

PROJETO TP01.2 – Protocolo de Comunicação DamasNet

1. Objetivo

O objetivo deste documento é especificar o protocolo de comunicação **DamasNet**, desenvolvido no contexto da TP01.2. O protocolo permite a comunicação remota entre dois dispositivos para realização de partidas do jogo de damas, por meio de troca de mensagens de texto simples e estruturadas, no nível de aplicação.

2. Modelo de Comunicação

O protocolo utiliza o modelo **cliente-servidor**, onde:

- O **servidor** é responsável por iniciar e manter a sessão de jogo.
- O **cliente** conecta-se ao servidor para disputar a partida.

Ambos os lados podem enviar e receber comandos, seguindo a ordem dos turnos no jogo.

3. Descrição Geral do Funcionamento

A comunicação ocorre por meio de **mensagens em texto plano**, com comandos bem definidos, terminando com \n. Cada comando inicia com uma palavra-chave e pode conter parâmetros separados por espaços ou vírgulas. O protocolo cobre:

- Início da partida
- Troca de jogadas válidas
- Promoção de peças
- Capturas e múltiplas capturas
- Tratamento de erros
- Finalização da partida por vitória, empate ou desistência

4. Conjunto de Comandos e Respostas

Comando	Descrição	Formato/Exemplo
START jogador	Inicia o jogo com jogador X ou O	START X
MOVE x1,y1 x2,y2	Move peça da posição inicial para a final	MOVE 2,3 3,4
VALID	Confirma jogada válida	VALID
INVALID motivo	Informa jogada inválida com motivo	INVALID movimento ilegal
CAPTURE x1,y1 x2,y2	Move com captura de peça adversária	CAPTURE 4,5 6,3
KING x,y	Promoção a dama	KING 7,0
WIN jogador	Declara o vencedor	WIN O
DRAW	Declara empate	DRAW
QUIT	Indica desistência de um jogador	QUIT
ERROR código motivo	Erro de comunicação ou jogada	ERROR 400 formato inválido
PING / PONG	Teste de latência	PING / PONG
RESTART	Solicita reinício da partida	RESTART

5. Formato das Mensagens

- Estrutura:

COMANDO | parametro1 | parametro2 | ... \n

- Delimitadores:
 - | separa comandos e parâmetros.
 - , separa coordenadas (ex.: 5, 2).
 - \n indica o fim da mensagem.

Exemplo de fluxo de conexão e partida:

START | Lucas

OK | Bem-vindo Lucas. Você é o jogador r

TURN | r

MOVE | 5,2 | 4,3

OK

TURN | b

CAPTURE | 2,3 | 4,5

OK

TURN | r

WIN | r

6. Tratamento de Erros e Conexões

6.1 Jogadas Inválidas

- Resposta com: INVALID motivo
- Exemplo: INVALID casa ocupada

6.2 Mensagens Malformadas

- Resposta com: ERROR 400 mensagem mal formada

6.3 Perda de Conexão

- Se não houver resposta por 5 segundos: ERROR 408 timeout
- Após 3 tentativas: QUIT automático

6.4 Finalização

- Vitória: WIN jogador
- Empate: DRAW
- Desistência: QUIT

7. Exemplos de Sessão

Exemplo 1 – Jogada Válida

CLIENTE: MOVE 2,3 3,4\n
SERVIDOR: VALID\n

Exemplo 2 – Captura

CLIENTE: CAPTURE 3,4 5,2\n
SERVIDOR: VALID\n

Exemplo 3 – Promoção a Dama

CLIENTE: MOVE 6,1 7,0\n
SERVIDOR: VALID\n
SERVIDOR: KING 7,0\n

Exemplo 4 – Vitória

SERVIDOR: WIN X\n

8. Critérios de Implementação

1. Validação:

O servidor deve verificar:

- a. Se as coordenadas estão dentro do tabuleiro (8x8).
- b. Se o movimento segue as regras do jogo de damas.
- c. Se é o turno do jogador correto.

2. Sincronização:

Ambos os jogadores devem receber atualizações de estado instantaneamente.

3. Timeout:

Conexões inativas por mais de 30 segundos devem ser encerradas com:

ERROR | Timeout

9. Extras Opcionais

- **Ranking:**

Atualização de pontuação dos jogadores:

SCORE | player | points

- **Reconexão:**

Permitir reconexão de um jogador desconectado:

RECONNECT | player_id

- **Chat entre jogadores:**

Comunicação de texto durante a partida:

CHAT | Olá, boa sorte!