

# Projeto Bubble Shooter

## Especificação de Requisitos

Versão 3.0  
05/07/2016

Versão	Autor(es)	Data	Ação
3.0	Bruno Honnef Raul M. Filho Ricardo Boing	05/07/16	Modificações para o correto funcionamento do código, além do aumento da facilidade de implementação.
2.0	Bruno Honnef Raul M. Filho Ricardo Boing	21/04/16	Nova descrição e maior clareza para os requisitos funcionais/não funcionais e sobre o jogo (incluindo suas regras). Também foi feito a troca do esboço da interface.
1.0	Bruno Honnef Raul M. Filho Ricardo Boing	13/04/16	Especificação de Requisitos.

## 1 Introdução

O jogo desenvolvido será baseado no clássico Puzzle Bobble, e deverá conter uma única opção de partida: ser disputado entre dois jogadores, de forma distribuída (ou seja, em rede). As regras e objetivos do jogo estão descritas a seguir.

### 1.1 Objetivos do jogo

A partida acontece dentro de um cenário retangular onde o jogador terá por objetivo destruir um conjunto de *bolas*<sub>[2]</sub>, localizadas na parte superior desse cenário. Para que isso seja possível o jogador contará com uma *máquina atiradora*<sub>[3]</sub> e um número infinito de bolas, iguais (as bolas) as que ele deve destruir. A máquina atiradora usará as bolas disponíveis ao jogador para atirar, uma a uma, na direção das bolas contidas no topo. A medida em que o jogador realiza um disparo (com a máquina), uma contagem é realizada de modo em que um número pré-definido de disparos acarretará na geração de novas bolas no topo da tela, que também deverão ser destruídas. Ou seja, após N rodadas, uma camada de bolas é acrescentada ao topo da tela de modo a aumentar o grau de dificuldade. Fazendo assim com que o número de bolas a ser estouradas aumente.

A disputa entre os jogadores será feita de maneira em que cada um terá seu próprio cenário de jogo, e quando um dos jogadores eliminar todas as bolas contidas em sua tela ele vencerá. Em contrapartida, caso o número de camadas de bolas no cenário seja superior ao limite permitido, a vitória é do adversário.

### 1.2 Regras do funcionamento do jogo

**Regra 1** – O jogo será iniciado contendo 3 (três) *camadas de bolas*<sub>[1]</sub>;

**Regra 2** – A destruição das *bolas<sub>[2]</sub>* deve cumprir três requisitos: (i) as bolas serão destruídas em conjunto, de modo que só acontecerá caso no mínimo três bolas estejam uma do lado da outra, em qualquer direção. Não é necessário que as três estejam simultaneamente ao lado de todas as outras, basta apenas que uma primeira esteja em contato com uma segunda e essa segunda com uma terceira, e assim por diante. (ii) as bolas devem pertencer ao mesmo tipo, ou seja, sejam da mesma cor. (iii) a destruição só será feita após houver a colisão entre as bolas. Caso o jogo se inicie com as bolas agrupadas, estas não deverão estourar;

**Regra 3** – Nenhuma *bola<sub>[2]</sub>* pode permanecer flutuando dentro do cenário, exceto no momento em que estiver sendo feito um disparo pela *máquina atiradora<sub>[3]</sub>*, no qual ela (a bola) percorrerá o cenário até atingir uma outra bola (ou o topo do cenário). Toda bola deve estar em contato com o topo do cenário ou com outra bolinha que siga essas mesmas especificações. Existe, contudo, a situação em que após a destruição de um grupo de bolas exista alguma que fique isolada no meio do cenário, portanto, como ela não obedece nenhuma das três especificações deverá ser destruída junto com as outras;

**Regra 4** – Cada jogador deve fazer sua jogada dentro de um limite de tempo de 10 (dez) segundos. Enquanto um jogador faz sua jogada o outro deve ficar bloqueado, sem a possibilidade de jogar. Caso o jogador não faça sua jogada dentro do tempo limite, a *bola<sub>[2]</sub>* contida na *máquina atiradora<sub>[3]</sub>* será disparada de forma automática, na direção em que a máquina estiver apontando naquele momento;

**Regra 5** – Após um disparo, a *bola<sub>[2]</sub>* deve sair da *máquina<sub>[3]</sub>* e seguir em linha reta, até encostar em outra bola ou no topo do cenário;

**Regra 6** – Duas telas serão mostradas a cada um dos usuários. Na tela à esquerda, este jogará e verá os resultados pelas jogadas procedidas. Enquanto a tela à direita, mostrará o resultado das jogadas do oponente;

**Regra 7** – Para movimentar a máquina afim de mirar na bola que deseje acertar, o jogador deve utilizar as setas de LEFT e RIGHT do teclado, fazendo com que a máquina seja movimentada horizontalmente. Basta apertar a tecla SPACE do teclado que a bola será lançada;

**Regra 8** – Para selecionar o que deseja no menu inicial do programa, o usuário deve utilizar as teclas UP e DOWN do teclado, selecionando o desejado com a tecla ENTER;

**Regra 9** – O usuário, para jogar, deve primeiramente se conectar ao servidor através do menu inicial e só após isso iniciar uma partida. Ele aguardará até que outro usuário se conecte e assim começará a partida.

