

Projeto Gomoku

Especificação de Requisitos de Software

Versão 1.0

09/09/2019

Versão	Autor(es)	Data	Ação
1.0	Alfeu Gonçalves dos Santos João Pedro Brognoli Guilherme Nascimento	09/09/2019	Estabelecimento dos requisitos
1.1	Alfeu Gonçalves dos Santos João Pedro Brognoli Guilherme Nascimento	16/09/2019	Correção de erros

Conteúdo:

1. Introdução;
2. Visão geral;
3. Requisitos de software;
4. Esboço da Interface.

1 Introdução

Objetivo: Desenvolvimento de um programa que suporte a disputa de partidas de Gomoku na modalidade usuários contra usuário.

Referências:

(1) SILVA, Ricardo Pereira e. **Engenharia de Requisitos Produção de Especificação de Requisitos**. Florianópolis, 2019. 57 slides. Disponível em: <https://www.inf.ufsc.br/~ricardo.silva/INE5417e5608/apostila/capitulo00_analise_de_requisitos_2016.pdf>. Acesso em: 08 set. 2019.

(2) SILVA, Ricardo Pereira e. https://www.inf.ufsc.br/~ricardo.silva/INE5417e5608/apostila/requisitos_J_Velha.pdf. Florianópolis, 2019. 4 slides. Disponível em: <https://www.inf.ufsc.br/~ricardo.silva/INE5417e5608/apostila/requisitos_J_Velha.pdf>. Acesso em: 08 set. 2019.

(3 - **REGRAS DO JOGO**) TICASDEMATEMA. **Gomoku**. 2009. Disponível em: <<https://ticasdematema.blogspot.com/2009/05/gomuko-japao.html>>. Acesso em: 08 set. 2019.

2 Visão Geral

Arquitetura do programa: Programa orientado a objetos, com execução distribuída.

Premissas de desenvolvimento:

- O programa deve apresentar uma interface gráfica bidimensional;
- O programa deve ser implementado em Java, devendo executar em qualquer plataforma que disponha da máquina virtual Java.
- O jogo deve obrigatoriamente executar distribuído, não poderá ser de tempo real.
- O jogo deve envolver interação entre os participantes (em turnos).

3 Requisitos de Software

3.1 Requisitos Funcionais:

Requisito funcional 1 – Iniciar Nova Partida: O programa deve apresentar a opção de menu “novo” para o início de uma nova partida, considerando as seguintes restrições:

- A operação cria dois jogadores e estabelece o jogador que inicia;
- Caso haja peças no tabuleiro as peças são removidas e o tabuleiro fica vazio esperando a primeira jogada;
- É iniciado um temporizador que irá mostrar o tempo de andamento da partida para os usuários

Requisito funcional 2 – Sair da Partida: O programa deve apresentar a opção de menu “Sair” para que os jogadores possam sair da partida, considerando as seguintes restrições:

- Caso haja uma partida em andamento e algum usuário clicar em sair, deve ser exibido uma mensagem que existe uma partida em andamento – neste caso, apenas com a autorização do usuário a operação pode prosseguir;
- Quando um usuário sair do jogo com partida em andamento o outro jogador é declarado vencedor da partida e a partida chega ao fim.

Requisito funcional 3 – Configurar Opções de Jogo: O programa deve apresentar a opção de menu “Configurações” para que os usuários possam definir o seu nome para aparecer durante a partida, caso o usuário não digite nenhum nome será colocado o nome padrão de Jogador 1 ou Jogador 2 ;

Requisito funcional 4 – Guardar Histórico: O programa deve apresentar a opção de menu “Histórico” para que os usuários possam ver um histórico das últimas partidas jogadas, nesse histórico deve conter as informações do vencedor, do perdedor e da data de finalização da partida.

Requisito funcional 5 – Mostrar Ranking: O programa deve apresentar a opção de menu “Ranking” para que os usuários possam ver os usuários que mais venceram partidas, nesse ranking deve ser possível visualizar o nome dos usuários e quantidade de vezes que eles ganharam partidas.

Requisito funcional 6 – Analisar jogada: A cada tentativa de procedimento de jogada do usuário o programa deve verificar se ela é bem sucedida ou não, ou seja, a cada jogada deve ser verificado se o lance pode ser feito ou existe alguma restrição, considerando as seguintes restrições:

- A matriz de posições deve ser atualizada com a presença de uma peça, no caso de procedimento de lance bem sucedido;
- Em caso de lance irregular, não deve ser inserida uma nova peça. Exemplos: tentativa de colocar peão em posição ocupada, ou com a partida já encerrada.

