

# Documentação Técnica — Jogo da Memória NFL com WebSockets

Ana Laura da Silva Nascimento  
Gabriel Ferreira de Sousa  
Heitor Ceolin Ribeiro

27 de agosto de 2025

## 1 Instruções de Execução

### 1.1 Requisitos

- Python 3.8 ou superior
- Bibliotecas:
  - customtkinter, pillow, tkinter
  - asyncio, websockets

### 1.2 Execução — Interface Gráfica

1. Certifique-se de ter Python 3.8+ instalado em sua máquina.
2. Instale as dependências com:

```
sudo apt-get install python3-tk    (apenas no Linux)
pip3 install customtkinter pillow websockets
```

3. Extraia o conteúdo do arquivo Jogo\_NFL.zip.
4. Navegue até o diretório do projeto.
5. Execute:

```
python3 jogo.py
```

6. A interface será exibida em tela cheia (pressione ESC para sair).

## 1.3 Execução — Terminal Multiplayer

O jogo também pode ser jogado no terminal em modo multiplayer.

**Servidor:**

```
python3 servidor.py
```

**Cliente (Linux):**

```
python3 jogo.py
```

## 2 Protocolo de Aplicação

### 2.1 Versão Anterior

Comandos principais:

- Start, Player?, Player On, Player Off
- Player Name, Cards, Card x,y
- Card Unavailable, Pair Cards, No Pair Cards
- +Point Player, Hide Cards
- Player x Off -> Player y Win
- No Cards Unavailable, Player x win
- Again, End

**Exemplo:**

```
Card 3,4  
Pair Cards 2,2;3,3  
+Point Player Ana
```

### 2.2 Versão Atual (Implementada no Código)

Comandos principais:

- CONNECT → Jogador se conecta
- TURN row col → Jogador seleciona carta
- CARD row col valor → Servidor revela carta
- MATCH / NO\_MATCH → Resultado da jogada
- SCORE x y → Atualiza placar
- BOARD → Estado do tabuleiro
- YOUR\_TURN / WAIT → Controle de vez

- WINNER 1, WINNER 2, WINNER DRAW
- END → Encerra manualmente
- ERRO ... → Mensagens de erro

**Exemplo:**

```
CONNECT
TURN 2 3
CARD 2 3 15
MATCH
SCORE 3 2
WINNER 1
END
```

## 3 Logs de Sessões de Teste

### 3.1 Servidor

```
Servidor rodando em ws://127.0.0.1:8765
Jogador 1 conectado.
Jogador 2 conectado.
```

### 3.2 Troca de Mensagens

```
Jogador 1 envia: TURN 2 3
Servidor responde: CARD 2 3 17
Servidor responde: BOARD [...]
Servidor responde: SCORE 1 0
```

## 4 Atualizações no Protocolo

- **Antes (v1.0):** protocolo verboso, suporte a mensagens como Player On, Pair Cards, +Point Player.
- **Agora (v2.0):** protocolo simplificado, comandos curtos em maiúsculo (TURN, CARD, MATCH, SCORE, WINNER), mais direto e fácil de processar.

## 5 Arquivos do Projeto

- `servidor.py` — servidor WebSocket que gerencia regras e turnos
- `jogo.py` — cliente/interface gráfica com suporte a multiplayer
- `imagens/` — pasta com cartas e verso