



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



TFG del Grado en Ingeniería
Informática

StatsGlowMind



Presentado por Roberto Lastras Santos
en Universidad de Burgos — 10 de junio
de 2024

Tutor: José Manuel Galán Ordax



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



D. José Manuel Galán Ordax, profesor del departamento de Ingeniería de Organización, área de Organización y Empresa.

Expone:

Que el alumno D. Roberto Lastras Santos, con DNI 70834970F, ha realizado el Trabajo final de Grado Ingeniería Informática titulado StatsGlowMind.

Y que dicho trabajo ha sido realizado por el alumno bajo la dirección del que suscribe, en virtud de lo cual se autoriza su presentación y defensa.

En Burgos, 10 de junio de 2024

Vº. Bº. del Tutor:

D. José Manuel Galán Ordax

Resumen

Este proyecto consiste en el desarrollo de una aplicación web dedicado a la Asociación Nacional de Baloncesto (NBA), la cual proporciona información de la NBA (clasificación, playoffs, partidos, estadísticas de los equipos y de los jugadores, etc) en tiempo real, utilizando APIs externas. También realiza un análisis del rendimiento de los jugadores, mediante estadísticas y gráficas. Además, predice los resultados de los partidos, utilizando un modelo Random Forest de Machine Learning, de modo que ofrece a los usuarios recomendaciones sobre qué equipos tienen más probabilidades de ganar en enfrentamientos futuros.

Esta aplicación web esta desarrollada en Angular mediante TypeScript, HTML y CSS; y utiliza la herramienta de Firebase como plataforma de alojamiento para la base de datos, almacenamiento de archivos, autenticación de usuarios, desarrollo backend mediante funciones (Python) en la nube y realiza el servicio de hosting.

Descriptores

Aplicación NBA, aplicación Web, baloncesto, Angular, Firebase, estadísticas, APIs externas, información en tiempo real, análisis de datos, gráficos de datos, Chart.js, predicción de partidos, Cloud Functions, Random Forest, internacionalización, ...

Abstract

This project involves the development of a web application dedicated to the National Basketball Association (NBA), which provides real-time information about the NBA (standings, playoffs, games, team and player statistics, etc.) using external APIs. It also performs player performance analysis through statistics and graphs. Additionally, it predicts game outcomes using a Random Forest machine learning model, offering users recommendations on which teams are more likely to win future matchups.

This web application is developed in Angular using TypeScript, HTML, and CSS; and it utilizes Firebase as a platform for database hosting, file storage, user authentication, backend development with cloud functions (Python), and hosting services.

Keywords

NBA application, web application, basketball, Angular, Firebase, statistics, external APIs, real-time information, data analysis, data graphs, Chart.js, game prediction, Cloud Functions, Random Forest, internationalization, ...

Índice general

| | |
|---|-----------|
| Índice general | iii |
| Índice de figuras | v |
| Índice de tablas | vi |
| 1. Introducción | 1 |
| 1.1. Estructura de la memoria | 2 |
| 1.2. Estructura de los anexos | 3 |
| 2. Objetivos del proyecto | 5 |
| 2.1. Objetivos generales | 5 |
| 2.2. Objetivos técnicos | 6 |
| 2.3. Objetivos personales | 7 |
| 3. Conceptos teóricos | 9 |
| 3.1. NBA | 9 |
| 3.2. API | 11 |
| 3.3. Protocolo HTTP | 12 |
| 3.4. Random Forest | 12 |
| 3.5. NoSQL | 14 |
| 4. Técnicas y herramientas | 15 |
| 4.1. Angular | 15 |
| 4.2. Firebase | 16 |
| 4.3. TypeScript | 18 |
| 4.4. HTML y CSS | 18 |

| | |
|---|-----------|
| 4.5. Visual Studio Code | 19 |
| 4.6. Bootstrap | 19 |
| 4.7. Angular Material | 20 |
| 4.8. Chart.js | 21 |
| 4.9. GitHub | 21 |
| 4.10. Trello | 22 |
| 4.11. Zotero | 23 |
| 4.12. LaTeX | 23 |
| 4.13. RapidAPI | 24 |
| 4.14. Microsoft Copilot | 24 |
| 4.15. Canva | 25 |
| 4.16. Flaticon | 25 |
| 4.17. Freepik | 25 |
| 4.18. Draw.io | 26 |
| 4.19. Figma | 26 |
| 4.20. Postman | 26 |
| 5. Aspectos relevantes del desarrollo del proyecto | 27 |
| 6. Trabajos relacionados | 35 |
| 6.1. Hoops Stats | 35 |
| 6.2. Basketball Reference | 36 |
| 6.3. PROBALLERS | 37 |
| 6.4. Tabla comparativa | 39 |
| 7. Conclusiones y Líneas de trabajo futuras | 41 |
| 7.1. Conclusiones | 41 |
| 7.2. Líneas de trabajo futuras | 42 |
| Bibliografía | 45 |

