

Projeto jogo Trilha

Especificação de Requisitos de Software

Versão	Autor(es)	Data	Ação
1.0	Felipe Backes Kettl Samantha Costa de Sousa Victor Rodrigues Gouvêa	15/09/2022	Estabelecimento dos requisitos
1.1	Felipe Backes Kettl Samantha Costa de Sousa Victor Rodrigues Gouvêa	16/10/2022	Adição de imagem nas regras do jogo e exclusão do requisito funcional "Começar nova partida"
1.2	Felipe Backes Kettl Samantha Costa de Sousa Victor Rodrigues Gouvêa	03/12/2022	Adição de menção ao caso de empate e outros detalhes de coerência entre os requisitos funcionais e a modelagem do jogo

Conteúdo:

1. Introdução
2. Visão geral
3. Requisitos do software

Apêndice: Regras do jogo Trilha

1.Introdução

1.1 Objetivo

Desenvolvimento de um programa distribuído que suporte a disputa de partidas do jogo Trilha na modalidade usuário contra usuário, on-line.

1.2 Definições, abreviaturas

Regras do jogo: ver apêndice.

1.3 Referências:

Apresentação das regras do jogo (vídeo do canal Vem Ka Jogar):

<https://www.youtube.com/watch?v=EzvqdQdCfAA>

Apresentação das regras do jogo (site Mega Jogos):

<https://www.megajogos.com.br/trilha-online/regras>

Design base da interface:

<http://windmill.cyrilf.com/>

2.Visão Geral

2.1 Arquitetura do programa

Cliente-servidor distribuído.

2.2 Premissas de desenvolvimento

- O programa deve ser implementado em Python;
- O programa deve usar DOG como suporte para execução distribuída;
- Além do código, deve ser produzida especificação de projeto baseada em UML, segunda versão.

3. Requisitos de Software

3.1 Regras de negócio:

Regra de negócio 1 - Movimentação de peças: A movimentação de peças é dividida entre a alocação (apêndice 1.1) e a movimentação propriamente dita (apêndice 1.2), esta última tendo limitações em função do número de peças remanescentes do jogador. Os movimentos devem ser validados considerando as seguintes fases do jogo:

1.1. Alocação de peças: o jogador insere as peças no tabuleiro até ter inserido o total de 9 peças.

1.2. Movimentação: após ter alocado as peças, o jogador somente poderá movimentar as peças ao longo das linhas do tabuleiro.

1.3. Voo: quando o jogador possui apenas 3 peças remanescentes, a movimentação de peças não possui limitação, ou seja, a peça poderá ser movida para qualquer espaço vago no tabuleiro.

3.2 Requisitos Funcionais:

Requisito funcional 1 – Iniciar programa: ao ser executado, o programa deve apresentar na interface o tabuleiro do jogo em seu estado inicial e solicitar o nome do jogador. Após isso, deve solicitar conexão com servidor do DOG (utilizando as ferramentas do DOG), sendo informado ao usuário o resultado desta solicitação. As demais funcionalidades deverão ser habilitadas apenas em caso de conexão sucedida. No caso de falha na conexão, o programa deve ser encerrado.

Requisito funcional 2 – Iniciar jogo: o programa deve apresentar a opção de “iniciar jogo” no menu, possibilitando o início de uma nova partida. O

procedimento de início de partida envia uma solicitação de início ao servidor Dog, que, por sua vez, retornará o resultado. Em caso de êxito, será identificada a ordem em que cada jogador irá jogar. Em caso de falha, o programa informa o motivo. A interface do programa deve ser atualizada com as informações recebidas e caso o jogador local seja quem inicia a partida, a interface deve estar habilitada para o procedimento de alocação de peças. Esta funcionalidade só deve estar habilitada se o programa estiver em seu estado inicial, isto é, sem partida em andamento e com o tabuleiro em seu estado inicial;

Requisito funcional 3 – Ver instruções do jogo: o programa deve apresentar a opção “ver instruções” no menu, para abrir uma janela pop-up com uma breve explicação de como jogar Trilha. Esta funcionalidade deve estar habilitada em qualquer estado do programa e não deverá interferir em partidas em andamento.

Requisito funcional 4 – Selecionar peça: O programa deve permitir ao jogador do turno selecionar uma peça presente em sua área de peças, caso esteja na fase “Colocar as peças”, ou no tabuleiro, caso esteja na fase “Mover as peças” (Ver Apêndice 1.1 e 1.2, respectivamente). A peça selecionada deve ser visualmente destacada na interface do programa. Se a ação for executada enquanto o jogador tem uma peça selecionada, a seleção atual será desfeita em prol da nova.

