



SIMULACIÓN **A3T5. INFORME FINAL**

Marco Antonio Kinil Kauil
Jorge Cesar Santos Ku
Javier Moises Huchim Villegas
Ariel Alberto Pat Canche

5-B

1. INTRODUCCIÓN Y CONTEXTO DEL PROBLEMA

1.1. Panorama del Scouting Deportivo en el Fútbol Profesional

En el contexto del fútbol profesional moderno, la toma de decisiones relacionadas con la adquisición de talento (fichajes) representa una de las actividades de mayor impacto financiero y deportivo para los clubes. Según datos de la FIFA y de ligas de élite como la Premier League, la Bundesliga o LaLiga, el mercado de transferencias mueve anualmente miles de millones de euros, con inversiones que frecuentemente superan los 100 millones de euros por jugador. No obstante, este mercado se caracteriza por una **alta incertidumbre y riesgo**, dado que el rendimiento futuro de un deportista está sujeto a múltiples variables dinámicas y probabilísticas, muchas de ellas difíciles de cuantificar con métodos tradicionales.

El proceso convencional de *scouting* se basa en gran medida en el análisis de estadísticas históricas y actuales (goles, asistencias, porcentaje de pases, distancia recorrida, entre otras), complementado con observación directa y criterio subjetivo de los ojeadores. Si bien estas métricas ofrecen una **instantánea del valor presente** del jugador, no logran capturar la **evolución temporal de su rendimiento**, la cual sigue una curva biológica típica: crecimiento acelerado en la juventud, estabilidad en la madurez (pico de rendimiento) y declive gradual en la etapa veterana. Esta limitación puede llevar a sobrevalorar jugadores en fase de declive o subestimar el potencial real de jóvenes promesas, generando **errores costosos** en las inversiones.

Ejemplos emblemáticos como el fichaje de Eden Hazard por el Real Madrid (2019, aproximadamente 100 millones de euros), cuyo rendimiento se vio mermado por lesiones recurrentes y un declive físico no anticipado, ilustran las consecuencias de una proyección insuficiente. Por el contrario, fichajes como el de Erling Haaland por el Manchester City (2022) demuestran el valor de identificar y proyectar correctamente el potencial de crecimiento de un joven talento.

1.2. Planteamiento del Problema

La **problemática central** que este proyecto aborda es la **incertidumbre en la proyección del rendimiento futuro** de los jugadores de fútbol, específicamente:

Falta de modelado dinámico: Las estadísticas tradicionales son estáticas y no incorporan la evolución natural del jugador a lo largo del tiempo.

Variabilidad estocástica no considerada: Factores impredecibles como lesiones, adaptación táctica, motivación psicológica o cambios de entorno (equipo, país, entrenador) introducen ruido significativo en la trayectoria de un deportista.

Sesgo subjetivo en la decisión: La dependencia excesiva del juicio humano puede llevar a decisiones inconsistentes o influenciadas por factores no deportivos.

Por lo tanto, se requiere una **herramienta analítica avanzada** que permita simular de manera estocástica la trayectoria futura de un jugador, integrando tanto la curva biológica esperada como la variabilidad aleatoria inherente al deporte.

1.3. Objetivo General del Proyecto

Desarrollar un **simulador de talento deportivo** basado en el paradigma de **Simulación de Eventos Discretos (SED)** que, a partir de estadísticas iniciales y parámetros de crecimiento/declive, genere proyecciones futuras del rendimiento de jugadores de fútbol, con el fin de:

Cuantificar el potencial a largo plazo.

Estimar la probabilidad de éxito en un horizonte temporal definido.

Proporcionar una base objetiva y analítica para la toma de decisiones de fichajes, reduciendo el riesgo financiero y deportivo.

Este informe documenta de manera integral el proceso completo del proyecto, desde el análisis conceptual y el diseño del modelo, hasta la implementación software, la ejecución de simulaciones, la discusión de resultados y las propuestas de mejora futura.

2. MARCO TEÓRICO BREVE SOBRE SIMULACIÓN APLICADA AL CASO

2.1. Simulación de Eventos Discretos (SED)

La **Simulación de Eventos Discretos** es una técnica de modelado computacional utilizada para representar sistemas cuyos estados cambian en puntos discretos en el tiempo, denominados "eventos". A diferencia de los modelos continuos, la SED avanza de evento a evento, lo que la hace eficiente para sistemas donde los cambios son abruptos y espaciados en el tiempo.

