

# AiSport (Sistema de gestión y análisis de rendimiento deportivo)

Integrantes:

Magaña López Miguel Ángel, Robles Pulido Efraín

Asesor:

Morales Ramirez Thelma Isabel

CENTRO UNIVERSITARIO DE CIENCIAS EXACTAS E

INGENIERÍAS, (CUCEI, UDG)

miguel.magana0967@alumnos.udg.mx

efrain.robles5009@alumnos.udg.mx

thelma.morales3416@academicos.udg.mx

**Abstract** — AiSport is a web application that combines the management of competitive events with the continuous evaluation of the performance of basketball athletes, integrating technologies such as machine learning for pattern recognition and sports analysis. This document describes the web application's technical architecture and development methodology, emphasizing areas of basketball sports improvement based on physical capabilities.

**Palabras claves** – Gestión de eventos deportivos en baloncesto, análisis deportivo, OpenCV, seguimiento de rendimiento, clasificación de deportistas, recomendación de áreas de mejora.

## Repositorios de Código Fuente:

AI. <https://github.com/StrangerPlay17/AiSport/tree/SportAI>  
Backend. <https://github.com/EfrainRP/AiSportBackend>  
Frontend. <https://github.com/EfrainRP/AiSportFrontend>

**Versión Actual del Código:** AiSport 2.0

**Licencia Legal del Código:** Código Abierto.

## I. INTRODUCCIÓN

El interés y crecimiento del baloncesto continúan expandiéndose cada día en el mundo. En México, existe una liga profesional de baloncesto que ha ganado reconocimiento internacional. Esta liga cuenta con un sitio web muy completo, donde se publican noticias relacionadas con el deporte, información sobre los jugadores que participan en el torneo, estadísticas detalladas de los equipos y los jugadores, así como la tabla de posiciones de la liga. Este desarrollo se debe al creciente éxito y popularidad del torneo oficial organizado por la Federación Mexicana de Baloncesto.

Pero, ¿qué pasa con los torneos no oficiales pero sí documentados?

En México se llevan a cabo varios torneos nacionales no oficiales que se organizan dentro de una misma zona o ámbito municipal. La mayoría de estos torneos utilizan métodos tradicionales para registrar estadísticas, como hojas de cálculo, PDFs, o imágenes que solo muestran puntuaciones generales del torneo y de los equipos, sin

incluir tecnologías avanzadas o datos detallados sobre estadísticas y los jugadores individuales

Este método de distribución de estadísticas no suele ser eficiente, debido a que puede estar sujeto a errores humanos durante su elaboración, constante tiempo y de revisiones con correcciones continuas, además de dificultar la recopilación precisa de datos.

En este artículo se presenta el sistema AiSport, una solución innovadora en forma de aplicación web diseñada para automatizar la gestión de torneos de baloncesto junto con el proceso de gestión de estadísticas de equipos e individuales. Con AiSport, puedes organizar y gestionar tus torneos, mostrando estadísticas generales del torneo, tablas de posiciones, partidos, datos de equipos, análisis del rendimiento individual en tiempo real. Esto en un ámbito semi-profesional integrado en un solo sistema.

### A. Planteamiento del problema

Dado la dependencia de métodos tradicionales y manuales especialmente en el crecimiento del baloncesto actual, es evidente que pueden ser propensos a errores humanos, requerir mucho tiempo de revisión, corrección y estar enfocados únicamente a la gestión de torneos donde la mayoría de plataformas online brindan acceso. La falta de automatización limita la capacidad de los organizadores, entrenadores y jugadores para tomar decisiones informadas basadas en estadísticas confiables por asociaciones oficiales.

### B. Justificación

AiSport se centra en integrar el ámbito de la gestión de torneos de baloncesto con la automatización de estadísticas basadas en métricas oficiales por la NBA (Asociación Nacional de Baloncesto) mediante un monitoreo de alta precisión de los entrenamientos de los jugadores de sus equipos a través del tiempo, lo que la lleva una supervisión continua de su rendimiento de forma precisa, eficiente y accesible.

## II. TRABAJOS RELACIONADOS

Al explorar el mercado actual, se encuentran diversas alternativas las cuales intentan abordar la problemática de la gestión y organización de torneos de esports, así como el uso de AI para analizar videos y estadísticas en baloncesto y deportes. Sin embargo, en su mayoría, abordan el problema de manera individual (analizar o gestionar esports).

Entre las plataformas que se pueden acceder para el análisis de rendimiento y video, se encuentra Hudl, software enfocado a video análisis táctico sin incluir un análisis biométrico en vivo, lo cual podría resultar en predicciones no muy precisas debido a que los movimientos de los jugadores para ciertas técnicas en el tiempo también suman un factor clave en el rendimiento. Además, plataformas como Kinexon ofrecen necesariamente usar sensores en el cuerpo de los jugadores para medir su movimiento a partir de datos biomecánicos recogidos, el cual puede ser algo ineficiente en deportes con alto contacto físico debido al desgaste y la necesidad del recurso material de los sensores para su óptimo funcionamiento.

En cuanto a aplicaciones en gestión de torneos de baloncesto, deportes y eventos competitivos, se encuentran distintos competidores más accesibles en listados a continuación:

- Challonge: Enfocada a la creación y gestión de torneos, programación y seguimiento de resultados con limitaciones premium.
- Battlefy: Permite promocionar, comunicar y transmitir en tiempo real la gestión de torneos sobre videojuegos, integrando búsquedas básicas de torneos de videojuegos.
- Toornament: Dirigida a más personalización avanzada con mayor cantidad de tipos de eventos competitivos y gestión equipos en transmisión en vivo.

