## Especificação de Requisitos de Software

**Projeto:** Boxe SIN

**Gênero:** Jogo

**Versão:** 1.0

**Data:** 28/04/2014

### Histórico:

<b>Versão</b>	<b>Autor(es)</b>	<b>Data</b>	<b>Ação</b>
1.0	João Paulo Martins João Mafra Pedro Henrique Mognon	28/04/2014	Estabelecimento dos requisitos.

## Índice

### **1. Introdução**

#### 1.1 Objetivo

#### 1.2 Regras do jogo BoxeSIN

#### 1.3 Referencias

### **2. Visão Geral**

#### 2.1 Arquitetura do Programa

#### 2.2 Premissas de Desenvolvimento

### **3. Requisitos de Software**

#### 3.1 Requisitos Funcionais

#### 3.2 Requisitos não funcionais

### **4. Esboço de Interface Gráfica**

# 1. Introdução

## 1.1 Objetivo

Desenvolver um programa que possibilite dois usuários jogarem entre si partidas do jogo BoxeSIN.

O jogo tem como objetivo levar o seu oponente a *Knock-Out*(K.O.), ou seja, golpeá-lo até que ele fique sem pontos de vida para persistir em combate.

O desenvolvimento desse programa(jogo) faz parte da disciplina de Análise e Projeto de Sistemas, do curso de Sistemas de Informação da Universidade Federal de Santa Catarina.

## 1.2 Regras do Jogo

O BoxeSIN é um jogo de turnos, onde cada jogador representa um pugilista, com o objetivo de levar seu adversário ao K.O.

O ringue (tela de jogo) contém a imagem dos dois lutadores e também os golpes e defesas que podem ser selecionados, além da vida de cada um.

Ao iniciar a partida cada pugilista tem 200 Pontos de Vida, um nome e um lado de melhor desempenho, que determina a posição da sua guarda, a potência de seus golpes. Quando ambos pugilistas forem cadastrados a luta pode ser iniciada.

O jogador atacante pode escolher um entre os seguintes golpes:

- ✓ Direto de Direita (Golpe em cima na direita);
- ✓ Direto de Esquerda (Golpe em cima na esquerda);
- ✓ Gancho de Direita (Golpe embaixo na direita);
- ✓ Gancho de Esquerda (Golpe embaixo na esquerda).

O jogador defensor tem quatro opções de defesa:

- ✓ Esquiva para Direita (Não protege de Direto de Esquerda);
- ✓ Esquiva para Esquerda (Não protege de Direto de Direita);
- ✓ Pendulo para Direita (Não protege de Gancho de Esquerda);
- ✓ Pendulo para Esquerda (Não protege de Gancho de Direita).

Cada turno se finaliza quando os movimentos de ataque e defesa são escolhidos. Dependendo da opção de guarda, caso o atacante acerte o golpe, ele tirará mais ou menos vida do adversário. A tela é atualizada com os novos pontos de vida e a vez é passada, fazendo com que o defensor vire atacante e vice-versa.

O vencedor é aquele deixar sem pontos de vida ou acertar três golpes consecutivos, levando seu adversário ao Knock-Out.

### 1.3 Referência

<http://en.wikipedia.org/wiki/Boxing>

## 2. Visão Geral

### 2.1 Arquitetura do Programa

Programa orientado a objetos com plataforma cliente/servidor para dois jogadores.

### 2.2 Premissas de Desenvolvimento

- O programa deve apresentar uma interface gráfica bidimensional;
- O programa deve ser implementado seguindo o paradigma de orientação a objetos, utilizando a linguagem de programação Java, devendo executar em qualquer plataforma que disponha da máquina virtual Java;
- Deve utilizar a ferramenta NetGamesNRT para fazer interação entre os usuários.

## 3. Requisitos de Software

### 3.1 Requisitos Funcionais

**Requisito funcional 01 – Conectar:** O programa deve apresentar a opção de “conectar” vinculada a ação de identificação do jogador para dar acesso à sequência do jogo. A conexão será executada somente caso o jogador se identifique, através da mesma interface gráfica, com um *nickname* (um nome qualquer).

**Requisito funcional 02 – Iniciar partida:** O programa deve apresentar a opção “iniciar” para o início de uma nova partida.

**Requisito funcional 03 – Inicializar Jogo:** Para que o jogo seja iniciado é necessário que o usuário escolha seu nome e seu lado de preferência de guarda.

**Requisito funcional 04 – Procedimento de jogada:** Para o procedimento de jogada é necessário que o jogador atacante escolha uma opção de ataque entre:

- ✓ Direto de Direita (Golpe em cima);
- ✓ Direto de Esquerda (Golpe em cima);
- ✓ Gancho de Direita (Golpe embaixo);
- ✓ Gancho de Esquerda (Golpe embaixo).

Enquanto o jogador defensor escolhe uma opção de defesa entre:

- ✓ Esquiva para Direita (Não protege de Direto de Esquerda);
- ✓ Esquiva para Esquerda (Não protege de Direto de Direita);
- ✓ Pendulo para Direita (Não protege de Gancho de Esquerda);
- ✓ Pendulo para Esquerda (Não protege de Gancho de Direita).

Então a jogada é tratada de acordo com as guardas selecionadas e as telas atualizadas. O jogador que após a atualização estiver com 0 Pontos de Vida ou tiver sofrido o terceiro ataque direto consecutivo será considerado o perdedor da partida. Caso contrário, o jogo prossegue e o atacante vira defensor e o defensor atacante.

**Requisito funcional 05 – Desconectar:** O programa deve apresentar a opção de menu “Desconectar”, permitindo ao usuário desconectar-se do servidor, encerrando assim uma possível partida em andamento.

### **3.2 Requisitos não funcionais**

**Requisito não funcional 01 – Especificação de projeto:** Além do programa em funcionamento, escrito na linguagem Java, deve ser produzida especificação de projeto baseada em UML;

**Requisito não funcional 02 – Interface gráfica para o usuário:** O programa deverá ter interface gráfica única, compartilhada pelos usuários;

**Requisito não funcional 03 – Tecnologia de interface gráfica para usuário:** A interface gráfica deve ser baseada em Java-Swing.

## 4. Esboço da Interface Gráfica