Requisito funcional 5 – Realizar movimento: O programa deve permitir a um jogador habilitado selecionar uma posição do tabuleiro, onde será colocada a peça previamente selecionada (de sua área de peças ou de uma posição do tabuleiro, dependendo da fase em que está o jogo). Esta funcionalidade só deve estar habilitada se o programa estiver com uma peça selecionada (ver Requisito funcional 4). O programa deve avaliar se a posição destino pode receber a peça selecionada, de acordo com as regras do jogo (ver apêndice e Regra de Negócio 1). Em caso positivo, deve efetuar a jogada - isto é, remover a peça de seu posicionamento inicial e colocá-la na posição destino - e, em caso negativo, notificar o lance irregular e o programa deve novamente aguardar a ação do jogador habilitado. No caso de êxito na colocação de peça em seu destino, o programa deve avaliar se há a formação de um novo “moinho”. Caso ocorra a formação do moinho, o programa altera a fase do jogo para “Remover peças adversárias” (apêndice 1.3), caso contrário, encerra o turno, passa a vez para o jogador adversário e envia a jogada ao adversário (utilizando os recursos de DOG). A jogada enviada deve conter a posição original da peça e a posição destino. Após passar o turno, deve ser

desabilitado o jogador local e o programa fica no aguardo de jogada do adversário (ver Requisito funcional 9) ou de notificação de abandono (ver Requisito funcional 10).

Requisito funcional 6 – Remover peça do adversário: o programa deve permitir ao jogador habilitado eliminar uma peça adversária do tabuleiro. Essa funcionalidade só ocorre no mesmo turno em que o jogo muda para a fase “Remover peças adversárias”. Caso o jogador tente selecionar uma peça própria, o programa deve mostrar a mensagem “selecione uma peça adversária para eliminar”. Caso o jogador selecione uma peça inválida, o programa deve mostrar a mensagem “não pode eliminar esta peça, selecione outra”. Caso todas as peças inimigas participem de um moinho, nenhuma peça inimiga poderá ser eliminada e a jogada é realizada sem eliminação de nenhuma peça. Após a seleção de uma peça válida, o programa remove a peça selecionada do tabuleiro e avalia o encerramento da partida, muda o jogo para a fase “Mover as peças” ou “Colocar as peças”, envia a jogada para o jogador adversário (utilizando os recursos de DOG) e passa o turno, ficando desabilitado enquanto aguarda a ação do jogador adversário. A jogada enviada deve conter a posição da peça removida. No caso de encerramento de partida, deve ser notificado o nome do jogador vencedor(ou empate, caso aconteça);

Requisito funcional 7 – Receber determinação de início: o programa deve poder receber uma notificação de início de partida, originada em Dog Server, em função de solicitação de início de partida por parte de outro jogador conectado ao servidor. O procedimento a partir do recebimento da notificação de início é o mesmo descrito no ‘Requisito funcional 2 – Iniciar jogo’, isto é, a interface do programa deve ser atualizada com as informações recebidas e caso o jogador local seja quem inicia a partida, a interface deve estar habilitada para seu procedimento de lance.

Requisito funcional 8 – Receber jogada: o programa deve poder receber uma jogada do adversário, enviada por Dog Server, quando for a vez do adversário do jogador local. A jogada recebida deve ser um lance regular e conter as informações especificadas para o envio de jogada no ‘Requisito funcional 6 – Realizar movimento’ ou no ‘Requisito funcional 7 – Remover peça do adversário’. O programa deve remover a peça de origem definida e colocá-la no destino caso seja uma jogada de movimento ou eliminar a peça alvo caso seja uma jogada de eliminação. Após isso, deve avaliar o encerramento de partida. No caso de encerramento de partida, deve ser notificado o nome do

jogador vencedor(ou empate, caso aconteça); no caso de não encerramento, deve ser habilitado o jogador local, para que possa proceder a seu lance.

Requisito funcional 9 – Receber notificação de abandono: o programa deve poder receber uma notificação de abandono de partida por parte do adversário remoto, enviada por Dog Server. Neste caso, a partida deve ser considerada encerrada, o abandono notificado na interface.