En el contexto deportivo, cada **mes** (o unidad de tiempo) en la carrera de un jugador puede considerarse un **evento discreto**, en el cual se actualizan sus estadísticas según reglas probabilísticas y factores de desarrollo. Esta aproximación permite capturar la **dinámica temporal** del rendimiento sin necesidad de resolver ecuaciones diferenciales continuas.

2.2. Modelado Estocástico y Curvas de Rendimiento Deportivo

El rendimiento deportivo no es determinista; está influenciado por una combinación de factores **sistemáticos** (edad, experiencia, entrenamiento) y **aleatorios** (lesiones, forma momentánea, suerte). Por ello, un modelo realista debe incorporar **componentes estocásticos**, típicamente representados mediante distribuciones de probabilidad (uniforme, normal, etc.).

La literatura científica en ciencias del deporte (ej., *Journal of Sports Sciences*, estudios de la UEFA) ha establecido que el rendimiento físico-técnico de un futbolista sigue una **curva en forma de campana (bell-shaped curve)** en función de la edad:

Fase de crecimiento ($\approx 18\text{--}24$ años): Mejora rápida en capacidades técnicas, tácticas y físicas.

Fase de pico ($\approx 25\text{--}29$ años): Máxima eficiencia y estabilidad; pequeñas fluctuaciones alrededor del óptimo.

Fase de declive (≥ 30 años): Reducción gradual de capacidades físicas (velocidad, resistencia), con posible mantenimiento o mejora de aspectos tácticos y experienciales.

Este proyecto **modela explícitamente** esta curva mediante factores multiplicativos dependientes de la edad, añadiendo ruido aleatorio para simular la variabilidad observada en la realidad.

2.3. SimPy como Herramienta de Implementación

SimPy es una biblioteca de Python diseñada específicamente para SED. Su arquitectura basada en **procesos** (generadores) y **eventos** permite modelar entidades concurrentes (en este caso, jugadores) que evolucionan en un entorno de tiempo simulado. Cada jugador se implementa como un **proceso SimPy** que ejecuta un ciclo de evolución mes a mes, actualizando sus estadísticas y registrando su historial. Esta elección tecnológica permite una implementación **clara, modular y escalable**, alineada con las mejores prácticas de la ingeniería de software.

3. METODOLOGÍA: ANÁLISIS, MODELADO Y SOFTWARE UTILIZADO

3.1. Análisis del Sistema y Delimitación

El sistema se concibe como un **modelo de simulación estocástica** con los siguientes componentes:

A. Entradas (Inputs):

Datos del Jugador:

Nombre (identificador).

Edad inicial (en años, con precisión decimal).

Estadísticas base por 90 minutos:

Goles/90min.

Asistencias/90min.

Porcentaje de pases acertados (0–100%).

Distancia recorrida (km/partido).

Parámetro de Control:

Tiempo a simular (en meses).

B. Procesos (Transformación):

Inicialización del entorno SimPy y creación de instancias de jugadores como procesos concurrentes.

Iteración temporal: Avance mes a mes (`env.timeout(1)`).

Motor estadístico de evolución: En cada mes:

Actualización de la edad.

Cálculo del **factor de desarrollo** según rango etario:

< 24 años: Crecimiento positivo (`uniform(1.001, 1.015)`).

24–29 años: Estabilidad (`uniform(0.995, 1.005)`).

≥ 30 años: Declive (`uniform(0.980, 1.000)`).

Aplicación de **ruido estocástico** (distribución normal) para técnica ($\sigma=0.02$) y físico ($\sigma=0.05$).

Actualización de cada estadística mediante fórmulas específicas:

Goles y asistencias: Multiplicación por (`factor + ruido_tecnico`).

Pases: Adición de variación uniforme y *clamp* entre 0 y 100.

Distancia: Multiplicación por (`factor_físico + ruido_físico*0.1`), con ajuste de declive acelerado después de los 28 años.