Sin embargo, únicamente ofrecen el servicio de la gestión esports sin incluir algún algoritmo de inteligencia artificial en la obtención de estadísticas o métricas, lo cual ambos tipos de competidores comparten una amplia similitud con nuestro sistema de forma no integrada en su mayoría con sutiles diferencias en su enfoque.

## III. DESCRIPCIÓN DEL DESARROLLO DEL PROYECTO MODULAR

### A. Hipótesis

La implementación de un sistema automatizado para la gestión de torneos y estadísticas de baloncesto mediante el uso de inteligencia artificial puede mejorar significativamente en comparación con los métodos tradicionales más usados del estado del arte. Lo cual se logrará integrando la gestión de torneos con distintos modelos de machine learning supervisados para una mejor eficiencia, usando redes neuronales convolucionales (CNN) para extraer características espaciales del entrenamiento y redes neuronales artificiales (ANN) para predicciones de rendimiento y mejora.

### B. Metodología

Inicialmente para comprender mejor la problemática de los usuarios y agilizar el proceso del desarrollo del proyecto, se implementó una combinación de las metodologías Design Thinking y SCRUM. De esta manera, en la fase Design Thinking se llevó a cabo una investigación y entrevistas con organizadores, entrenadores y jugadores del área deportiva de baloncesto para entender mejor la problemática actual, permitiendo identificar a detalle las necesidades principales del área.

Asimismo, como resultado de la investigación y lluvia de ideas, se concluyó la posible mejor solución la cual consiste en desarrollar una plataforma web que permita a los usuarios gestionar sus torneos y acceder a estadísticas y herramientas de análisis de rendimiento reconocidas por instituciones nacionales de baloncesto para monitorear su constante desempeño con mayor confiabilidad y precisión.

Por otra parte, se eligió el desarrollo ágil del proyecto con la metodología SCRUM debido a que permite entregar funcionalidades de la aplicación web de forma corta, tales como la gestión de torneos, equipos, partidos, notificaciones por rol y por último las estadísticas, con tal de poder recibir retroalimentación temprana, además de que permite la adaptación rápida a los cambios, esto es sumamente útil para ajustar estadísticas con distintas ponderaciones y su interacción con el sistema a lo largo del tiempo. Al igual que está metodología facilitó la colaboración entre los miembros del equipo como la priorización de funcionalidades básicas antes de las complejas para asegurar la calidad del producto.

Para la posterior implementación de las metodologías, se utilizaron herramientas de gestión de tareas y visualización como Trello y Git/GitHub con la finalidad de tener un seguimiento constante de los avances del proyecto.

### C. Diseño y Fase del Prototipo del Proyecto Modular

Una vez se han establecido las metodologías a seguir para el desarrollo del proyecto, se optó por aplicar la ingeniería de requerimientos para especificar la solución en que debe hacer el sistema y cómo debe hacerlo, con la finalidad de tener puntos clave que guíen al desarrollo del mismo.

#### Requerimientos Funcionales

- RF1 Log In y Sign Up de Usuarios: Los usuarios deben poder registrarse e iniciar sesión con sus credenciales para el uso del sistema.
- RF2 Gestión de Torneos: El sistema debe permitir a los usuarios registrar, editar, eliminar, visualizar cuadros de torneos y asignar equipos existentes con control de acceso autorizado.
- RF3 Gestión de Equipos: El sistema debe permitir a los usuarios registrar, editar, eliminar y visualizar sus equipos como asignar jugadores a un equipo con control de acceso autorizado.
- RF4 Gestión de Partidos: El sistema debe permitir el registrar, editar, eliminar y visualizar cada partido de torneos existentes.
- RF5 Notificaciones de Usuario por Rol: El sistema debe permitir a los usuarios el acceso a

notificaciones de acuerdo a su rol actual en la aplicación.

- RF6 Estadísticas: El sistema debe permitir la generación y visualización de estadísticas de torneos, equipos e individuales de forma automatizada a lo largo del tiempo.

#### Requerimientos No Funcionales

- RNF1 Usabilidad: La aplicación debe ser responsive para dispositivos móviles, intuitiva y fácil de usar.
- RNF2 Rendimiento: La aplicación debe manejar múltiples usuarios sin degradación del rendimiento y generar la conexión en tiempo real con baja latencia.
- RNF3 Seguridad: La aplicación debe garantizar la privacidad de los datos y el acceso autorizado por usuario y rol.
- RNF4 Escalabilidad: La aplicación debe soportar un aumento en el número de torneos y usuarios, además de contar con una base de datos optimizada en consultas eficientes.
- RNF5 Integración: La aplicación debe integrarse con APIs externas para su modularidad y funcionalidad de estadísticas.
- RNF6 Disponibilidad: La aplicación debe tener un tiempo de actividad constante para entrenamientos y consultas.

