

# Trabalho Prático 3 (TP3) — Comunicação via WebSockets com Protocolo de Aplicação

---

## Objetivo

O objetivo deste trabalho é integrar comunicação em rede às aplicações desenvolvidas no TP1 (implementação do jogo), utilizando WebSockets para permitir a troca de mensagens entre os jogadores. O protocolo de aplicação descrito no TP2 deverá ser colocado em prática, sendo utilizado para formatar as mensagens enviadas entre as instâncias do jogo.

Além disso, o sistema deverá suportar o envio e recebimento de comandos diretamente por linha de comando (terminal), de forma a simular a interoperabilidade com diferentes interfaces e validar a robustez do protocolo de aplicação.

## Requisitos Funcionais

### 1. Comunicação com WebSockets

- O jogo deverá ser adaptado para funcionar em um modelo cliente-servidor ou P2P utilizando WebSockets.
- Utilizando conexão via IP e porta configuráveis.
- Assim que uma ação for realizada, uma mensagem deverá ser enviada para o outro jogador ou para o servidor contendo os dados da ação, seguindo o protocolo de aplicação definido no TP2.

### 2. Protocolo de Aplicação

- O formato das mensagens deve obedecer integralmente ao protocolo de aplicação descrito e documentado no TP2.
- É permitido fazer ajustes ou extensões no protocolo se necessário, mas quaisquer mudanças devem estar documentadas e justificadas.

### 3. Terminal de Comando (Interface Alternativa)

- Deve ser possível enviar comandos diretamente via terminal (linha de comando) para um IP e porta específicos, utilizando o mesmo protocolo do jogo.
- Essa funcionalidade tem como objetivo simular outras interfaces (como bots, painéis administrativos ou sistemas automatizados).
- O terminal deve permitir:
  - Enviar comandos válidos do protocolo de aplicação.
  - Receber e visualizar as respostas ou eventos gerados pelo jogo.

## Requisitos Técnicos

- A comunicação deve ser feita obrigatoriamente via WebSockets.
- O projeto pode ser implementado em qualquer linguagem com suporte à WebSockets.
- Toda a lógica de rede deve estar isolada da lógica do jogo na medida do possível, facilitando a manutenção e a reutilização.
- A interface do terminal pode ser simples (linha de comando padrão), desde que permita envio e recebimento de mensagens conforme o protocolo.

## Entrega

- Código-fonte completo do jogo com suporte à comunicação via WebSockets.
- Script ou utilitário para envio de comandos via terminal (interface alternativa).
- Documentação contendo:
  - Instruções de execução.
  - IPs e portas utilizados para testes.
  - Explicação de como o protocolo foi implementado e exemplos de mensagens.
  - Logs de sessões de teste entre dois jogadores e com o terminal.
  - (Se aplicável) Atualizações no protocolo de aplicação.

## Critérios de Avaliação

Critério	Pontos
Comunicação funcionando via WebSockets	25%
Envio correto de mensagens no protocolo definido	25%
Recepção e processamento de comandos no terminal	25%
Conformidade com o protocolo de aplicação	15%
Organização, clareza e documentação	10%

## Observações

- É **altamente recomendável** realizar testes entre diferentes máquinas (via rede local ou internet, se possível).
- A entrega será feita via repositório GitHub/GitLab, com instruções claras para execução e testes.
- Durante a avaliação, o professor poderá:
  - Jogar contra o aluno utilizando o terminal X aplicação, ou aplicação X aplicação.
  - Verificar logs de mensagens.
  - Testar falhas de comunicação ou comandos malformados.