

## Trabajo Practico: Fase 2 - Red

### Objetivo

Implementar una arquitectura cliente servidor para el juego de fútbol desarrollado en la fase 1.

Tomar como guía el juego “**Tehkan World Cup**”  
 (“<https://www.youtube.com/watch?v=FbAzuqk0WZc>”)

### El Servidor

El servidor deberá proveer una partida multijugador de hasta 4 participantes como máximo. El valor de jugadores admitidos será configurable desde el archivo yaml. La partida iniciará una vez que se ha alcanzado ese número de jugadores conectados. No es necesario que el servidor tenga una interfaz gráfica, basta con lanzarlo por consola.

Cuando un jugador se desconecta voluntariamente ó por falla técnica, su futbolista activo deberá comportarse como cualquier otro futbolista no activo.

Si un jugador se quiere conectar al servidor y se supera la cantidad máxima de jugadores para la partida, el servidor le informa al cliente que la partida ya está llena.

Deben existir al menos cinco credenciales de usuarios en el archivo yaml las cuales se usarán para validar el login de los jugadores.

### El Cliente

Al iniciar la aplicación cliente, el usuario deberá poder conectarse al servidor mediante una pantalla de login y luego de ser aceptado deberá esperar a que se inicie la partida. El login de usuario deberá ser a través de una interfaz gráfica.

### Partido

El partido inicia con cada equipo en su mitad de cancha y comienza el juego con el primer pase en el círculo central. Se debe incluir la lógica de pases con pelota incluida.

### Cámara de seguimiento

La cámara de visualización debe estar centrada y seguir en todo a la pelota, excepto en los casos donde se alcanzan los límites del campo de juego.

## **Configuración**

El archivo de configuración será un archivo de texto plano en formato YAML, el cual contendrá los parámetros de configuración inicial de la aplicación. En caso de no encontrar el archivo, se debe leer un archivo de configuración por defecto.

Ejemplo de configuración:

```
---
degug:
  level: debug
equipo:
  casaca: principal
  formacion: 3-3
max_jugadores: 3
usuarios:
  -
    - juan
    - xxx
  -
    - martin
    - xxx
  -
    - diego
    - xxx
...
```

Pueden agregarse la cantidad de parámetros que se consideren necesarios.

## **Restricciones**

- La implementación deberá estar hecha en C/C++.
- Para la lectura y escritura de archivos YAML debe utilizarse una biblioteca.
- Para el manejo de gráficos se deberá usar la biblioteca SDL.
- Todo el código debe ser desarrollado íntegramente por cada grupo. No se permite la reutilización de código de cuatrimestres anteriores o de otras materias. Ante cualquier duda se deberá consultar con los docentes. La reutilización de código sin consulta previa será condición suficiente para la desaprobación de la materia.

**Fechas**

(Este cronograma puede sufrir modificaciones)

<b>Abril</b>	
4	Consultas y Avance TP Fase 1
11	Consultas y Avance TP Fase 1
18	Enunciado TP Fase 2 1er. Entrega TP Fase 1
25	Consultas y Avance TP Fase 2 2da. Entrega TP Fase 1
<b>Mayo</b>	
2	Consultas y Avance TP Fase 2 3er. Entrega TP Fase 1
9	Consultas y Avance TP Fase 2
16	Enunciado TP Fase 3 1er. Entrega TP Fase 2
23	Consultas y Avance TP Fase 3 2da. Entrega TP Fase 2
30	Consultas y Avance TP Fase 3 3er. Entrega TP Fase 2
<b>Junio</b>	
6	Consultas y Avance TP Fase 3
13	1er. Entrega TP Fase 3
20	Feriado
27	2da. Entrega TP Fase 3
<b>Julio</b>	
4	3er. Entrega TP Fase 3