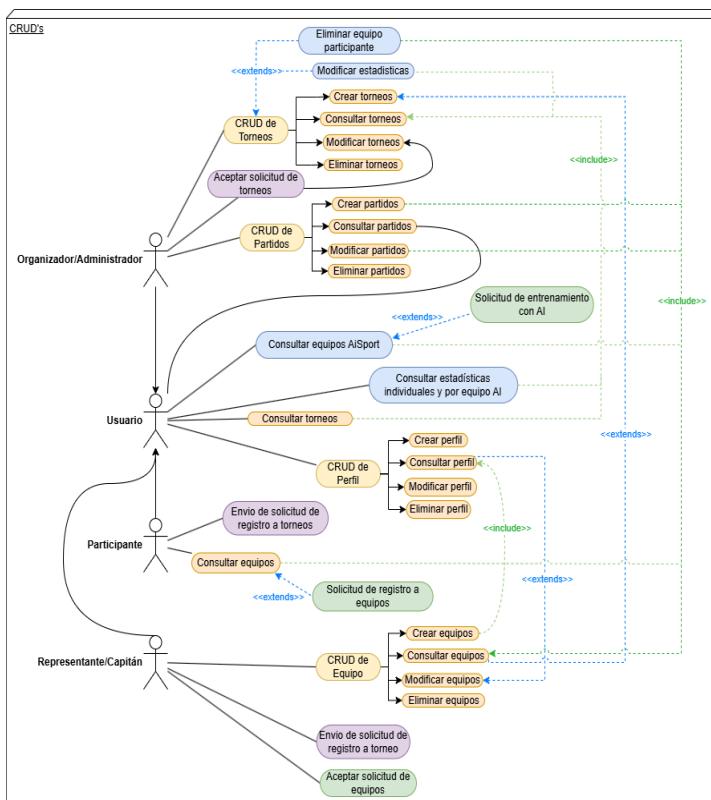


Fig. 1 Diagrama Caso de Uso del Sistema

Definiendo previamente los requerimientos, se optó por utilizar tecnologías ampliamente usadas en el mercado con alta compatibilidad de bibliotecas y tiempos de respuesta altos, entre estas, React para construir la interfaz de usuario (UI) usando componentes de Material UI y Vite. Además, se

eligió Node JS para tener un entorno de ejecución en el lado del servidor, a este entorno, se le sumó el framework de Express.js el cual permitió manejar rutas, middlewares y solicitudes HTTP para la construcción de APIs. Adicionalmente, se decidió hacer uso de la herramienta Prisma ORM (Object-Relational Mapping) compatible con Node JS para simplificar la interacción con la base de datos y mejorar la productividad.

En la base de datos, se eligió hacer uso de un modelo relacional para mantener los datos normalizados e interrelacionados, para este propósito se utilizó el gestor MySQL.

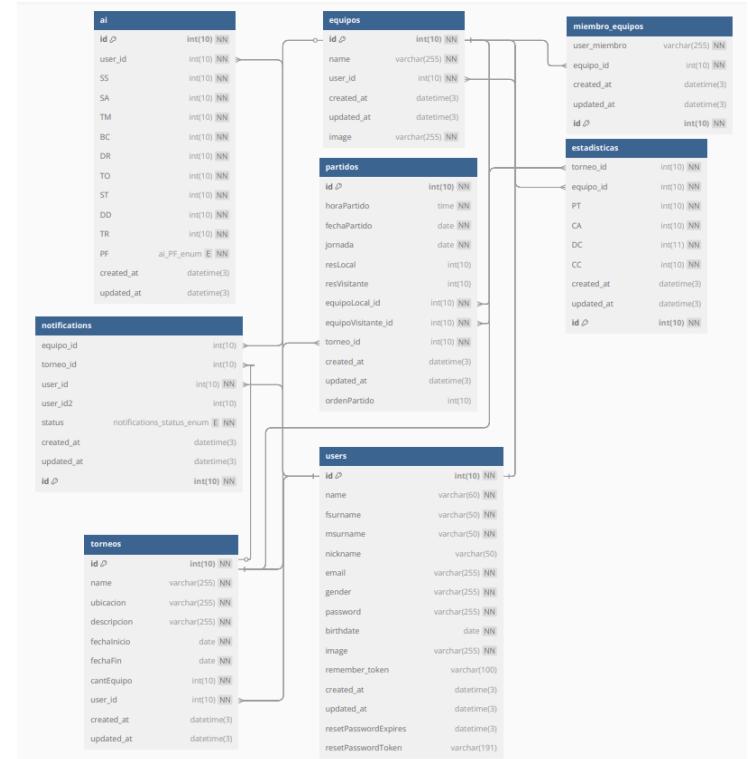


Fig. 2 Modelo Entidad-Relación. (MER 3-FN)

Se definió la arquitectura del proyecto con el patrón de diseño MVC (Model-View-Controller) con el fin de separar las responsabilidades de la aplicación, siendo estos:

- **Modelo:** Gestiona los datos, consultas a la base de datos por medio del ORM de Prisma.
- **Controlador:** Recibe las solicitudes del cliente (React y AI), ejecuta la lógica y responde por medio de las rutas y APIs configuradas a la vista.
- **Vista:** Es la interfaz de usuario, mostrando los datos solicitados, así como de capturar las entradas del usuario (React).

Se hizo uso de otro componente (servidor), siendo el encargado del procesamiento exclusivo de la IA, donde se procesan los datos (frames) en lo cual los resultados de las estadísticas son enviados y validados en el controlador en tiempo real para uso más eficiente de la aplicación web.

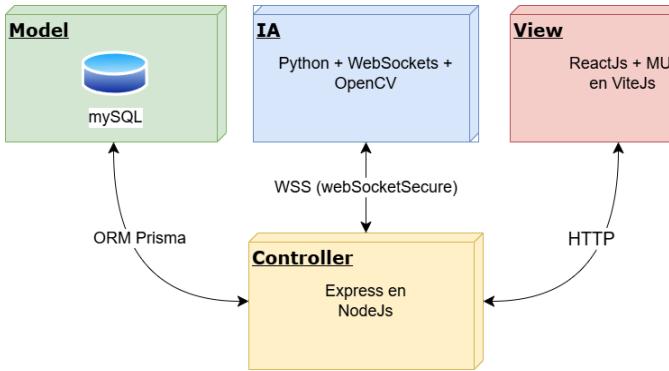


Fig. 3 Diagrama a Bloques de la Arquitectura del Sistema.