Cálculo del Puntaje General Ponderado:

$$Puntaje = (G \times 40) + (A \times 30) + (P \times 0.5) + (D \times 3)$$

$$Puntaje = (G \times 40) + (A \times 30) + (P \times 0.5) + (D \times 3)$$

Con penalización del 10% si edad > 30 años.

C. Salidas (Outputs):

Tabla de potencial: Estadísticas proyectadas y puntaje final de cada jugador.

Recomendaciones de scouting: Identificación del mejor jugador en cada métrica (goleador, asistidor, pasador, físico) y del **mejor prospecto general** (mayor puntaje ponderado).

Historial de evolución (opcional para análisis detallado).

3.2. Diseño del Software

El sistema se implementó en **Python 3**, utilizando:

SimPy para el núcleo de simulación.

Tkinter para la interfaz gráfica de usuario (GUI).

Random para la generación de números aleatorios.

La arquitectura sigue un patrón **claro de separación entre backend y frontend**:

Backend (lógica de simulación):

Clase Jugador: Representa a un jugador, con atributos (estadísticas, edad) y el método `evolucionar()` que define su comportamiento en el tiempo simulado.

Métodos de cálculo de puntaje y registro de historial.

Frontend (interfaz de usuario):

Clase `ScoutingApp`: Gestiona la ventana principal de Tkinter, con:

Formulario para ingreso de datos de jugadores.

Tabla (Treeview) para visualización de jugadores cargados.

Controles para configurar tiempo de simulación.

Área de texto para mostrar resultados detallados.

Esta separación facilita el mantenimiento, las pruebas y futuras ampliaciones (por ejemplo, añadir visualizaciones gráficas con Matplotlib).

3.3. Diagrama de Flujo del Proceso de Simulación

Aunque no se incluye gráficamente en este texto por limitaciones de formato, el flujo lógico puede describirse textualmente:

Inicio: Aplicación GUI en ejecución.

Carga de datos: Usuario ingresa datos de uno o más jugadores mediante formulario.

Validación: Verificación de que todos los campos numéricos son válidos.

Configuración: Usuario define meses a simular (por defecto 12).

Ejecución: Al presionar "EJECUTAR SIMULACIÓN TEMPORAL":

Se crea entorno SimPy.

Cada jugador se instancia como proceso.

Se ejecuta la simulación hasta el tiempo especificado.

En cada mes, cada jugador actualiza sus estadísticas según las reglas estocásticas.

Post-procesamiento: Se calculan puntajes, se identifican mejores en cada categoría.

Visualización de resultados: Se muestran en el área de texto de la GUI.

Fin: Usuario puede limpiar datos y realizar una nueva simulación.

4. RESULTADOS GLOBALES DE LA SIMULACIÓN

4.1. Configuraciones de Prueba y Parámetros Evaluados

Para validar el modelo, se ejecutaron múltiples simulaciones con tres jugadores de perfil diferenciado, cuyos datos iniciales se muestran en la siguiente tabla:

Jugador	Edad (años)	Goles/90min	Asist./90min	% Pase	Distancia (km/partido)	Perfil Analizado
P-01	19.2	0.45	0.20	85.5	10.5	Promesa (joven)
P-02	27.8	0.75	0.35	92.0	11.8	Pico (madurez)
P-03	33.5	0.20	0.10	88.0	9.0	Veterano (declive)

Se evaluaron dos horizontes temporales: **12 meses** (1 año) y **24 meses** (2 años). Debido a la naturaleza estocástica del modelo, cada ejecución produce resultados ligeramente diferentes; los valores presentados corresponden a una ejecución representativa.

4.2. Resultados Cuantitativos

Simulación a 12 meses:

Jugador	Edad Final	Goles/90min	Asist./90min	% Pase	Distancia (km)	Puntaje General
P-01	20.2	0.52	0.23	83.9	11.07	102.72
P-02	28.8	0.77	0.36	92.1	11.26	121.30
P-03	34.5	0.19	0.10	81.7	7.66	67.04

Simulación a 24 meses:

Jugador	Edad	Goles/90mi	Asist./90mi	%	Distanci	Puntaj
---------	------	------------	-------------	---	----------	--------

r	d Fina l	n	n	Pas e	a (km)	e Genera l
P-01	21.2	0.44	0.20	91.2	12.76	107.46
P-02	29.8	0.60	0.28	77.8	9.56	100.05
P-03	35.5	0.16	0.08	76.0	6.39	59.22

4.3. Análisis Cualitativo y Tendencias Observadas

Jugador Joven (P-01):

Tendencia general de crecimiento, especialmente en métricas físicas (distancia aumentó de 10.5 a 12.76 km en 24 meses, +21.5%).