Requisito funcional 7 – Proceder lance: O programa deve suportar o procedimento de lance dos usuários por meio de click de botão esquerdo do mouse sobre a posição em que deve ser colocada uma peça, considerando as seguintes restrições:

- O programa deve permitir apenas lances válidos, de acordo com as regras do jogo (ver referência 3);
- A cada lance válido, o programa deve verificar se o jogador que jogou ganhou a partida;
- A cada lance válido, caso não haja vencedor, o programa deve verificar se a partida encerrou empatada.
- Caso a partida terminar, será atualizado o ranking, histórico e finalizado o tempo de partida.

Requisito funcional 8 – Finalizar partida: Caso algum jogador ganhe a partida ou o algum jogador vença, deve ser exibido a mensagem mostrando o resultado da partida. Deve ser proibido novas jogadas e um dos usuários deverá criar uma nova partida para que possam ser efetuadas novas jogadas. Será atualizado o ranking, histórico e finalizado o tempo de partida.

Requisito funcional 9 – Mostrar Versão do Software: O programa deve apresentar a opção de menu “Sobre” para que os usuários possam ver a versão em que se encontra o jogo e os nomes dos desenvolvedores.

3.2 Requisitos Não Funcionais

Requisito não funcional 1 – Especificação de projeto: Além do código Java, deve ser produzida especificação de projeto baseada em UML, segunda versão;

Requisito não funcional 2 – Interface gráfica para usuário: A interface vista pelos 2 jogadores deve ser a mesma, ou seja, partilhada pelos usuários, sempre que houver a alteração de alguma peça ou alguma coisa no tabuleiro os 2 usuários devem ver a mudança;

Requisito não funcional 3 – Símbolos dos jogadores: A presença de uma peça em posição deve ser representada por um “bolinha branca” dentro da posição para um dos jogadores e uma “bolinha preta” para o outro;

Requisito não funcional 4 – Tecnologia de interface gráfica para usuário: A interface gráfica deve ser baseada em Java-Swing.

Requisito não funcional 5 – Linguagem da programação: O programa deve ser implementado em Java, devendo executar em qualquer plataforma que disponha da máquina virtual Java.

Requisito não funcional 6 – Modelagem UML: A modelagem UML deverá usar a ferramenta Visual Paradigm (community version)

Requisito não funcional 7 – NetGamesNRT: O jogo deve obrigatoriamente executar distribuído, com o suporte para aplicações distribuídas, NetGamesNRT