Por otra parte, en la implementación de la automatización de estadísticas, se optó por hacer uso de tecnologías y herramientas que permitan el uso de visión artificial como redes neuronales, tales como OpenCV, YOLOv8 y Tensorflow en el lenguaje de Python, debido a su alta compatibilidad e integración. Esta elección da acceso al uso de la herramienta WebSocket compatible con Python, la cual permite crear una API con el servidor de IA para establecer una comunicación bidireccional con el navegador y facilitar el intercambio de información de estadísticas en tiempo real.

Adicionalmente, se usaron herramientas como Axios para la realización de solicitudes HTTP de forma más eficiente y Asyncio para manejar las operaciones de entrada y salida de manera concurrente, con la finalidad de mantener la aplicación más escalable.

Para la implementación de la interfaz de usuario (Front-End-UI) se desarrolló con Material UI (MUI), una librería de componentes visuales con la finalidad de hacer más intuitiva la aplicación para usuarios destino, conteniendo componentes tales como botones, formularios, tarjetas, menús, llaves, sidebars que siguen las guías de diseño de Google, en donde fueron ajustados para la aplicación web. Se decidieron utilizar estas herramientas por su rendimiento, desarrollo modular, gran comunicación, documentación y compatibilidad con herramientas modernas como MUI. Permitiendo tener rápidas actualizaciones en la interfaz y de forma eficiente con el Virtual DOM de ReactJS, facilidades de trabajo en equipo, e incorporación de nuevas tecnologías respaldadas por documentación oficial.

Para la implementación del servidor (Back-End), se basó en una arquitectura Cliente-Servidor, donde ReactJS actúa como cliente que consume los servicios ofrecidos por el back-end a través de APIs. Configurado con núcleos, políticas, servicios internos, externos, cabeceras de seguridad, incluyendo helmet para la protección contra ataques XSS, Clickjacking y Sniffing. Adicionalmente, se trataron códigos de estado HTTP tales como el código 500 (Internal-Server-Error), 404 (Page-Not-Found), 403 (Forbidden) realizándose mediante middlewares para restringir rutas con policies, incluyendo para mejor protección, el uso del ORM de Prisma para realizar consultas SQL y evitar ataques Injection-SQL.

Por otra parte, el NodeJS ofrece consultas de la base de datos al cliente (ReactJS), esta comunicación es posible por medio de los endpoints(APIs) del back-end para mandar las consultas requeridas o procesadas al cliente, usando los

métodos esenciales de comunicación del protocolo HTTP (POST, PUT, GET, DELETE).

Además, ReactJS está configurado con un proxy que permite que solo se reciba información de un servidor destino para mayor seguridad mediante la librería Axios, la cual realiza peticiones HTTP basado en promesas para NodeJS y el navegador.

El sistema de análisis de rendimiento deportivo en tiempo real de la plataforma AiSport está diseñado para operar mediante una cámara o dispositivo encargado de capturar imágenes usando la biblioteca de OpenCV, esto con el fin de analizar la jugabilidad del jugador y obtener un acercamiento a su rendimiento.

Inicialmente para la detección satisfactoria de elementos clave en el entrenamiento como el balón, canasta y posiciones de jugadores, se procedió a realizar un etiquetado de imágenes de canastas y balones, obteniendo de esta forma un dataset de aproximadamente 3000 imágenes por clase para aumentar su generalización y variabilidad. Se utilizó una división del dataset por convicción en el aprendizaje de patrones en un 80% para entrenamiento y un 20% para prueba del modelo, usando el algoritmo de optimización del descenso de gradiente con AdamW, con tal de minimizar la función de pérdida durante el entrenamiento y el error en las predicciones.

Por otra parte, la implementación de detección de poses se realizó mediante el uso del modelo Pose-Ultralytics de YOLO pre entrenado con un dataset amplio el cuál permitió detectar un total de 17 puntos clave del cuerpo humano, mostrando en la siguiente figura (Fig. 4) las clases detectadas mediante predicciones.

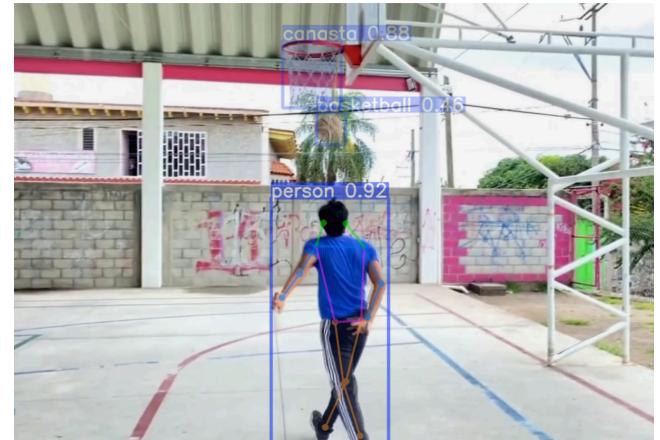


Fig. 4 Predicciones de modelos con YOLOv8 mediante OpenCV.

De esta manera, se cuentan con los modelos esenciales para la detección en un plano 2D, al proporcionar cajas delimitadoras que encierran los balones, canastas y poses detectadas en coordenadas (X,Y) donde son expresados en términos de píxeles dentro de la imagen.

El procesamiento de las imágenes se realizó mediante un método de determinación de intersección y transición en un sistema de coordenadas en un espacio 2D, en el cual inicialmente, se hace una predicción del modelo para obtener la posición del objeto detectado y de esta forma desempaquetar sus coordenadas directamente, teniendo la

localización en coordenadas (X, Y) del centro de la caja delimitadora. Por consiguiente, se aplica una normalización lineal a las coordenadas obtenidas con el ancho y el alto de la imagen, con la finalidad de transformar las coordenadas absolutas (en píxeles) a coordenadas relativas en un rango [0,1], lo cual proporciona un sistema de referencia escalable y compatible con distintas resoluciones de imagen, siendo de la siguiente manera:

$$x_{\text{norm}} = \frac{x}{W}, \quad y_{\text{norm}} = \frac{y}{H}$$



Fig. 5 Normalización lineal de coordenadas aplicado al modelo de detección de balones.