Variabilidad en métricas técnicas (goles fluctuaron), reflejando el **ruido estocástico** y el proceso de aprendizaje.

Puntaje general aumentó de 102.72 a 107.46, confirmando su potencial de apreciación.

Jugador en Pico (P-02):

Estabilidad relativa a 12 meses, con pequeñas mejoras en goles y asistencias.

Declive incipiente a 24 meses, especialmente notable en distancia (9.56 km, -19.0% respecto a inicial) y porcentaje de pases (77.8%, -15.4%).

Puntaje general descendió significativamente (121.30 → 100.05), ilustrando el inicio de la fase de declive después de los 29 años.

Jugador Veterano (P-03):

Declive pronunciado y consistente en todas las métricas.

Distancia reducida en un 29.0% a 24 meses (9.0 → 6.39 km).

Puntaje general bajo (67.04 → 59.22), con penalización por edad >30 años.

4.4. Recomendaciones de Scouting Generadas por el Sistema

Para la simulación de 24 meses, el sistema produjo el siguiente análisis automático:

text

--- REPORTE DE SCOUTING --- (Proyección a 24 meses)

SI BUSCAS GOLES: Ficha a P-02

- Proyección: 0.60 goles/90min (Edad futura: 29.8)

SI BUSCAS ASISTENCIAS: Ficha a P-02

- Proyección: 0.28 asist/90min

SI BUSCAS POSESIÓN (PASES): Ficha a P-01

- Proyección: 91.2% de efectividad

SI BUSCAS INTENSIDAD (FÍSICO): Ficha a P-01

- Proyección: 12.76 km/partido

MEJOR PROSPECTO GENERAL (Balance Calidad/Edad): P-01

- Este jugador ofrece el mejor retorno de inversión basado en su curva de desarrollo.

Estos resultados demuestran la capacidad del sistema para:

Diferenciar jugadores por perfil y necesidad táctica.

Identificar al mejor prospecto general considerando edad y potencial de crecimiento.

Proporcionar recomendaciones específicas y cuantificadas.

4.5. Validación del Modelo frente a la Realidad Deportiva

Las tendencias proyectadas coinciden con la evidencia empírica reportada en la literatura deportiva:

Crecimiento de jóvenes: Mejora en físico y técnica variable.

Declive de veteranos: Reducción marcada en métricas físicas.

Pico entre 25–29 años: Máximo rendimiento con estabilidad relativa.

La inclusión de **ruido estocástico** evita proyecciones demasiado optimistas o deterministas, incorporando la imprevisibilidad real de las carreras deportivas.

5. PROPUESTAS DE MEJORA AL SISTEMA ANALIZADO

Si bien el sistema actual cumple con los objetivos planteados, su diseño modular y la claridad de su implementación permiten identificar varias oportunidades de mejora, tanto técnicas como funcionales:

5.1. Mejoras en el Modelo de Simulación

Incorporación de Correlaciones entre Métricas:

Problema actual: Las estadísticas evolucionan de manera independiente.

Mejora propuesta: Implementar una **matriz de correlaciones** (por ejemplo, mediante cópulas o modelos de redes bayesianas) para reflejar relaciones reales como:

Mayor distancia recorrida podría correlacionarse con más asistencias en mediocampistas.

Porcentaje de pases podría influir positivamente en la posesión y, indirectamente, en oportunidades de gol.

Beneficio: Proyecciones más realistas y coherentes desde el punto de vista táctico.

Modelado de Lesiones Graves como Eventos Discretos:

Problema actual: El ruido estocástico simula variaciones menores, pero no eventos catastróficos.

Mejora propuesta: Introducir un proceso de **riesgo de lesión** con probabilidad dependiente de la edad, historial de minutos jugados y fatiga acumulada. Una lesión grave podría reducir temporal o permanentemente ciertas estadísticas.