4 Esboço da Interface

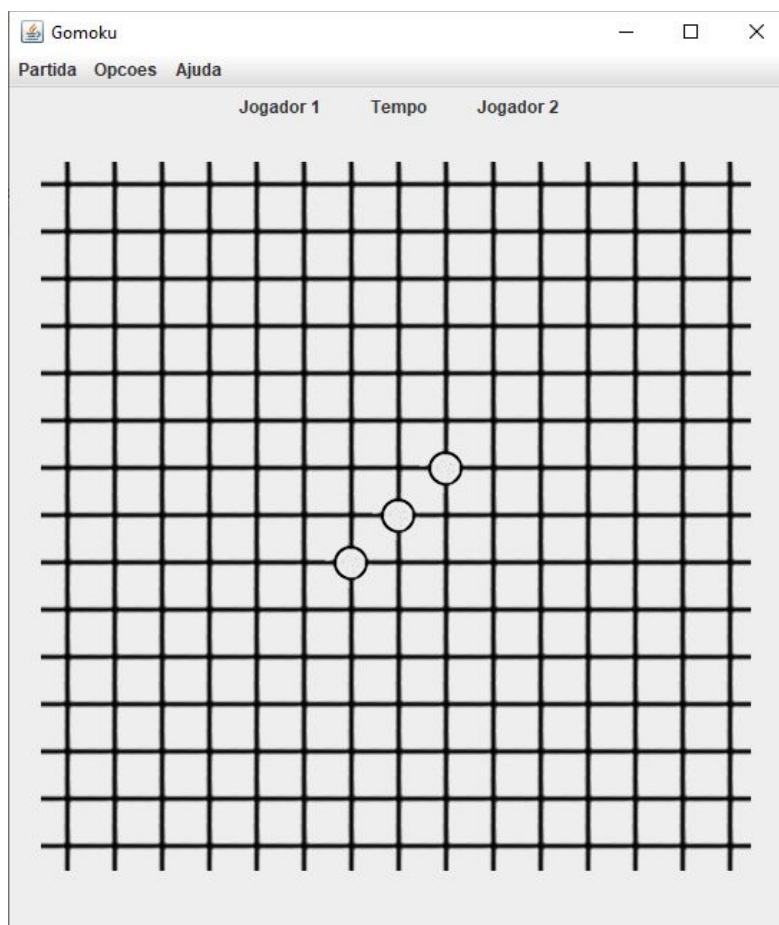


Figura 1: Tela inicial

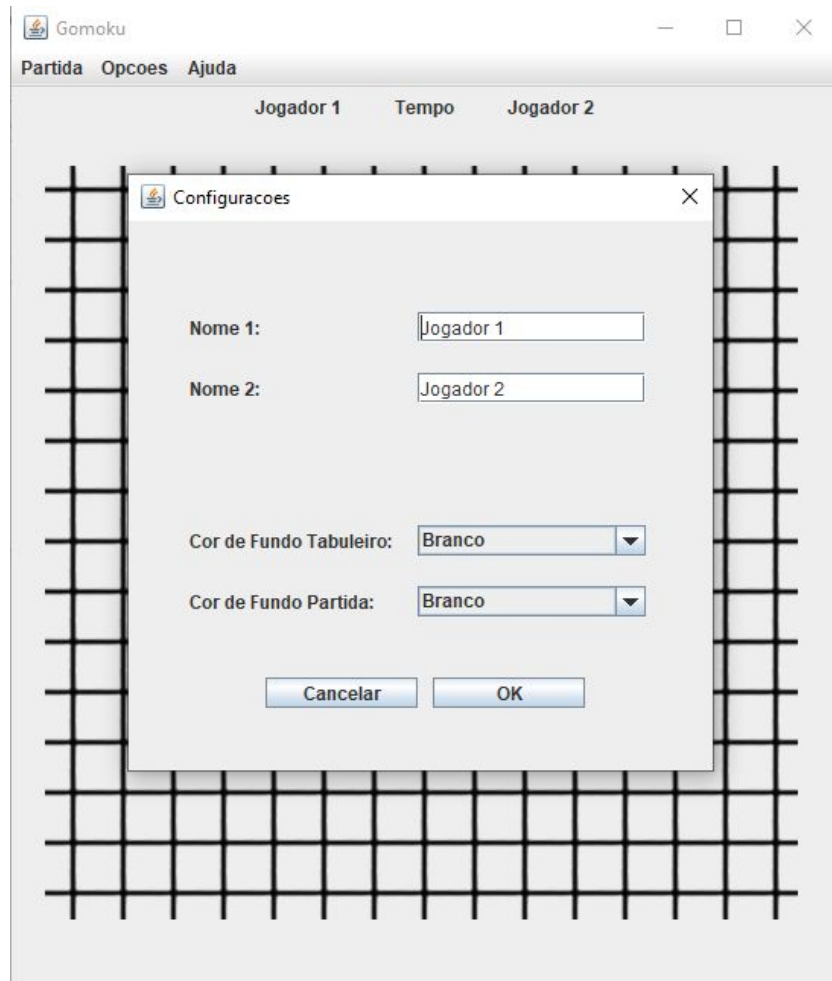