Visualizándose que (x, y) son las coordenadas en un punto a normalizar, "W" corresponde al ancho de la imagen en pixeles y "H" al alto respectivamente.

Al contar con las coordenadas previamente normalizadas, es posible realizar un enfoque simplificado de la distancia entre dos puntos basado en diferencias absolutas para detectar interacciones entre el jugador, la canasta y el balón, logrando calcular distancias en los ejes x e y por separado. Se optó por este método debido al bajo costo computacional que proporciona en donde se busca una fluidez estable para el analizador. Adicionalmente, se hace uso de comparaciones de umbrales en el espacio 2D con lógica condicional para determinar la métrica que tendrá un próximo recálculo en conteo, siendo de la siguiente manera condicional: (Si  $(\Delta x \geq \text{metrics\_threshold}_x) \vee (\Delta y \geq \text{metrics\_threshold}_y)$ ) | Si  $(\Delta x \leq \text{metrics\_threshold}_x) \wedge (\Delta y \leq \text{metrics\_threshold}_y)$ ) donde el umbral es distinto dependiendo de la métrica a calcular, por lo que, las diferencias absolutas de las poses son representadas de la siguiente forma en su cálculo:

$$\Delta x = |x_{\text{norm}} - x_{\text{ankle}}|$$

$$\Delta y = |y_{\text{norm}} - y_{\text{ankle}}|$$

$$\Delta x_{\text{wrist}} = |x_{\text{wrist}} - x_{\text{ball}}|$$

$$\Delta y_{\text{wrist}} = |y_{\text{wrist}} - y_{\text{ball}}|$$

De esta manera, el análisis de coordenadas permite calcular las métricas que en conjunto conformarán el rendimiento del jugador, por lo que el análisis se enfoca en recolectar especialmente métricas por su relevancia en baloncesto siguiendo un estándar de la Asociación Nacional de Baloncesto (NBA), indicando mediante paréntesis su tipo de clasificación en el deporte, considerando las siguientes:

1. Tiros anotados con éxito al (Presición)
2. Intentos de tiro a canasta (Volumen)
3. Duración del entrenamiento (Tiempo-resistencia)
4. Duración del balón sostenido (Control)
5. Dribbles (Agilidad)
6. Touches (Velocidad)
7. Pasos (Movilidad)
8. Double Dribbles (Violación de regla)
9. Travels (Violación de regla)

Para lo que, la métrica duración de entrenamiento y pasos no son registradas en estadísticas de juego en baloncesto de forma oficial. Sin embargo, para poder hacer del rendimiento una métrica medible a lo largo del entrenamiento y considerar la técnica de los jugadores en tiempos de ejecución cortos que simulan un partido real, estas son incluidas.



Fig. 6 Métricas analizadas en los modelos (CNN) con YOLOv8.

Una vez se recolectan las métricas mediante modelos de redes neuronales convolucionales (CNN) con YOLOv8, al finalizar el entrenamiento estas son enviadas como entrada a un modelo de redes neuronales artificiales (ANN) tipo MLP (Multi-Layer-Perceptron) el cual permitirá realizar una clasificación del rendimiento del jugador basado en sus métricas obtenidas.

Para este propósito, el modelo fue entrenado con un dataset de 100,000 registros con la finalidad de tener mayor variabilidad, de ese total un 20% es asignado para prueba y un 80% en entrenamiento para garantizar un buen equilibrio en el aprendizaje de patrones. En este modelo, se optó por usar la normalización Min-Max-Scaling, el cual es basada en la siguiente fórmula:  $x' = \frac{x - \text{xmin}}{\text{xmax} - \text{xmin}}$ . Dado un conjunto de datos con características X, donde cada valor pertenece a una característica  $X_j$ , así,  $x'$  es el valor normalizado,  $x$  significa el valor original y  $\text{xmin}$ ,  $\text{xmax}$  los valores mínimos-máximos de las características

en el conjunto de entrenamiento, por lo que este método asegura la distribución de los datos sin cambiar su forma.

Adicionalmente, se utilizan parámetros como Regularización L2 el cual agrega un término de penalización indicado por  $\lambda$ :  $\text{Loss} = \text{Loss original} + \lambda \sum w^2$  combinado con DropOut decreciente de (0.4–0.3) para evitar overfitting.

Con ajustes precisos de parámetros del modelo se consiguió una optimización con 99.85% de precisión y una pérdida en validación de 0.0045 sin afectar a la generalización.

```
Epoch 30/30
625/625 - 5s - loss: 0.0085 - accuracy: 0.9985 - val_loss: 0.0045
1/1 [=====] - 0s 302ms/step
1/1 [=====] - 0s 46ms/step
```

Fig. 7 Resultados de entrenamiento MLP usando Regularización L2..

Al haber concluido el entrenamiento, el modelo realizará clasificaciones de acuerdo a las siguientes 5 categorías posibles del rendimiento de un jugador para áreas de mejora, en base en sus métricas:

- Clase 0 = Deficiente
- Clase 1 = Mejorable
- Clase 2 = Bueno
- Clase 3 = Muy bueno
- Clase 4 = Excepcional

Posteriormente, con el propósito de proporcionar al usuario una retroalimentación con áreas de mejora en su entrenamiento, se realizó un sistema de sugerencias basado en aquellas métricas en las que el jugador tuvo un rendimiento más deficiente, con la finalidad de que pueda mejorar su desempeño. Para este procedimiento, se llevó a cabo un análisis del promedio de los rangos y evaluación de las métricas obtenidas en un determinado tiempo de entrenamiento por distintos jugadores de baloncesto experimentados, promedio y novatos en el deporte.