## 1.3 Termos importantes

**1- Camadas de bolas:** Agrupará um número finito de *bolas<sub>[2]</sub>*, dispostas uma ao lado da outra, na posição horizontal. As camadas se agrupam umas acima/abaixo das outras. Dentro do jogo podem existir 0 ou N camadas contendo de 0 a X bolas, de modo que nenhuma camada pode estar abaixo da linha limitadora<sub>[5]</sub>;

**2- Bola:** objetos no formato circular, semelhante a uma bola no mundo real. Existem 4 (quatro) tipos de bolas, que são classificados pela sua cor: verde, vermelho, azul, amarelo;

**3- Máquina atiradora:** uma engenhoca cuja função é se deslocar horizontalmente e atirar bolas, com o objetivo de acertar outras bolas na parte superior do cenário. Possui uma seta que serve como mira, localizada no topo da máquina;

**5- Linha delimitadora:** uma linha que limitará o espaço disponível para ser ocupado por *camadas de bolas<sub>[1]</sub>*, dentro do cenário. Seu posicionamento será na parte inferior do cenário, acima apenas do da *máquina atiradora<sub>[3]</sub>*. Nenhuma camada pode aparecer abaixo dessa linha;

**6 – Servidor:** servidor NetGamesBR.

## 2. Visão Geral

### 2.1 Arquitetura do programa

- **Tipo de aplicação:** é uma aplicação do tipo distribuída, baseada em uma arquitetura cliente-servidor;
- **Paradigma de programação:** linguagem orientada a objetos.

### 2.2 Premissas de desenvolvimento

- O programa apresentará uma interface gráfica em 2D;
- O jogo acontecerá em rodadas, de modo que cada jogador fará uma jogada, e o conjunto de duas jogadas (uma de cada jogador) forma uma rodada.

## 3. Requisitos de Software

### 3.1. Requisitos funcionais

**Requisito funcional 1 (conectar à rede):** a primeira interação com o usuário se dará através da opção (um botão) de se conectar ao *servidor<sub>[6]</sub>*. Logo após ele entrará em uma fila de espera até que um outro jogador também se conecte ao jogo;

**Requisito funcional 2 (desconectar da rede):** o jogador deve ser capaz de se desconectar do *servidor<sub>[6]</sub>* quando assim desejar, e portanto deve haver um botão oposto ao de conectar a rede, que deverá desconectar o jogador do servidor;

**Requisito funcional 3 (iniciar partida):** em modo background, o sistema iniciará uma nova partida com base nas informações recebidas pelo *servidor<sub>[6]</sub>*. Assim que houverem os dois jogadores necessários para o início da partida, deve-se aparecer um aviso (para ambos jogadores) informando que a partida iniciará. O usuário que apertou o botão de iniciar partida terá o privilégio de iniciar o jogo com o primeiro lance;

**Requisito funcional 4 (procedimento de lance):** o programa deve suportar o procedimento de deslocar horizontalmente a *máquina atiradora<sub>[3]</sub>* e através dela realizar o disparo de uma *bola<sub>[2]</sub>*. O disparo deve acontecer de acordo com o descrito nas regra 4 e 7.

A cada lance o programa deve realizar os procedimentos de teste necessários para verificar se a partida terminou (e nesse caso também verificar quem venceu) ou se bolas foram destruídas.

Por fim, deve ser feito o envio das informações referentes ao estado atual da partida;

**Requisito funcional 5 (receber jogada):** o programa deve receber a jogada do oponente. Ou seja, quando o oponente realizar sua jogada, o resultado desta deve aparecer na tela à direita do usuário, logo após o procedimento de lance, mostrando o resultado em seu jogo;

### 3.2. Requisitos não funcionais:

**Requisito não funcional 1 (paradigma de programação):** o software deve seguir o paradigma de programação orientada a objetos;

**Requisito não funcional 2 (linguagem de programação):** deve ser desenvolvido utilizando a linguagem de programação Java, devendo executar em qualquer plataforma que disponhada da máquina virtual Java;

**Requisito não funcional 3 (especificação de projeto):** as especificações do projeto devem seguir o padrão UML 2;

**Requisito não funcional 4 (tecnologia de interface gráfica):** a interface gráfica deve ser baseada

em Java swing;

**Requisito não funcional 5 (distribuição do jogo em rede):** o framework NetGamesBR deve ser usado para tornar o jogo distribuído.

## 4. Interface Gráfica

Quando o programa iniciar deve aparecer uma tela com um menu contendo os seguintes botão: conectar, desconectar, iniciar partida e sair.

Ao dar início a uma partida deverão existir duas telas de jogo para ambos jogadores, numa mesma janela, sendo que a primeira (localizada na parte esquerda) mostrará o jogo/jogadas dele, e a segunda (localizada a direita) mostrará as de seu oponente. Cada uma das telas deverão ser compostas por bolas (de mesmo tamanho), camadas de *bolas<sub>[1]</sub>* (de mesmo tamanho), uma *máquina atiradora<sub>[3]</sub>* e uma *linha delimitadora<sub>[5]</sub>*. A distribuição desses objetos será feita da seguinte forma:

- *Bola<sub>[2]</sub>*: um conjunto de bolas ficaram posicionadas na parte superior do cenário, formando *camadas de bolas<sub>[1]</sub>*, e uma única bola deve estar presente na máquina atiradora;
- *Máquina atiradora<sub>[3]</sub>*: estará posicionada na parte inferior do cenário.

### 4.1 Esboço da interface de jogo

