

# Relatorio T2 Redes

Mateus Siqueira Ruzene - 20221223<sup>1</sup>, Matheus Moraes Piovesan - 20224383<sup>1</sup>

<sup>1</sup>Departamento de Informática  
Redes 1 – I1058  
Universidade Federal do Paraná (UFPR)  
Curitiba – PR – Brasil

{msr22, mmp22}@inf.ufpr.br

## Visão Geral

O jogo segue as regras básicas do Blackjack e inclui as seguintes funcionalidades:

- **Dealer Rotativo:** Cada rodada, um jogador assume o papel de dealer.
- **Apostas e Distribuição de Cartas:** Os jogadores realizam apostas e recebem cartas de forma distribuída.
- **Pontuação e Decisões:** Cada jogador pode decidir se deseja *hit* (pedir carta) ou *stand* (manter sua mão).
- **Resultados e Gerenciamento de Fichas:** Pontuações são comparadas, e as fichas são ajustadas com base nos resultados.

## Protocolo de Comunicação

O jogo utiliza **UDP** (User Datagram Protocol) para troca de mensagens entre os jogadores. A escolha pelo UDP foi feita devido à sua baixa latência, já que o protocolo não exige confirmação de recebimento, tornando-o ideal para jogos em tempo real.

As mensagens são transmitidas em formato **JSON**, permitindo a troca estruturada de informações entre os jogadores.

## Estados do Protocolo

1. **BETTING:** Fase de apostas, onde cada jogador realiza sua aposta inicial.
2. **PLAYING:** Fase de jogo, onde os jogadores recebem cartas e tomam suas decisões.
3. **RESULTS:** Comparação de pontuações e distribuição de fichas.
4. **END\_ROUND:** Preparação para a próxima rodada.
5. **GAME\_OVER:** Encerramento do jogo quando todos os jogadores perdem suas fichas.

## Rede em Anel

A rede é estruturada como um anel lógico:

- Cada jogador possui dois sockets: um para receber mensagens do jogador anterior e outro para enviar mensagens ao próximo jogador.
- O controle do jogo é passado de jogador para jogador por meio de um **token** (chamado **BASTÃO** no código). Somente o jogador que possui o bastão pode gerenciar a fase atual.

## Como Funciona o Envio de Pacotes

- Os pacotes são enviados utilizando a função `sock.sendto`, que transmite mensagens para o próximo jogador no anel.
- A estrutura das mensagens segue um formato JSON contendo informações como estado do jogo, apostas, mãos de cartas e pontuações.
- A recepção das mensagens é gerenciada por `sock.recvfrom`, que decodifica os pacotes e executa ações apropriadas com base no estado atual.

## Instruções para Execução

### Pré-requisitos

- Python 3.x instalado.
- Bibliotecas padrão utilizadas: `socket`, `json`.

### Execução

1. Inicie o jogo passando o índice do jogador como argumento:  

```
python blackjack.py 0
```
2. Repita o comando em outros terminais, alterando o índice de 0 a 3.

### Configuração

O arquivo de configuração deve definir as portas e hosts para cada jogador.

### Recursos Importantes

- **Modularidade:** O código está organizado em múltiplos módulos, incluindo:
  - `main.py` lógica completa do jogo (protocolo de comunicação, `socket` DGRAM...)
  - `deck.py` para manipulação do baralho.
  - `player.py` para gerenciar os jogadores.
  - `card.py` para gerenciar as cartas do baralho.
  - `utils.py` para funções auxiliares, como leitura de configuração.