Dentro de este análisis, se encontraron principalmente 10 tipos de rangos de tiempo base en métricas del entrenamiento (rangos que van en 30 seg hasta llegar a 5 min de entrenamiento como máximo) donde métricas como dribbles pueden tener un impacto positivo en el rendimiento pero métricas como double dribbles un impacto negativo. Dado estos casos, se encontraron principalmente los siguientes para poder una clasificación de sugerencias:

- El jugador tuvo mucha agilidad, pero no precisión.
- El jugador no tuvo agilidad, pero obtuvo precisión.
- El jugador no tuvo agilidad ni precisión.
- El jugador obtuvo tanto habilidad como precisión.

De esta manera, se adaptan los 10 tipos de rangos en base a las métricas promedio que pueden obtener los jugadores en un determinado lapso de tiempo de entrenamiento. Posterior a eso, se hace una comparación de las métricas obtenidas del jugador con las métricas promedio en un tiempo de

entrenamiento particular y se elige aquella más deficiente para hacer una sugerencia.

En el transcurso del entrenamiento, está información es enviada al navegador usando el protocolo WebSocket RFC 6455, en donde mediante excepciones se obtienen los códigos de cierre de conexión para tolerancia a fallos y manejar conexiones desconectadas por el servidor (1001 Going Away), errores inesperados (1011 Internal Error) y conexiones cerradas de forma controlada (1000 Normal Closure) principalmente. Por otra parte, se hace uso de Asyncio para que el servidor pueda manejar múltiples conexiones de manera concurrente sin bloquearse. De este modo, al concluir se envía la predicción del rendimiento y sugerencia de mejora al usuario. Cabe destacar que debido a la complejidad de los modelos de detección en pesos y el alto costo computacional que requieren, en esta versión 2.0 de AiSport el usuario debe esperar un tiempo determinado para entrenar si otro usuario se encuentra en entrenamiento de forma concurrente. Por lo que, la conexión usando el protocolo WebSocket y el navegador es representada mediante la siguiente figura. (Fig. 8)

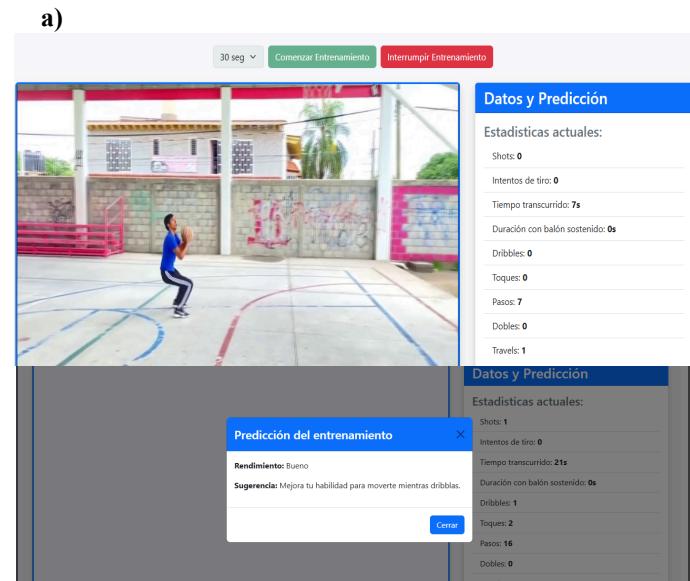


Fig. 8 Conexión con navegador mediante WebSockets. La figura a) representa una conexión iniciada exitosa. La figura b) muestra una conexión finalizada y el envío de la predicción de rendimiento al usuario.

Por otra parte, la distribución de la plataforma web se llevó a cabo mediante tecnologías como Ngrok con VSCode Port Forwarding en el servidor AI, acelerando los servicios de red debido a su soporte con WebSocket Seguro (WSS), accesibilidad de integración, bajo costo y permitiendo reducir la latencia al contar con servidores en California U.S.A, adicionalmente, se cuenta con Railway para un escalado vertical y alojamiento de la base de datos en un entorno flexible. De esta manera, se garantiza una mayor calidad del servicio, disponibilidad y escalabilidad sin que el rendimiento sea comprometido.

A continuación, se utilizó Firebase para el despliegue del backend en Node.js y el frontend en React.js, facilitando una integración ágil y moderna con servicios de autenticación, hosting estático, notificaciones push y funciones en la nube.

Firebase ofrece una infraestructura escalable y confiable con una interfaz intuitiva, permitiendo un despliegue continuo (CI/CD), seguridad con reglas personalizables y un entorno sin servidor (serverless) debido a sus funciones en la nube.

Esta integración con Railway para la base de datos permite separar la lógica del backend del almacenamiento, otorgando mayor control, modularidad y flexibilidad al proyecto, otorgando una sinergia entre servicios para mejorar la experiencia de desarrollo y el rendimiento del sistema completo. De esta manera, el paso de entorno local a entorno de producción en la aplicación AiSport se realizó de forma progresiva, iniciando con pruebas en un entorno de desarrollo local, donde se validó la funcionalidad de la inteligencia artificial, la estructura del backend y la integración con el frontend.

Una vez superadas las etapas de prueba y validación, se procedió al despliegue en producción utilizando Firebase y Railway como plataformas principales. La decisión de migrar a este entorno de producción se basó en varios factores clave: la facilidad para implementar cambios mediante CI/CD, el bajo tiempo de respuesta de los servidores de Firebase, el soporte nativo para HTTPS, WebSocket y funciones serverless, así como la estabilidad y escalabilidad de Railway para manejar la base de datos. Esta arquitectura permitió que AiSport contará con una mayor disponibilidad y rendimiento, asegurando que los usuarios pudieran interactuar con el sistema en tiempo real sin interrupciones, lo cual es crítico para una plataforma basada en análisis en vivo y automatización de estadísticas mediante inteligencia artificial.