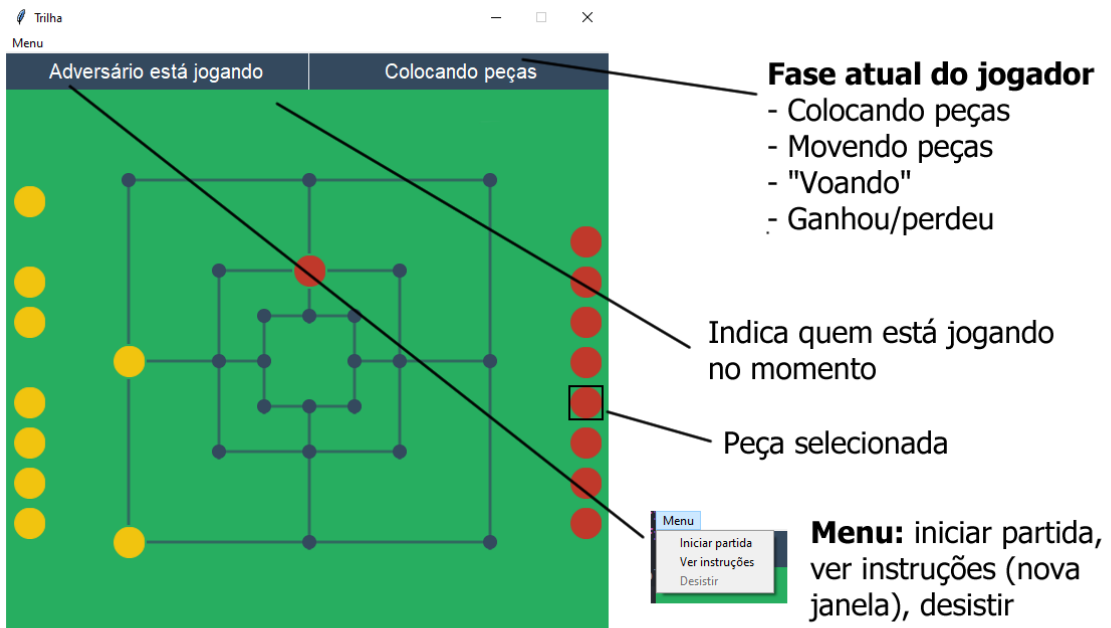
Requisito funcional 10 – Desistir da partida: o programa deve apresentar a opção de desistir no menu. Esta funcionalidade só é habilitada caso haja um partida em andamento. Caso o jogador clique em desistir, a partida é encerrada e é enviada a notificação de abandono ao jogador adversário (Requisito funcional 10).

3.3 Requisitos Não Funcionais:

Requisito não funcional 1 – Tecnologia de interface gráfica para usuário: A interface gráfica deve ser baseada em TKinter;

Requisito não funcional 2 – Suporte para a especificação de projeto: a especificação de projeto deve ser produzida com a ferramenta Visual Paradigm;

Requisito não funcional 3 – Interface do programa: A interface do programa será produzida conforme o esboço da imagem abaixo.



”

Apêndice: Regras do jogo Trilha

O jogo Trilha tem dois participantes que disputam entre si em um tabuleiro de 24 casas interligadas horizontalmente e verticalmente. Possui 18 peças, sendo repartidas 9 para cada jogador. O objetivo do jogo é deixar o adversário com 2 peças no tabuleiro ou deixá-lo sem movimentos.

1.0 Fases do jogo

O jogo consiste em três fases principais:

1.1 Colocar as peças: Esta é a fase inicial do jogo onde cada jogador coloca uma peça de cada vez alternando entre jogadores, caso um dos jogadores forme uma linha horizontal ou vertical com três peças (um moinho), ele terá o direito de remover uma peça de seu adversário do tabuleiro.

1.2 Mover as peças: Esta fase se inicia quando ambos os jogadores colocarem suas nove peças em jogo. Consiste em mover suas peças ao longo de uma das linhas do tabuleiro para uma outra casa adjacente. Caso um dos jogadores tenha somente 3 peças em jogo, ele pode "voar" com suas peças, podendo mover para qualquer casa que não esteja ocupada por uma peça do adversário.

1.3 Remover peças adversárias: Em qualquer uma das fases acima quando um jogador forma uma linha horizontal ou vertical com 3 peças ele fará um "moinho", isso lhe dá o direito de remover uma peça de seu adversário, contudo você não poderá remover uma peça do seu adversário que faz parte de um moinho dele, o que resulta na troca de turno onde não ocorre remoção de peça adversária.

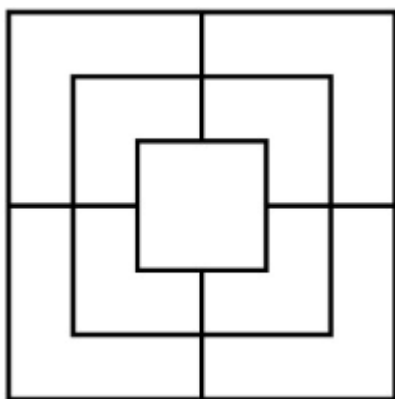


Figura 1: Tabuleiro do jogo Trilha

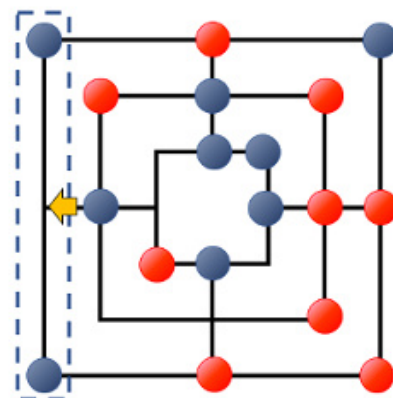


Figura 2: Possível situação de moinho

2.0 Encerramento de partida

O jogo termina quando 3 situações são alcançadas:

Se um jogador reduzir as peças de seu adversário para 2.

Se um jogador deixar seu adversário sem nenhuma jogada válida. Caso seu adversário tenha somente 3 peças em jogo, ele não poderá ser "trancado".

Se ambos jogadores estiverem com 3 peças em jogo e, a partir deste momento, se em 10 jogadas não houver vencedor, o jogo terminará e será declarado um empate.'