Índice de figuras

| | |
|---|----|
| 3.1. Algoritmo Random Forest | 13 |
| 5.1. URL de las APIs. | 30 |
| 5.2. Entrenamiento del Modelo. | 31 |
| 5.3. Servicio PredictionService. | 32 |
| 6.1. Estadísticas de los jugadores. | 36 |
| 6.2. Juego interactivo. | 37 |
| 6.3. Página inicio. | 38 |

Índice de tablas

| | |
|---|----|
| 3.1. Equipos de la Conferencia Este. | 11 |
| 3.2. Equipos de la Conferencia Oeste. | 11 |
| 6.1. Tabla comparativa de funcionalidades | 39 |

1. Introducción

En el panorama deportivo contemporáneo, la tecnología desempeña un papel fundamental al proporcionar herramientas que no solo aumenta la experiencia del aficionado, sino que también aportan estadísticas e información valiosa para equipos, entrenadores y analistas.

En este contexto, este trabajo se centra en el desarrollo de una aplicación web dedicada a la Asociación Nacional de Baloncesto (NBA), una plataforma que va más allá de la simple exhibición de estadísticas y resultados, para ofrecer una serie de funcionalidades diseñadas para informar, analizar y predecir.

Históricamente, la comprensión del desempeño deportivo se basaba en datos estáticos y en análisis subjetivos. Sin embargo, con la incorporación de las tecnologías web y la disponibilidad de datos en tiempo real, surge la oportunidad de transformar radicalmente la forma en que se interpreta y se interactúa con el mundo del baloncesto profesional.

Este proyecto aborda una serie de desafíos y oportunidades clave en el ámbito del análisis deportivo y la experiencia del usuario. Utilizando Angular y Firebase como marco tecnológico, junto con la integración de varias APIs, la aplicación ofrece al usuario de una forma dinámica y accesible, adentrarse al mundo de la liga de baloncesto más prestigiosa del mundo, la NBA.

Uno de los aspectos más destacados de esta aplicación es su capacidad para proporcionar análisis de jugadores en tiempo real, aprovechando las estadísticas de los jugadores en cada partido de la temporada, se realizan gráficas de análisis que muestran el rendimiento de los jugadores durante la temporada de la liga.

Además, este proyecto va más allá, al ofrecer predicciones sobre los resultados de los partidos futuros que puede seleccionar el usuario. Integrando un modelo con el algoritmo Random Forest de Machine Learning entrenado con datos históricos de la temporada actual, la aplicación proporciona a los usuarios la probabilidad de victoria de cada equipo, agregando más interés y emoción al seguimiento de la temporada de la NBA.

En resumen, esta aplicación representa una mezcla entre el fanatismo por el baloncesto con la tecnología moderna. Al proporcionar una plataforma integral para la información, el análisis y la predicción, busca mejorar la experiencia de los aficionados y apoyar la toma de decisiones estratégicas a los equipos, entrenadores y analistas.

1.1. Estructura de la memoria

- **Introducción:** Descripción general del propósito y contexto del proyecto y estructura de la memoria y de los anexos.
- **Objetivos del proyecto:** Listado de los objetivos generales y técnicos que se pretenden alcanzar con el proyecto, además de una lista de objetivos personales.
- **Conceptos teóricos:** Explicación de los fundamentos teóricos importantes relacionados con el proyecto, para tener los conocimientos necesarios para su comprensión.
- **Técnicas y herramientas:** Proporciona las técnicas y herramientas que han sido utilizadas en el desarrollo del proyecto.
- **Aspectos relevantes del desarrollo del proyecto:** Muestra los aspectos a destacar que han ocurrido en el desarrollo del proyecto.
- **Trabajos relacionados:** Investigación previa de trabajos relacionados con el proyecto, realizando una comparativa de funcionalidades de cada aplicación.
- **Conclusiones y líneas de trabajo futuras:** Conclusiones obtenidas al finalizar el proyecto y propuesta de posibles ideas de mejora para futuras actualizaciones.

1.2. Estructura de los anexos

- **Plan de proyecto software:** Planificación del proyecto, incluyendo por una parte la planificación temporal y por otra el estudio de viabilidad económica y legal.
- **Especificación de requisitos:** Especificación de los requisitos según los objetivos establecidos al comienzo del proyecto y los que se han ido añadiendo durante el proyecto.
- **Especificación de diseño:** Especificación de las distintas estructuras de diseño que se han empleado en el proyecto.
- **Documentación técnica de programación:** Documentación de los conceptos más técnicos, como la estructura del proyecto, manual del programador, instalación de librerías, ejecución de la aplicación y la realización de las pruebas.
- **Documentación de usuario:** Guía o manual de usuario para entender y utilizar la aplicación, sin problemas.
- **Anexo de sostenibilización curricular:** Reflexión sobre los aspectos de la sostenibilidad que se abordan en el trabajo.

2. Objetivos del proyecto

En este apartado se detallan los objetivos que se persiguen con la realización del proyecto, distinguiendo entre objetivos generales, relacionados con los requisitos del software a construir; objetivos técnicos, necesarios para llevar a cabo los anteriores y objetivos personales, relacionados con las metas que me gustaría alcanzar al finalizar el proyecto.

