

Projeto Xiangqi

Especificação de requisitos de software

REQ_PR_Xiangqi_2023aug2

Versão 1.2

19/09/2023

Versão	Autor(es)	Data	Ação
1.0	Gustavo Konescki Führ, Enzo da Rosa Brum, Fábio Pereira dos Santos,	08/09/2023	Especificação de requisitos funcionais e não funcionais
1.1	Fábio Pereira dos Santos	19/09/2023	Atualização do requisito não funcional da interface
1.2	Gustavo Konescki Führ	07/10/2023	Atualização dos requisitos

Conteúdo:

1. Introdução
 - 1.1 Objetivo de desenvolvimento
 - 1.2 Definições e abreviações
 - 1.3 Referências (regras do jogo)
2. Visão geral
 - 2.1 Arquitetura de software
 - 2.2 Premissas de desenvolvimento
3. Regras de negócio
 - 3.1 Requisitos de domínio
4. Requisitos de software
 - 4.1 Requisitos funcionais
 - 4.2 Requisitos não funcionais

1. Introdução

1.1 Objetivo de desenvolvimento

Objetivo é desenvolver um programa do jogo Xiangqi de forma distribuída com suporte usuário X usuário, online.

1.2 Definições e abreviações

- RN: Regra de negócio
- RF: Requisito funcional
- RNF: Requisito não funcional

1.3 Referências (regras do jogo)

<https://brainking.com/pt/GameRules?tp=68>

2. Visão geral

2.1 Arquitetura de software

Cliente-servidor distribuído

2.2 Premissas de desenvolvimento

- O programa deve ser implementado em Python.
- O programa deve usar DOG como suporte de sistema distribuído.
- O projeto deve ser produzido com Visual Paradigm para a documentação UML.

3. Regras de negócio

3.1 Requisitos de domínio

RN01: A partida ocorre com apenas dois jogadores.

RN02: O jogo termina se um jogador abandonar a partida.

RN03: O jogo termina se ocorrer um cheque-mate ou um afogamento no rei adversário.

RN04: O jogo termina empatado se nenhum dos lados tem condições para provocar um cheque-mate ou um afogamento no rei adversário.

RN05: A peça só poderá se mover para uma posição, se seguir as regras características da peça e do jogo.

RN06: Um jogador não pode fazer mais de três cheques seguidos com a mesma peça na mesma posição.

RN07: Um jogador não pode ameaçar a mesma peça inimiga (que não esteja protegida por outra peça) movendo interminavelmente entre dois únicos pontos.

RN08: Os reis não poderão estar na mesma coluna se visíveis um ao outro.

4. Requisitos de software

4.1 Requisitos funcionais

RF01 - Iniciar programa: Quando executado, o programa deve inicializar a interface gráfica e tentar estabelecer conexão com o servidor (DOG). Se a operação for bem sucedida o usuário poderá iniciar partida, caso contrário, o usuário terá que encerrar o programa.

RF02 - Iniciar partida: Após estabelecer a conexão com o servidor, o programa deve apresentar a opção "Iniciar partida". O jogador terá a possibilidade de iniciar a partida clicando nessa opção. Nesse momento, o servidor buscará um oponente para o usuário. Se um oponente for encontrado, as peças do tabuleiro serão colocadas em suas posições iniciais, o programa determinará qual jogador usará as peças pretas e qual usará as peças vermelhas e será definido a vez e começará a contar o tempo para o jogador com peças vermelhas. Caso não seja possível encontrar um oponente, o usuário receberá uma notificação via interface (**RN01**).

RF03 - Selecionar peça: O programa deve permitir ao jogador da vez selecionar uma peça de sua cor. Quando isso ocorrer, o programa deve mostrar as posições que a peça selecionada pode alcançar na rodada seguindo os requisitos **RN05, RN06, RN07, RN08**.

RF04 - Mover peça: O programa deve permitir ao jogador da vez mover uma peça de uma posição a outra. O procedimento só poderá ser realizado após o jogador selecionar uma peça (**RF03**) e a ação deve cumprir com os requisitos **RN05, RN06, RN07, RN08**, ou seja, a posição deve ser uma das posições marcadas no tabuleiro após o procedimento **RF03**. Se a jogada não for válida, o programa aguarda uma nova ação. Caso a jogada seja válida, o programa deverá enviar a jogada para o servidor DOG e deverá retirar a peça da posição que estava e colocá-la na nova posição e, se houver, remover a peça adversária da posição. Após isso, o programa deve verificar se a partida acabou (**RN03 e RN04**). Se a partida acabar, deverá mostrar a cor da peça do jogador vencedor via interface. Caso contrário, passa a vez para o próximo jogador.

RF05 - Receber determinação de início: O programa deve receber notificação de início de partida, enviada pelo servidor DOG. Quando isso ocorrer, o

procedimento será o mesmo descrito na **(RF02)** sobre a inicialização da interface e lógica da partida.

RF06 - Receber jogada: O programa deve ser capaz de receber uma jogada válida feita pelo jogador adversário. Quando isso ocorrer, o procedimento será o mesmo descrito na **(RF04)** sobre o ajuste da interface e verificação de encerramento de partida.

RF07 - Receber notificação de abandono: O programa deve receber uma notificação de abandono de partida, enviada pelo servidor DOG. Quando isso ocorrer, a partida será encerrada e o abandono será comunicada via interface **(RN02)**.

4.2 Requisitos não funcionais

RNF01 - Tecnologia de interface gráfica para usuário: A interface gráfica deve ser implementada usando a ferramenta Tkinter.

RNF02 - Interface do programa: A interface do programa será produzida conforme o esboço da imagem abaixo:



