



**Antonio José de Sucre**  
CORPORACIÓN UNIVERSITARIA

## EXÁMEN FINAL DE PROGRAMACIÓN

### TEMA:

CASO 5 – SISTEMA DE GESTIÓN DE TORNEO DEPORTIVO

### ESTUDIANTES:

Richard Manuel Vergara Ríos

### DOCENTE:

JAIDER JAVIER DE LA ROSA BERTEL

### FECHA:

17/11/2025

**UNIVERSIDAD ANTONIO JOSE DE SUCRE**



## ANÁLISIS DEL PROBLEMA

La administración de torneos deportivos multidisciplinario implica manejar una gran cantidad de información de manera organizada y eficiente. Entre los procesos necesarios se encuentran: el registro de equipos, el control del número de jugadores permitidos por cada disciplina, la programación de partidos (o fixture), la gestión de estadísticas individuales, y el cálculo automático de tablas de posiciones, registro de estadísticas para los equipos y deportes específicos y reportes generales que nos den información más detallada.

Sin un sistema automatizado, estas tareas suelen llevarse a cabo de forma manual, lo que incrementa la probabilidad de errores, inconsistencias y pérdida de datos. Además, la generación de calendarios deportivos implica cálculos complejos para garantizar una equidad entre los enfrentamientos, evitando repeticiones, asegurando descansos adecuados y adaptando agendas a la disponibilidad de canchas y horarios.

Este proyecto aborda el diseño e implementación de un sistema en Python capaz de gestionar un torneo deportivo para **tres disciplinas principales: fútbol, baloncesto y voleibol**. El sistema permite registrar jugadores, equipos y estadísticas en tiempo real. También incorpora funcionalidades más avanzadas como lo son:

- Validación automática de cupos mínimos y máximos por deporte.
- Generación del fixture mediante algoritmo Round Robin.
- Control de partidos jugados y pendientes.
- Registro de resultados con actualización automática de la tabla de posiciones.
- Obtención de rankings por jugador según estadísticas propias del deporte.
- Reportes analíticos, tales como historial de enfrentamientos, predicción de campeón y próximos partidos.

El sistema trabaja a través de una interfaz de menú en consola, lo que permite su uso sin requerir conocimientos avanzados de programación. Los datos son almacenados en estructuras internas (listas y diccionarios), lo que facilita búsquedas, actualizaciones y ordenamientos.



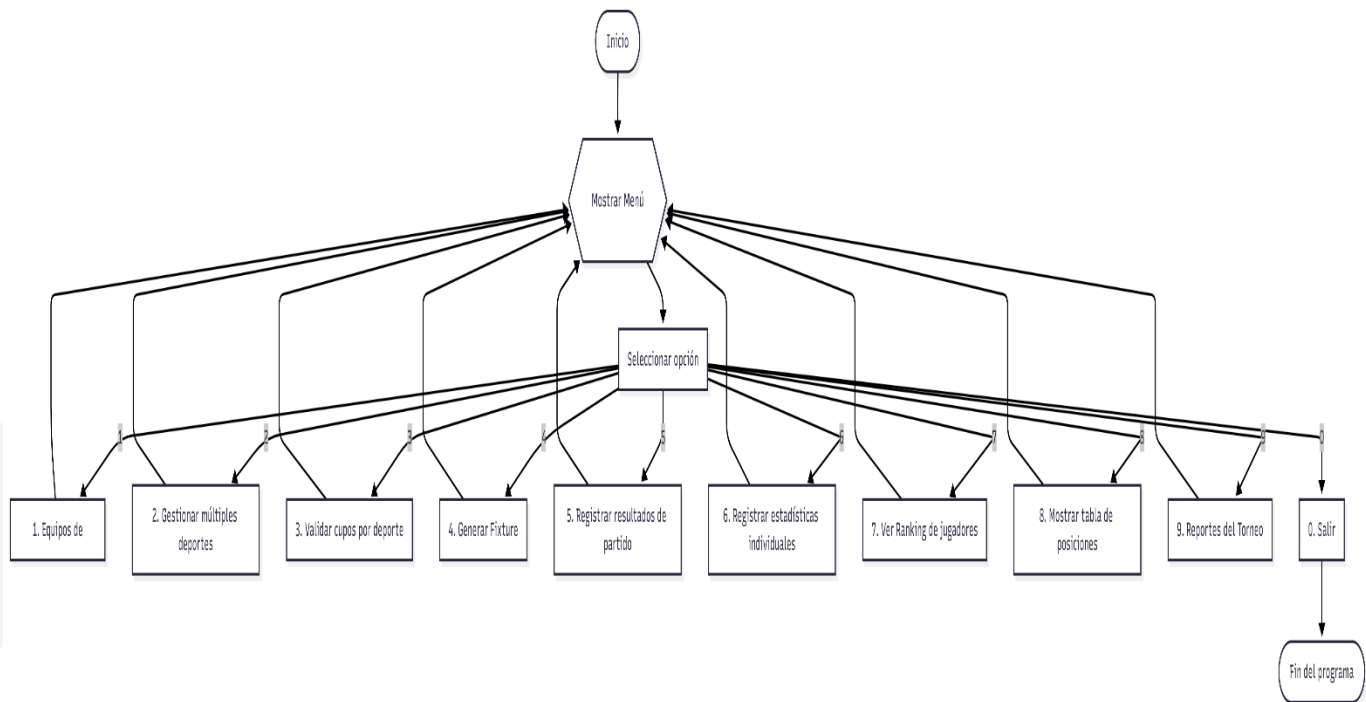
Este análisis evidencia la necesidad de un sistema automatizado para gestionar información deportiva de manera estructurada y más cómoda. El código desarrollado resuelve estas problemáticas mediante el uso de estructuras de datos eficientes, funciones bien organizadas y algoritmos adecuados para la gestión de calendarios y estadísticas.





