



PONG

Desenvolvimento de Sistemas Distribuidos

Fabiana Campos

João Roberto

Luis Gustavo



- Jogo Pong em uma arquitetura distribuída;
- O servidor (que também executa o jogo no Pygame) recebe comandos de dois clientes (jogadores) conectados via protocolo UDP;
- Usa o Pygame para criar a interface gráfica e a lógica do jogo;
- As raquetes dos jogadores (direita e esquerda) têm suas posições atualizadas com base nas mensagens UDP recebidas.

```
import socket

# Configurações do servidor
server_ip = "127.0.0.1" # Endereço IP do servidor
server_port = 1060      # Porta do servidor

# Criando o socket UDP
client_socket = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)

while True:
    try:
        # Mensagem a ser enviada
        message = input("Digite uma tecla: ")

        # Enviando a mensagem ao servidor
        client_socket.sendto(message.encode(), (server_ip, server_port))
        print(f"Mensagem enviada para {server_ip}:{server_port}")

    except Exception as e:
        print(f"Erro: {e}")
```

client.py :

- Envio de mensagens para o servidor via protocolo UDP;
- Coleta dados do usuário digitados no terminal e os envia ao servidor especificado (endereço IP e porta).

pongServidor.py :

- Responsável por coordenar a lógica do jogo e a interação com os jogadores;
- Recebe mensagens via UDP para atualizar a posição das raquetes dos jogadores;
- Usa o framework Pygame para renderizar o jogo, incluindo a bola, as raquetes e as pontuações na tela;
- Atualiza as posições das raquetes com base nas mensagens recebidas dos jogadores.

```

def envia_mensagem_udp(message):
    sock.sendto(message.encode(), (UDP_IP, UDP_PORT))
    print(f"Enviado: {message}")

running = True
while running:
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            running = False

        if event.type == pygame.KEYDOWN:
            if event.key == pygame.K_UP: # Seta para cima
                envia_mensagem_udp("jogador1_w")
            elif event.key == pygame.K_DOWN: # Seta para baixo
                envia_mensagem_udp("jogador1_s")

    screen.fill(bg_color)
    text = font.render("Para cima (↑) | Para baixo (↓)", True, light_grey)
    screen.blit(text, (screen_width / 2 - text.get_width() / 2, screen_height / 2))
    pygame.display.flip()

pygame.quit()

```

jogador.py :

- Permite que o jogador controle a raquete direita usando as setas para cima (↑) e para baixo (↓);
- Envia mensagens UDP específicas ("jogador1_w" ou "jogador1_s") para o servidor, informando os movimentos do jogador.

DEMONSTRAÇÃO

OBRIGADO!