Beneficio: Mayor capacidad para evaluar el **riesgo de fichaje**, crucial en inversiones millonarias.

Simulación Monte Carlo con Múltiples Corridas:

Problema actual: Una sola ejecución proporciona una posible trayectoria, pero no una distribución de resultados.

Mejora propuesta: Ejecutar **1000+ simulaciones** para cada jugador y calcular:

Valor esperado de cada estadística.

Intervalos de confianza (ej., 95%) para cuantificar la incertidumbre.

Probabilidad de alcanzar umbrales de rendimiento (ej., probabilidad de que un delantero supere 0.5 goles/90min).

Beneficio: Información mucho más rica para la toma de decisiones bajo incertidumbre.

Factores Externos Contextuales:

Problema actual: El modelo no considera el entorno (liga, nivel del equipo, estilo de juego del entrenador).

Mejora propuesta: Añadir parámetros de **ajuste de liga** (multiplicadores que reflejen la dificultad de la competición) y **compatibilidad táctica** (bonificaciones o penalizaciones según el rol del jugador en el sistema del equipo).

Beneficio: Proyecciones más personalizadas y adaptadas al contexto del club comprador.

5.2. Mejoras en la Interfaz de Usuario y Experiencia

Visualizaciones Gráficas Integradas:

Problema actual: Los resultados se presentan solo en texto.

Mejora propuesta: Integrar **Matplotlib** para generar gráficos automáticos dentro de la GUI:

Curvas de evolución de cada estadística a lo largo del tiempo.

Diagramas de radar (spider charts) para comparar perfiles de jugadores.

Histogramas de distribución de puntajes en simulaciones Monte Carlo.

Beneficio: Mayor impacto visual y facilidad para identificar patrones.

Exportación de Resultados a Formatos Estándar:

Problema actual: No hay funcionalidad de exportación.

Mejora propuesta: Añadir botones para exportar:

Tabla de resultados a **CSV** (compatible con Excel, Tableau).

Gráficos a **PNG/PDF**.

Reporte completo a **PDF**.

Beneficio: Facilitar la integración con flujos de trabajo profesionales de análisis deportivo.

Base de Datos de Jugadores Precargada:

Problema actual: El usuario debe ingresar manualmente todos los datos.

Mejora propuesta: Conectar con una **API de datos deportivos** (ej., Football-Data.org, Transfermarkt) o incluir una base de datos local con estadísticas de jugadores reales.

Beneficio: Ahorro de tiempo y reducción de errores de entrada.

Análisis de Sensibilidad Interactivo:

Problema actual: Los parámetros del modelo (rangos de factores, desviaciones del ruido) son fijos en el código.

Mejora propuesta: Permitir al usuario ajustar estos parámetros mediante **sliders o entradas numéricas** en la GUI, y observar en tiempo real cómo cambian las proyecciones.

Beneficio: Mayor transparencia y comprensión del modelo por parte del usuario final.

5.3. Mejoras en la Robustez y Escalabilidad del Código

Pruebas Unitarias y de Integración:

Problema actual: El código no incluye un conjunto formal de pruebas.

Mejora propuesta: Desarrollar tests con **pytest** o **unittest** para:

Verificar la correcta inicialización de jugadores.

Validar las fórmulas de evolución estadística.

Asegurar que la GUI responde correctamente a entradas válidas e inválidas.

Beneficio: Mayor confiabilidad del software y facilidad para realizar modificaciones futuras.

Arquitectura Cliente-Servidor o Web:

Problema actual: Aplicación de escritorio monolítica.

Mejora propuesta: Refactorizar el backend como un **servicio REST API** (con Flask o FastAPI) y el frontend como una **aplicación web** (con React, Vue.js) o al menos con Tkinter conectado vía HTTP.

Beneficio: Escalabilidad (muchos usuarios simultáneos), acceso remoto y posibilidad de integración con otras herramientas de analítica.

Paralelización de Simulaciones:

Problema actual: Las simulaciones se ejecutan de forma secuencial.

Mejora propuesta: Utilizar **multiprocessing** o **concurrent.futures** para ejecutar múltiples corridas Monte Carlo en paralelo, aprovechando múltiples núcleos de CPU.

Beneficio: Reducción drástica del tiempo de cálculo para análisis complejos.