## Diseño de la Solución

Flujo principal - Gestión Torneo Multidisciplina





## Explicación de las Estructuras de Datos Utilizados

### Listas y Diccionarios.

El programa hace uso principalmente de las listas y diccionarios (que estructuras nativas de Python) para almacenar equipos, jugadores, partidos y la tabla de posiciones. Estas estructuras permiten un acceso flexible, búsquedas sencillas, actualizaciones en tiempo real y un sentido de organización más cómodo y menos enrevesado.

Una de las listas principales que conforma este código es **equipos**, de tipo lista de diccionario, su propósito es almacenar los equipos registrados junto a otros datos como nombres, deportes, lista de jugadores, etc...

El ejemplo más sencillo es este:

```
{  
  "Nombre": "Atlético Nacional",  
  "Deporte": [1, 2, 3] # 1=fútbol,2=baloncesto,3=voleibol  
}
```

Con esto pueden llevarse a cabo las búsquedas lineales (`next((eq for eq in equipos if eq["Nombre"] == nombre), None)`), que sirve para agregar equipos y actualizar la lista de ID's de los jugadores.

Otra de las listas son **jugadores\_futbol**, **jugadores\_baloncesto** y **jugadores\_voleibol**. Que también son listas de diccionarios, su propósito es el de tener un diccionario con datos individuales para cada jugador y sus estadísticas específicas en dicho deporte.

Un ejemplo es este:

```
{
```





```
"ID": 11,  
"Nombre": "David Villa",  
"Edad": 26,  
"Posición": "Delantero",  
"Número de Camiseta": 10,  
"Equipo": "Atlético Nacional",  
"Goles": 6,  
"Asistencias": 3,  
...  
}
```

Se utilizan operaciones típicas, tales como búsqueda por ID, conteos por equipo y la actualización de estadísticas para ir acumulando jugadores una vez se vayan añadiendo según el usuario requiera.

Luego está **partidos\_pendientes**, **partidos\_jugados**, que se encarga principalmente de almacenar el fixture por jugar; **partidos\_jugados** lo que hace es guardar la información de los partidos jugados junto a su resultado.

Un ejemplo es este:

```
{  
  "Deporte": 1,  
  "Equipo 1": "Atlético Nacional",  
  "Equipo 2": "Deportivo Cali",  
  "Fecha": "20-11-2025",  
  "Hora": "10:00AM",  
  "Cancha": "Cancha A",
```



"Resultado": "2 - 1", # agregado al finalizar

"Ganador": "Atlético Nacional"

}

Donde enumeramos la lista para la selección del usuario. También utilizamos **Append y Pop** para mover los partidos entre las listas.

### Diccionarios.

Uno de los diccionarios clave en este programa es **tabla\_posiciones**, teniendo como propósito llevar una clasificación del puntaje de todo el equipo, tomando partidos jugados, partidos ganados, partidos empatados, partidos perdidos, puntos a favor, puntos en contra y la diferencia de puntaje.

**límite** es un diccionario tipo tupla, definiendo el mínimo y máximo de jugadores permitidos por deporte. Esto sirve para más adelante validar los equipos al agregar jugadores antes de un partido.