#### *Pruebas del Sistema*

Para comprobar la disponibilidad, integridad y confiabilidad de la información en la plataforma, se realizaron pruebas multinivel entre ellas unitarias, de integración, rendimiento y usabilidad, concluyendo con pruebas de validación y defecto. Esto para asegurar que AiSport sea una aplicación funcional, accesible y de calidad, contando principalmente con las siguientes enlistadas:

- Pruebas de validación en campos de login y registro, acceso permitido en credenciales correctas y verificación de sesión activa mediante token.
- Pruebas de validación y defecto en operaciones CRUD's de forma aislada, como gestión de usuarios autorizados para modificación y visualización de la información mediante policies/middlewares.
- Pruebas de integración de usuarios, torneos, equipos y partidos mediante Thunder Client y Prisma Studio.
- Pruebas de integración de API's y manejo de errores mediante Postman.
- Pruebas de rendimiento en usuarios simultáneos y transmisión de estadísticas con un retraso entre 50-150ms.
- Pruebas de compatibilidad en dispositivos móviles y navegadores mediante React Responsive.

- Pruebas de usabilidad y evaluación UX/UI con usuarios del ámbito deportivo del baloncesto para el diseño de navegación e interfaz.

#### *Módulo I Justificación de Arquitectura y Programación de Sistemas*

En este módulo, se utilizaron lenguajes de programación tales como (HTML, CSS, JS, Python) debido a su alta compatibilidad con librerías y uso en el ámbito del mercado. Además, se usaron metodologías ágiles como SCRUM y Design Thinking para innovar y entregar valor de manera ágil a los usuarios. Por otra parte, se usó un mapeo relacional de objetos mediante Prisma ORM para simplificar el acceso a la base de datos mediante modelos esquemáticos y control de versiones de código en Git para separar las partes del proyecto de acuerdo al patrón de diseño MVC. Para ello en este módulo se hizo uso de la ingeniería de requerimientos para conocer la complejidad del proyecto, los recursos disponibles, alcance, escalabilidad y marcar objetivos.

#### *Modulo II Justificación de Sistemas Inteligentes*

En este módulo, se implementó un sistema de visión artificial utilizando modelos de YOLOv8 con redes neuronales convolucionales (CNN) para detección de articulaciones, balones y canastas, junto con un modelo de redes neuronales artificiales (ANN) tipo MLP (Multi-layer Perceptron) para reconocer patrones en las estadísticas obtenidas y así clasificar el rendimiento de un jugador. Se optó por este enfoque debido a que YOLOv8 proporciona técnicas avanzadas como Non-Maximum Suppression (NMS) y data augmentation, lo que garantiza una alta precisión en la detección al entrenar, además de su capacidad de detectar varios objetos en una sola imagen. Adicionalmente, se consideró un modelo MLP por su capacidad de modelar relaciones no lineales entre las estadísticas y el rendimiento del jugador, especialmente al permitir clasificar el rendimiento de los jugadores en categorías y aprender a generalizar nuevas métricas. Esta combinación de técnicas permitió conseguir un alto nivel de precisión en predicciones del análisis de entrenamiento.

#### *Módulo III Justificación de Sistemas Distribuidos*

En este módulo, se empleó una arquitectura distribuida para separar responsabilidades y mejorar la escalabilidad mediante tres servidores. Por lo que se hizo uso de la arquitectura cliente-servidor para permitir la distribución de tareas, recursos y facilitar la interoperabilidad. Además, se implementó un servidor mediante WebSockets el cual, permitió tener la transmisión bidireccional de información continua, siendo utilizado para poder integrar el analizador de rendimiento en tiempo real y enviar estadísticas y sugerencias a los clientes. Por este motivo, el protocolo WebSocket con comunicación full-duplex utilizado mejoró considerablemente la eficiencia en la transmisión de estadísticas y análisis de video en vivo, logrando una muy baja latencia al eliminar la sobrecarga de establecer una nueva conexión para cada solicitud. Adicionalmente, se utilizó la arquitectura RESTful para la implementación de APIs.

#### IV. RESULTADOS OBTENIDOS DEL PROYECTO

Al haber llevado a cabo el entrenamiento de los respectivos modelos, se observó una matriz de confusión con (399 TP, 8 FP, 9 FN) promedio de casos detectados para los modelos de detección YOLOv8 más un ajuste de umbral de confianza mayor al 60% minimizando los falsos positivos, asimismo, una matriz de confusión de (3993 TP, 7 FP, 7 FN) promedio para el modelo de clasificación MLP indicando alta precisión.

Por otra parte, al realizarse las pruebas unitarias a los módulos de gestión de torneos, equipos, estadísticas y partidos dieron resultados satisfactorios, como las pruebas de integración entre los diferentes módulos y servidores para su interacción y comunicación se obtuvo una baja latencia en la transmisión en tiempo real de estadísticas.

Por lo que, en la parte de cuentas de usuarios y roles, se hizo uso de policies y middlewares los cuales permiten el control al acceso no autorizado de información, además de validación de requests por parte del backend para la prevención de ataques SQL Injection como XXS.

El proceso de entrenamiento y la integración de ambos modelos de redes neuronales convolucionales (CNN) con redes neuronales artificiales (ANN) logró clasificar con alta precisión el rendimiento jugadores de baloncesto en donde se integró un umbral de tiempo para hacer del rendimiento una métrica medible a través del entrenamiento, dando así, aproximaciones de jugabilidad no únicamente en tácticas de baloncesto sino también en movilidad y en períodos de ejecución.

