

# Especificação de Requisitos do Jogo NIM

Versão	Autores	Data	Ação
1.1	Otto Raphael Klotz D'Abril Matheus Alberto Pereira	17/04/2018	Estabelecimento de Requisitos

## Conteúdo

- 1- Introdução
- 2- Visão Geral
- 3- Requisitos de Software
- 4- Esboço da Interface

## 1 – Introdução

### Objetivo

Desenvolver um programa para realizar uma disputa de NIM entres dois usuários remotamente.

### Descrição do Jogo

Nim é um jogo em que dois jogadore se alternam na remoção de peças dispostas em um tabuleiro até que todas a peças sejam removidas. O objetivo do jogo é evitar ser o último jogador a retirar uma peça do tabuleiro porque este sera o perdedor da partida.

Tipicamente o tabuleiro esta montado ao criar 3 pilhas com 3, 5 e 7 peças cada e então a partida de fato se inicia. Posteriormente os jogadores se alternam retirando uma ou mais peças de uma, e somente uma, das pilhas até que a última peça seja removida. O jogador que remover a última peça será considerado o perdedor da partida.

## 2- Visão Geral

### Arquitetura do Programa

Arquitetura cliente-servidor, distribuído.

### Premissas

1. Implementação do programa feita em Java de modo a permitir a execução em qualquer máquina virtual Java.
2. Interface gráfica simples mas que permita a fácil participação em um partida.
3. Execução distribuída utilizando o NetGamesNRT.

## 3- Requisitos de Software

### Requisitos Funcionais

1. **Iniciar:** O jogo deve ser capaz de se comunicar com o servido do NetGamesNRT e iniciar uma partida com outro jogador.
2. **Conectar:** A aplicação deve ser capaz de estabelecer um conexão com o servidor.
3. **Desconectar:** Deverá ser possível desconectar do servido.
4. **Visualização:** O programa deve mostrar o “tabuleiro” do jogo contendo as pilhas e suas devidas quantidades de elementos.
5. **Finalizar:** O jogo dve ser capaz de reconhecer quando uma partida termina.
6. **Ganhador:** O Jogo deve também ser capaz de identificar o vencedor ao final de cada partida.
7. **Enviar jogada:** Deverão ser enviadas ao servidor as jodadas executadas.
8. **Receber jogada:** As jogadas do adversário deverão ser recebidas o estado atual devera ser autalizado.
9. **Jogar:** O jogador deve ser capaz de fazer as jogadas quando for sua vez e ser impossibilitado de alterar o estado do jogo quando do contrario.

### Requisitos Não Funcionais

1. **Suporte a aplicações distribuídas:** o programa deve executar distribuído ao NetGamesNRT.
2. **Interface com o usuário:** deve ser implementada uma interface em modo texto mostrando o atual estado do jogo de qual jogador é a vez.
3. **Especificação de projeto:** Além do código em Java o projeto deve apresentar especificação baseada nos padrões UML.

4- Interface