### Uso combinado: listas y diccionarios.

El diseño del código mezcla listas (para las colecciones con orden y posibles duplicados) y diccionarios para los accesos directos por clave. La combinación de estas funciones puede guardar jugadores en la lista **jugadores\_futbol** para luego, poder iterar y mostrar resultados. Lo que también funciona para mantener una referencia rápida de ranking/estadísticas agregadas por equipo en **tabla\_posiciones**.



## Manual de Usuario

### 1. INTRODUCCIÓN AL SISTEMA.

Primero que nada, el sistema se ejecuta en consola y permite administrar torneos de **fútbol, baloncesto y voleibol**. Todas las operaciones se realizan mediante un menú interactivo, en el cual el usuario selecciona opciones numéricas para navegar entre las funciones del mismo.

Al iniciar se te mostrará:

---Bienvenido al sistema de gestión de torneos deportivos multi-disciplina---

1. Registrar equipos.
2. Gestionar múltiples deportes (fútbol, baloncesto, voleibol).
3. Validar cupos por deporte.
4. Generación de Fixture.
5. Registrar resultados de partido.
6. Registrar estadísticas individuales de cada jugador.
7. Ver Ranking de jugadores.
8. Mostrar tabla de posiciones.
9. Reportes del torneo.
0. Salir.

### 2. REGISTRO DE EQUIPOS.

Nos permite ingresar nuevos equipos al torneo, indicando el nombre y el deporte al que pertenecen.

Primero se selecciona la opción 1 del menú, se ingresa el nombre del equipo y se elige del 1 al 3 en deportes siendo 1. Fútbol, 2. Baloncesto y 3. Voleibol. El equipo se agrega a la lista **equipos**. Y por defecto se crea una lista vacía de jugadores para ese equipo.





### 3. GESTIÓN DE JUGADORES POR DEPORTE.

Te deja elegir entre los 3 deportes (del 1 al 3) y acto seguido te muestra las siguientes opciones:

- Agregar jugador.
- Eliminar jugador.
- Ver lista de jugadores.
- Ver todos los equipos de x deporte.
- Volver al menú principal.

Una vez elijas una de las opciones se te pedirá o te mostrará la debida información que necesites para el momento. Tener en cuenta que cada deporte tiene su mínimo y máximo de jugadores... (no se puede rebasar el límite, pero en validación te avisará que tu equipo está completo.)

Para el fútbol: 11-18.

Para el baloncesto: 5-12.

Para el Voleibol: 6-14.

### 4. VALIDACIÓN DE CUPOS POR DEPORTE.

Esto se encarga de validar si cada equipo tiene el número mínimo y máximo permitido de jugadores antes de permitir que participe en partidos. Una vez selecciones la opción te dará un reporte de todos los equipos registrados hasta el momento.

### 5. GENERACIÓN DEL FIXTURE.

Esto genera todos los enfrentamientos del torneo utilizando el algoritmo **Round Robin** (todos contra todos), asignando automáticamente la fecha, la hora, los equipos que jugarán y en qué cancha lo harán. Puedes elegir una fecha o utilizar la de hoy para que el programa se encargue de darle horarios a los equipos.



## 6. REGISTRAR RESULTADOS DE PARTIDOS

Esta opción nos permite ingresar el marcador final de un partido pendiente. Pero antes de eso hay que seleccionar **la opción 5** para generar partidos de un equipo contra otro, una vez hecho esto... pasaremos a seleccionar el partido para colocar los resultados.

Únicamente puede haber empates para el fútbol, dado el caso, al equipo solo ganará un punto, mientras que una victoria otorgará 3 puntos al equipo ganador. Estos datos actualizarán la tabla de posiciones y mueve el partido a **partidos\_jugados**.

## 7. REGISTRAR ESTADÍSTICAS INDIVIDUALES.

Esta opción permite sumarle estadísticas a un jugador seleccionado después de un partido, según el deporte:

- Para fútbol: Goles, Asistencias, Tarjetas Amarillas, Tarjetas Rojas.
- Para Baloncesto: Puntos, Rebotes, Asistencias y Robos.
- Para Voleibol: Aces, bloqueos, recepciones.

Te pedirá los datos del jugador al que le quieras modificar las estadísticas.

## 8. VER RANKING DE JUGADORES.

Esto únicamente es un ranking ordenado según las estadísticas principales del deporte seleccionado. La estructura del Ranking está hecha para que se vea el jugador que tiene más de cada cosa según la categoría. Ya sea en tarjetas amarillas, bloqueos, anotaciones, etc...