Se alcanzaron varios objetivos claves en la finalización:

- Automatización de la gestión de torneos, equipos, partidos y estadísticas.
- Interfaz de usuario intuitiva y sistema modular tolerante a fallos.
- Sistema de notificaciones integrado por rol.
- Gestión de estadísticas integrado por medio de IA en tiempo real con baja latencia.

#### V. CONCLUSIONES Y TRABAJO A FUTURO

En conclusión, el proyecto cumplió con su propósito establecido en su desarrollo, el cual fue automatizar la gestión de eventos competitivos de baloncesto y obtención de estadísticas mediante el análisis del rendimiento de un jugador, donde fue posible mejorar la administración del deporte al reducir considerablemente errores humanos, optimizar estrategias y el tiempo manual con resultados significativamente precisos en comparación con distintos métodos documentados con modelos de redes convolucionales (CNN) y redes artificiales (ANN) por separado, destacando la importancia de explorar distintas técnicas que puedan trabajar en conjunto para crear un todo.

Para futuras implementaciones a largo plazo, se considera la expansión hacia otros ámbitos competitivos y disciplinas, esto es tomando en cuenta su gestión y respectivas métricas oficiales del deporte o evento competitivo con la finalidad de hacerlo lo más profesional y preciso posible en su área a fin. Además, mediante tecnologías más potentes, recursos o el uso de GPU (Graphics Processing Unit) en la nube con servidores distribuidos en puntos clave, sería posible aplicar

métodos aún más precisos que requieran un mayor costo computacional y reducir la latencia desde la mayoría de puntos geográficos, pudiendo considerarse un análisis de entrenamiento por equipos con alta precisión mediante enfoques como redes neuronales recurrentes (LSTM), lo que permitiría aprender patrones de juego o uso de la regresión cuadrática la cual sería capaz de detectar trayectorias y dar predicciones con métricas actuales en disciplinas que lo requieran. Adicionalmente, servicios como OpenAI permitirían automatizar y escalar el sistema de sugerencias en áreas de mejora más detalladamente y obtener un análisis más profesional. De igual manera, se puede considerar un crecimiento de la plataforma a otros dispositivos por medio de una aplicación móvil que esté interconectada al mismo sistema web y compatibilidad con distintas cámaras (baja-alta resolución) con la finalidad de una mejor visualización, acceso y disponibilidad de la aplicación.

#### RECONOCIMIENTOS

Deseamos expresar nuestro más sincero agradecimiento a nuestra asesora Morales Ramirez Thelma Isabel por habernos apoyado incondicionalmente en el desarrollo del proyecto, además de proporcionarnos una lluvia de ideas, interés, experiencia, dedicación y sobre todo motivación de forma perseverante lo cual fue el pilar del éxito del proyecto.

#### REFERENCIAS

- [1] MUI: The React component library you always wanted. (s/f). Mui.com. Recuperado el 2 de febrero de 2025, de <https://mui.com>
- [2] Vite. (s/f). Vitejs. Recuperado el 2 de febrero de 2025, de <https://vite.dev>
- [3] Prisma. (s/f). Prisma. Recuperado el 2 de febrero de 2025, de <https://www.prisma.io>
- [4] Challonge.com. [En línea]. Disponible en: <https://challonge.com/es> [Consultado: 2-feb-2025].
- [5] “Toornament - Software de gestión de torneos de Esports”, Toornament. [En línea]. Disponible en: <https://www.toornament.com/es/> [Consultado: 2-feb-2025].
- [6] “NBA Fantasy”. NBA Fantasy. Accedido el 2 de febrero de 2025. [En línea]. Disponible: <https://es.nbatruefantasy.nba.com/statistics>
- [7] NBA ID. “¿Se puede medir el rendimiento de un jugador NBA con un solo número? Estas estadísticas lo intentan.” NBA ID - Reportajes: ¿Se puede medir el rendimiento de un jugador NBA con un solo número? Accedido el 2 de febrero de 2025. [En línea]. Disponible: <https://spain.id.nba.com/reportajes/medir-rendimiento-jugador-nba>
- [8] Ultralytics. “Pose”. Home - Ultralytics YOLO Docs. Accedido el 2 de febrero de 2025. [En línea]. Disponible: <https://docs.ultralytics.com/es/tasks/pose/>
- [9] “Official NBA Stats | Stats | NBA.Com”. NBA.Com. Accedido el 2 de febrero de 2025. [En línea]. Disponible: <https://www.nba.com/stats>
- [10] “NBA Rulebook”. NBA Official. Accedido el 2 de febrero de 2025. [En línea]. Disponible: <https://official.nba.com/rulebook/>
- [11] “Home”. OpenCV. Accedido el 2 de febrero de 2025. [En línea]. Disponible: <https://opencv.org/>
- [12] “Players Traditional | Stats | NBA.com”. The official site of the NBA for the latest NBA Scores, Stats & News. | NBA.com. Accedido el 2 de febrero de 2025. [En línea]. Disponible: <https://www.nba.com/stats/players/traditional>
- [13] Ultralytics. “Modelos compatibles con Ultralytics”. Home - Ultralytics YOLO Docs. Accedido el 2 de febrero de 2025. [En línea]. Disponible: <https://docs.ultralytics.com/es/models/#why-should-i-use-ultralytics-hub-for-machine-learning-projects>
- [14] Battlefy, “Battlefy”, Battlefy.com. [En línea]. Disponible en: <https://battlefy.com> [Consultado: 2-feb-2025].

- [15] “Herramientas de análisis del rendimiento para equipos deportivos y atletas | Hudl”. Accedido el 2 de febrero de 2025. [En línea]. Disponible: <https://es.hudl.com/es-xl>