Figura 2: Tela de Configurações

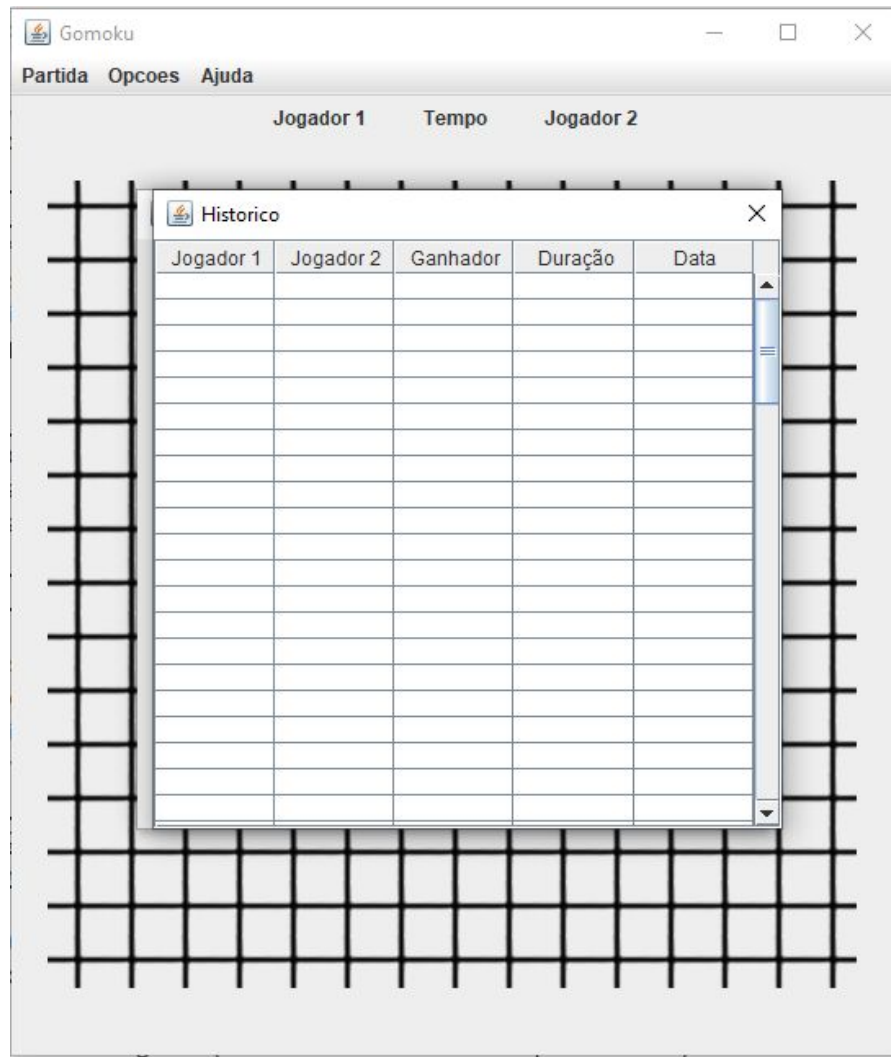


Figura 3: Tela de Histórico de Partidas

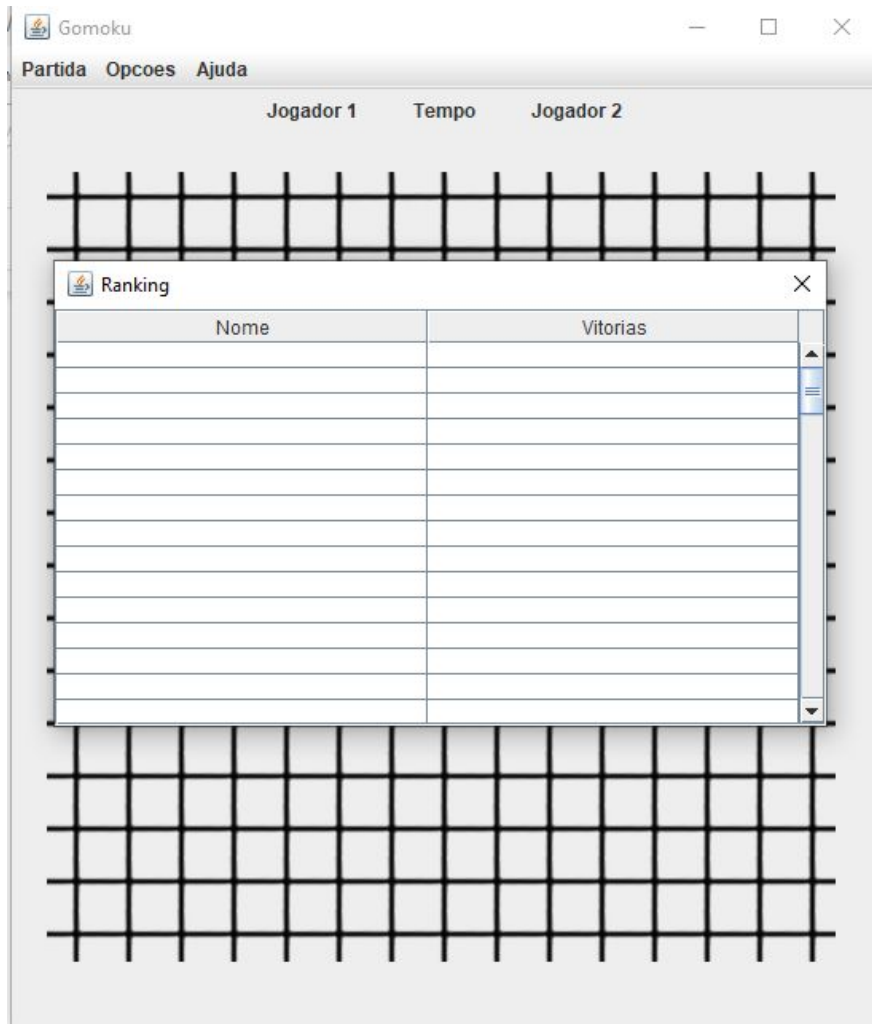