## 9. MOSTRAR TABLA DE POSICIONES.

Mostrará la tabla de posiciones, en donde se encontrarán ubicados los equipos que ya tengan un puntaje existente registrado. Lo que incluye la tabla de posiciones es:



- Puntaje.
- Partidos Jugados (PJ).
- Partidos Ganados (PG).
- Partidos Empatados (PE).
- Partidos Perdidos (PP).
- Puntos a Favor (PF).
- Puntos en Contra (PC).
- Diferencia de Puntos.

Únicamente tienes que seleccionar la opción 8, elegir un deporte y se te mostrarán todos estos datos ordenados según el equipo.

## 10. MÓDULO DE REPORTES DEL TORNEO.

La opción 9 abre un submenú que ofrece las siguientes opciones:

- **Reporte completo del fixture.**
- **Equipos más ofensivos/defensivos**
- **Próximos partidos de un equipo.**
- **Historial de enfrentamiento.**
- **Predicción del campeón.**
- **Regresar al menú.**

Que nos genera un reporte completo de la opción que elijamos, aunque eso sí... para utilizar algunas opciones ya debe haber resultados en la lista de posiciones.

## 11. SALIDA.

Esto hace referencia a la **opción 0**, opción que básicamente rompe el código... es decir, te saca del código por completo. Dándote un mensaje de despedida.



## Caso de Prueba

En un caso de prueba, para verificar que el sistema permita registrar correctamente a un nuevo equipo asociado a un deporte. Usaremos como datos de entrada a los siguientes: **Tigres FC** del deporte **Fútbol (1)**.

En donde, al inicio utilizaremos la opción 1 para registrar el equipo, guardándose en la lista **equipos** con su atributo de **jugadores** vacío.

Luego nos vamos a la **opción 2**, en fase de agregar un nuevo jugador a un equipo ya existente. Esto serían los datos de entrada:

- ID: 101
- Nombre: Juan Pérez
- Edad: 22
- Posición: Delantero
- Número de camiseta: 9
- Equipo: *Tigres FC*
- Deporte: *1*.

De esa manera, el jugador registrado será añadido al diccionario jugadores\_futbol. Lo más recomendable sería añadir todos los jugadores necesarios para hacer del equipo funcional.

Luego pasamos a la **opción 3**, donde verificaremos si el equipo cumple con las condiciones para empezar a jugar, esta opción es únicamente para ver si todos los equipos cumplen con los requerimientos, así que no tiene mucha más profundidad.

Pasamos a la **opción 4**, lugar que servirá para generar el fixture, o mejor dicho, los partidos todos contra todos. Se verá a nuestro equipo registrado según el deporte que hayamos elegido, así, jugará contra otros equipos de forma aleatoria.

Pasamos a la **opción 5**, en donde podremos registrar los resultados de los partidos pendientes, eliges el que más te convenga y le asignas el puntaje a los equipos que te





convengan. Ejemplo. Tigres FC – 5 vs Titanes – 2. La victoria otorgará 3 puntos, mientras que un empate (sólo disponible en fútbol), otorgará 1 punto.

Con la **opción 6**, registraremos las estadísticas personales de los jugadores del equipo tras un partido, se nos mostrará unos datos de entrada en donde nos permitirán actualizar la información del jugador... por ejemplo:

Jugador ID 101 (Fútbol):

- Goles: 1.
- Asistencias: 1.
- Tarjetas amarillas: 0.
- Tarjetas rojas: 0.

Y el sistema se encargará automáticamente de actualizar al jugador en jugadores\_futbol.

Con la **opción 7**, se verán a todos los jugadores de dicho deporte ordenados según sus estadísticas actuales del “mejor” en cierta categoría al “peor”. Tendría que verse a nuestros jugadores creados con sus debidas estadísticas actualizadas.

Con la **opción 8**, una vez hemos hecho todos los pasos anteriores para decidir un resultado de un enfrentamiento, se mostrará nuestro equipo creado formando parte de la tabla de posiciones con el debido puntaje que ganó tras haber vencido o perdido contra su contrincante.

Con la **opción 9**, finalmente podremos ver un reporte completo sobre nuestro equipo, equipos con los que nos hayamos enfrentado y, además... podremos ver la trayectoria de otros equipos... viendo cuál se encuentra en el top ahora mismo y cuál es el equipo que tiene más probabilidades en ganar, basado en estadísticas.