

Especificações de Requisitos

Projeto Pai Sho - versão street

Versão: 2.0

Data: 31/01/2022

Versão	Autores	Data	Ação
1.0	Gabriel da Silva Cardoso, Larissa Gremelmaier Rosa, Leonardo Seishi Yamazaki.	22/11/2021	Levantamento de requisitos para o projeto
2.0	Gabriel da Silva Cardoso, Larissa Gremelmaier Rosa, Leonardo Seishi Yamazaki.	31/01/2022	Conexão com netgames e atualização de diagramas de classe e casos de uso. Atualizado requisito não funcional RNF 3.

Conteúdo:

1. Introdução
2. Visão Geral
3. Requisitos de Software

1. Introdução

1.1 Objetivos do desenvolvimento

Desenvolvimento de um jogo distribuído que permite realizar partidas de Street Pai Sho entre dois jogadores.

1.2 Referências e regras

As regras e referências desse jogo podem ser encontrados nesse link, em língua inglesa: <https://skudpaisho.com/site/games/street-pai-sho/>

2. Visão Geral do Sistema

2.1 Arquitetura do Software

- Utiliza o paradigma Orientado a Objetos;
- Possui interface gráfica com os usuários;
- Sua apresentação é distribuída.

2.2 Premissas do desenvolvimento

- Deve ser desenvolvido utilizando a linguagem de programação Java;
- Sua modelagem será feita a partir da linguagem UML 2;
- A interface gráfica deve ser construída em duas dimensões (2D);
- Os jogadores interagem ao se conectarem no servidor Netgames e iniciarem uma partida.

3. Requisitos de Software

3.1 Requisitos funcionais

[RF1] Iniciar partida

(**WHAT**) Iniciar partida é a ação de posicionar e alocar todo recurso necessário para início do jogo, entre eles: colocar as 8 peças de cada jogador na posição correta do tabuleiro, escolher qual lado ficará com cada cor de peça e estabelecer as conexões necessárias com o servidor. (**WHY**) É necessário pois todo jogo pressupõe um início. (**WHO**) A partida tem início quando um jogador conectado ao servidor decide iniciar um jogo, (**WHEN**) e a qualquer momento, desde que não haja nenhuma partida em andamento, que o jogador esteja devidamente conectado e consiga ser pareado com outro jogador. (**WHERE**) A ação de iniciar uma partida ocorre direto na interface gráfica, ao clicar no botão “iniciar jogo”. (**HOW**) O sistema decide quem vai ser o jogador branco e quem vai ser o preto, após isso, coloca de

acordo com a matriz de possíveis posições as 8 peças no seu local inicial e manda um aviso para o jogador do primeiro turno.

[RF2] Fazer Jogada

(**WHAT**) O jogador que estiver em seu turno deve fazer a jogada, de acordo com a disponibilidade de peças e regras do jogo (**WHY**) para que o jogo tenha continuidade. (**WHO**) O jogador que estiver em seu turno (**WHEN**) enquanto o jogo não chegar na situação final de vitória (**WHERE**) A ação acontece na interface gráfica por meio do botão “Colocar peça” ou ao mover uma peça já em jogo. (**HOW**) O jogador deverá escolher entre duas opções de jogada, ao clicar em uma peça ou no botão de adicionar peças ao jogo sua jogada ficará marcada.

[RF3] Passar turno

(**WHAT**) Após terminar sua jogada, o jogador em turno deve submetê-la ao outro jogador, por meio do botão “Enviar jogada”(**WHY**) visto que é preciso alternância entre jogadores para o sentido do jogo. (**WHO**) É feito pelo jogador em turno (**WHEN**) após terminar sua jogada [RF2]. (**WHERE**) A ação de iniciar uma partida ocorre direto na interface gráfica, ao clicar no botão “Enviar Jogada”. (**HOW**) O usuário, após receber a mensagem de “jogada registrada”, deve clicar no botão na interface gráfica, assim, o sistema envia sua resposta salva localmente para o servidor, e troca o turno para o outro usuário conectado.

[RF4] Encerrar partida

(**WHAT**) Tal como iniciar partida, esse requisito significa o término do jogo (**WHY**) com o objetivo de avisar o ganhador do jogo quando qualquer um dos casos de vitória/derrota se concretizarem. (**WHO**) Qualquer jogador (**WHEN**) quando clicar em “desistir” em seu turno, quando desconectar da partida ou caso a situação de vitória ocorra (**WHERE**) A ação de encerrar partida pode acontecer na parte lógica do sistema, quando a situação de vitória for realizada ou pelos jogadores ao saírem da interface gráfica ou interagirem com a mesma por meio do botão “desistir”. (**HOW**) A cada jogada, o sistema calcula se a situação de vitória ocorreu, se sim, a partida acaba e a interface gráfica de ambos jogadores é atualizada com aquele que venceu a partida, voltando então para situação inicial. O mesmo ocorre caso algum jogador saia/desista da partida.

3.2 Requisitos não funcionais

[RN1] Implementação

Deverá ser implementado na linguagem de programação Java, na versão 8, com intermédio do framework Java swing.

[RN2] Interface gráfica

A interface com o usuário deverá ser construída em 2D, com destaque para o tabuleiro, área de mensagens e botões para “iniciar”, “desistir”, “passar turno” e “adicionar peças”.

[RN3] Conexão cliente-servidor distribuída

O jogo rodará distribuído, com os jogadores se conectando ao servidor e sendo pareados uns com os outros. Para isso, usaremos a ferramenta NetGames.