Documentación Automatizada y Ayuda en Línea:

Problema actual: La documentación está separada del código.

Mejora propuesta: Usar **docstrings** consistentes (formato NumPy o Google) y generar documentación automática con **Sphinx**. Incluir un sistema de **tooltips** o ayuda contextual en la GUI.

Beneficio: Mejor mantenibilidad y facilidad de uso para nuevos analistas.

5.4. Consideraciones Éticas y de Uso Responsable

Transparencia del Modelo:

Problema actual: El usuario final podría percibir el sistema como una "caja negra".

Mejora propuesta: Incluir una sección explicativa en la GUI que detalle los supuestos del modelo, las limitaciones y el significado de los parámetros.

Beneficio: Decisiones más informadas y evitación de sobreconfianza en la herramienta.

Evitar Sesgos en los Datos de Entrada:

Problema actual: El modelo puede perpetuar sesgos existentes si los datos iniciales no son representativos (ej., subvaloración de jugadores de ligas menos mediáticas).

Mejora propuesta: Implementar **mecanismos de normalización** estadística que ajusten las métricas según la liga, posición y minutos jugados.

Beneficio: Evaluaciones más justas y comparables a nivel global.

Complementariedad con el Juicio Humano:

Mejora propuesta: Diseñar la salida del sistema para que sea un **instrumento de apoyo a la decisión**, no un reemplazo del criterio experto de scouts, directores deportivos y entrenadores.

Beneficio: Toma de decisiones híbrida, combinando lo mejor del análisis cuantitativo y la experiencia cualitativa.

6. CONCLUSIONES Y APRENDIZAJES

6.1. Conclusiones Generales

El proyecto "**Simulador de Talento Deportivo con SimPy**" ha demostrado ser una **herramienta viable y valiosa** para la proyección estocástica del rendimiento futuro de jugadores de fútbol. A través de la implementación de un modelo de Simulación de Eventos Discretos, se ha logrado:

Capturar la dinámica temporal del rendimiento deportivo, superando la limitación de las estadísticas estáticas.

Incorporar la incertidumbre inherente mediante componentes estocásticos (factores de desarrollo aleatorios y ruido), generando proyecciones más realistas.

Generar recomendaciones accionables para el scouting, diferenciadas por perfil táctico y horizonte temporal.

Proporcionar una base cuantitativa objetiva que complementa y enriquece el juicio subjetivo de los especialistas.

El sistema cumple satisfactoriamente con los **objetivos específicos** planteados:

Modelar la evolución estocástica mediante un motor estadístico dependiente de la edad.

Definir agentes de simulación con la clase Jugador como proceso SimPy.

Generar una métrica de potencial (puntaje general ponderado) para clasificación.

Crear una interfaz de usuario intuitiva con Tkinter para entrada de datos y visualización de resultados.

6.2. Aprendizajes Clave del Proyecto

Aprendizajes Técnicos:

SimPy es una herramienta poderosa y elegante para SED. Su paradigma basado en generadores y eventos simplifica enormemente la modelación de sistemas dinámicos concurrentes.

La separación clara entre lógica de negocio (backend) e interfaz (frontend) es fundamental para la mantenibilidad y la escalabilidad del software.

La estocasticidad bien calibrada añade realismo, pero también requiere múltiples ejecuciones y análisis estadístico robusto para extraer conclusiones confiables.

Python, con su ecosistema de bibliotecas (SimPy, Tkinter, random), es un lenguaje excepcionalmente adecuado para prototipado rápido y desarrollo de herramientas de simulación académicas y profesionales.

Aprendizajes de Modelado:

La simplificación es necesaria, pero debe ser consciente. El modelo actual hace supuestos (independencia de métricas, curvas de crecimiento universales) que son aproximaciones útiles, pero cuyas limitaciones deben ser comunicadas al usuario.

La validación contra datos reales es un paso crítico para cualquier modelo de simulación con pretensiones de aplicación práctica. En este proyecto, la validación fue conceptual (coincidencia con tendencias documentadas); una versión profesional requeriría un proceso riguroso de backtesting con datos históricos.

La interactividad y la visualización no son "lujos", sino componentes esenciales para la adopción de herramientas analíticas por usuarios no técnicos.