Figura 4: Tela de Ranking de Jogadores

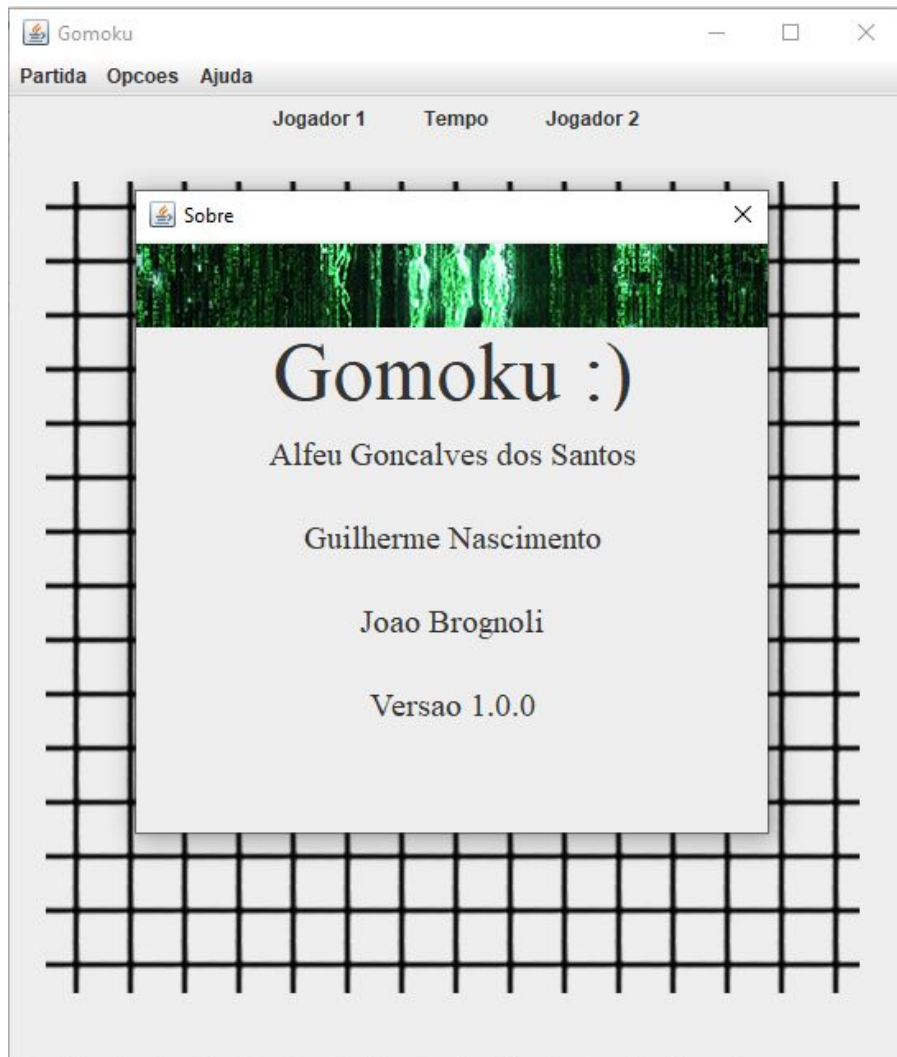


Figura 5: Tela com informações sobre o Jogo