

## 1. Introducción y Descripción del Proyecto

Nombre del proyecto: Basic movement for LAN multiplayer game.

El objetivo del proyecto es desarrollar un sistema de movimiento “multijugador” en LAN, en tiempo real. El propósito del proyecto es crear una base para videojuegos de este estilo. No hay un problema en particular que me haya motivado a hacer este proyecto. Este proyecto permite conectar múltiples jugadores en una LAN y mantener sincronizada la posición de cada jugador.

## 2. Alcance del Proyecto

Funcionalidades incluidas:

Movimiento de jugadores con teclas de dirección.

Sincronización de posiciones mediante un servidor LAN usando Pickle.

Funcionalidades no incluidas:

UI y lógica de videojuego.

Conexión a servidores externos o Internet.

Usuarios finales:

Jugadores locales conectados a la misma red.

## 3. Justificación y Necesidad

El proyecto es relevante porque permite experimentar el desarrollo de juegos multijugador en red, una habilidad útil en programación de videojuegos.

Se contextualiza dentro del área de desarrollo de videojuegos y networking, ya que combina la lógica de Pygame con la comunicación entre clientes y servidor usando Pickle, lo que permite aprender sobre sincronización de datos, manejo de sockets y diseño de juegos en tiempo real.

## 4. Tecnologías a Utilizar

Python: Lenguaje versátil, fácil de integrar con Pygame y networking.

Pygame: Librería para desarrollo de videojuegos 2D y manejo de gráficos, y eventos de usuario:

Sockets (Python): Permite la comunicación entre servidor y clientes en LAN de manera eficiente.

Pickle: Librería para serializar objetos de Python (jugadores, balas) y enviarlos por sockets.

Threading: Para manejar múltiples conexiones de manera simultánea en el servidor.

## 5. Estructura del proyecto

Esquema de archivos:

```
tpfinal/
├── server.py      # Servidor LAN, recibe y envía datos a los clientes
├── network.py    # Sistema de conexión
├── client.py      # Cliente que ejecuta el juego
├── player.py      # Clase Player y métodos de movimiento
├── requirements.txt
└── README.md
```

Diagrama de flujo de datos:

```
[Cliente 1] <---> [Servidor LAN] <---> [Cliente 2]
|               |               |
|<---- Posición ----->|       |
|               | <--- Posición --->|
```

## 6. Casos de Uso o Ejemplo de Funcionamiento

Caso de uso 1:

El jugador A se mueve hacia la derecha.  
El cliente envía la posición al servidor usando Pickle.  
El servidor transmite la posición a todos los demás clientes.  
Todos los jugadores ven la posición actualizada de A en tiempo real.

Caso de uso 2:

Dos jugadores se mueven simultáneamente en la LAN.  
Cada cliente envía su posición al servidor serializada con Pickle.  
El servidor sincroniza las posiciones y todos los jugadores ven los movimientos en tiempo real.

## 7. Conclusión y Futuras Mejoras

Este permite experimentar la programación de juegos multijugador en red local, enseñando sincronización de datos y comunicación cliente-servidor.

Mejoras futuras:

Implementar lógica de videojuego.  
Añadir UI.  
Agregar sistema de puntuación y ranking de jugadores.  
Optimizar la comunicación para manejar más jugadores simultáneamente.