Aprendizajes de Trabajo en Equipo y Gestión de Proyectos:

La documentación temprana y continua (como la reflejada en "A1T5. Documentación.docx") es vital para alinear al equipo, definir alcances y facilitar la integración final.

La división de tareas según fortalezas (uno enfocado en la lógica de simulación, otro en la GUI, otro en el análisis de resultados) permite una ejecución más eficiente y un producto final más pulido.

La revisión crítica entre pares del código y los resultados enriquece el proyecto, identificando errores y oportunidades de mejora que un solo desarrollador podría pasar por alto.

6.3. Impacto Potencial y Aplicabilidad

Este simulador, aunque desarrollado en un contexto académico, tiene un **claro potencial de transferencia a entornos profesionales**:

Clubes de fútbol (desde divisiones inferiores hasta élite): Como herramienta de apoyo en los departamentos de scouting y análisis de rendimiento.

Agencias de representantes de jugadores: Para valorar y proyectar el desarrollo de sus representados, optimizando estrategias de carrera y negociaciones contractuales.

Medios de comunicación y analistas deportivos: Para generar contenido basado en datos y realizar comparativas objetivas entre jugadores.

Entrenadores y preparadores físicos: Para planificar cargas de trabajo y expectativas de desarrollo individualizadas.

La **escalabilidad y modularidad** del código base lo convierten en un excelente punto de partida para proyectos más ambiciosos, como sistemas de simulación multi-agente para equipos completos o modelos predictivos integrados con técnicas de machine learning.

6.4. Reflexión Final

La simulación computacional se erige como un **punto indispensable** entre la teoría deportiva y la toma de decisiones práctica en un entorno de alta incertidumbre y elevadas apuestas económicas. Este proyecto no solo ha cumplido con los requisitos académicos de la asignatura de Simulación, sino que ha logrado **materializar un concepto teórico en una herramienta software funcional y con aplicabilidad real**.

El camino forward, delineado en las propuestas de mejora, señala hacia un futuro donde la **analítica deportiva de vanguardia** esté al alcance no solo de los clubes más ricos, sino de cualquier organización que valore la toma de decisiones basada en evidencia. Este trabajo constituye un paso significativo en esa dirección, demostrando que con sólidos fundamentos teóricos, herramientas de software apropiadas y un enfoque metódico, es posible construir soluciones inteligentes para problemas complejos del mundo real.

7. REFERENCIAS (Formato APA)

Banks, J., Carson, J. S., Nelson, B. L., & Nicol, D. M. (2014). *Discrete-event system simulation* (5th ed.). Pearson.

Bradley, P. S., & Noakes, T. D. (2013). Match running performance fluctuations in elite soccer: indicative of fatigue, pacing or situational influences? *Journal of Sports Sciences*, 31(15), 1627–1638. <https://doi.org/10.1080/02640414.2013.796062>

Carling, C., Le Gall, F., & Dupont, G. (2012). Are physical performance and injury risk in a professional soccer team in match-play affected over a prolonged period of fixture congestion? *International Journal of Sports Medicine*, 33(01), 36–42. <https://doi.org/10.1055/s-0031-1283190>

FIFA. (2022). *FIFA Transfer Matching System Report*. Fédération Internationale de Football Association.

Matloff, N. (2008). *Introduction to Discrete-Event Simulation and the SimPy Language*. University of California, Davis, Dept of Computer Science. Retrieved from <https://simpy.readthedocs.io/>

Meyer, D., & Charnley, E. (2021). *Football Analytics with Python & R*. Manning Publications.

PyPI. (2023). SimPy: Discrete event simulation for Python. *Python Package Index*. <https://pypi.org/project/simpy/>

Rein, R., & Memmert, D. (2016). Big data and tactical analysis in elite soccer: future challenges and opportunities for sports science. *SpringerPlus*, 5(1), 1410. <https://doi.org/10.1186/s40064-016-3108-2>

UEFA. (2021). *The European Club Footballing Landscape*. Union of European Football Associations.

Wasserstein, R. L., & Lazar, N. A. (2016). The ASA statement on p-values: context, process, and purpose. *The American Statistician*, 70(2), 129–133. <https://doi.org/10.1080/00031305.2016.1154108>