2.1. Objetivos generales

- **Desarrollo de la aplicación web con Angular:** Desarrollar la lógica de la aplicación web utilizando el framework Angular, aprovechando su estructura de componentes y servicios para construir una aplicación modular y escalable.
- **Obtener y gestionar datos de la NBA:** Implementar la integración de una API con información de la NBA (API-NBA) para recopilar datos en tiempo real sobre clasificación, equipos, jugadores, partidos y estadísticas relevantes.
- **Integración de usuarios:** Implementar un sistema completo de autenticación de usuarios que permita el registro, inicio de sesión y modificación de perfiles, proporcionando una experiencia de usuario fluida y segura.
- **Proporcionar información detallada sobre la NBA:** Mostrar información de la clasificación, playoffs, partidos y líderes de la sesión 23-24 de la NBA. Facilitar estadísticas exhaustivas y perfiles detallados de equipos y jugadores de la temporada actual.
- **Realizar análisis de rendimiento de jugadores:** Implementar técnicas de análisis de datos y la librería Chart.js, para evaluar me-

dian­te gráficas el desempeño y eficiencia de los jugadores en diferentes aspectos del juego, como puntos, asistencias, rebotes, entre otros.

- **Generar predicciones de resultados de partidos:** Aplicar el método Random Forest de Machine Learning que estudie las estadísticas y resultados de cada partido jugado por cada equipo y entrene un modelo para predecir los resultados de futuros partidos y proporcionar a los usuarios las probabilidades de victoria sobre los posibles ganadores.
- **Facilitar la interacción de los usuarios con la aplicación:** Desarrollar un diseño de interfaces visual e intuitivo en las funcionalidades a las que puedan acceder los usuarios para facilitar el uso de la aplicación.

2.2. Objetivos técnicos

- **Integración con Firebase:** Utilizar Firebase como plataforma de alojamiento para la base de datos, almacenamiento de archivos, autenticación de usuarios, servicio de hosting y funciones en la nube, para la aplicación web.
- **Gestión de base de datos NoSQL Firestore:** Adquirir conocimientos y habilidades en el uso de Firestore de Firebase como una base de datos NoSQL para almacenar y gestionar los datos de la aplicación web.
- **Implementación de Bootstrap:** Utilizar Bootstrap para mejorar la estética y la usabilidad de la aplicación web, haciéndola así una aplicación responsive para poder acceder a ella desde cualquier dispositivo sin perder información.
- **Implementación de Angular Material:** Integrar Angular Material en la aplicación web para aprovechar su biblioteca de componentes predefinidos y estilizados, garantizando así una interfaz de usuario moderna y consistente en toda la aplicación.
- **Implementación de Firebase Cloud Functions:** Inicializar Firebase Cloud Functions en mi proyecto, para utilizarlo como backend del proyecto y poder realizar la función de predicción de partidos mediante Python.
- **Seguridad de datos:** Aplicar medidas de seguridad para proteger la integridad y confidencialidad de los datos de los usuarios utilizando Firebase Authentication, y añadir las claves y credenciales de acceso en .gitignore para proteger la seguridad de la aplicación.
- **Internacionalización del proyecto:** Realizar la internacionalización del proyecto, para que los usuarios tengan la posibilidad de cambiar de idioma siempre que lo deseen.

- **Documentación del proyecto:** Elaborar una documentación completa que permita facilitar la comprensión y mantenimiento del proyecto.

2.3. Objetivos personales

- **Profundizar en el desarrollo frontend con Angular:** Aumentar mi experiencia y habilidades en el desarrollo de aplicaciones web frontend utilizando Angular.
- **Dominar el uso de Firebase como plataforma de desarrollo web:** Adquirir un conocimiento sólido sobre Firebase, tanto en su utilización para alojamiento de bases de datos, alojamiento de archivos y autenticación de usuarios, como en la implementación de funciones de hosting y cloud functions.
- **Aprender técnicas de Machine Learning:** Investigar y aplicar técnicas de Machine Learning en el contexto de predicciones de resultados de partidos en la NBA.
- **Adquirir experiencia en la integración de APIs externas:** Aprender a integrar y trabajar con APIs externas, para obtener y procesar datos relevantes de manera eficaz y segura dentro de la aplicación web.
- **Mejorar habilidades de diseño:** Incrementar y mejorar mis habilidades en diseño de interfaces de usuario, asegurando que la aplicación sea intuitiva, atractiva y fácil de usar para los usuarios finales.
- **Desarrollar habilidades de resolución de problemas:** Reforzar mis habilidades en la identificación y resolución de problemas técnicos y desafíos durante el desarrollo del proyecto.
- **Documentar adecuadamente el proceso de desarrollo:** Asegurar una documentación completa y detallada del proceso de desarrollo.

3. Conceptos teóricos

3.1. NBA

La NBA (National Basketball League) [41] es una liga de baloncesto, considerada la mejor del mundo. Esta competición ha vivido una amplia evolución a lo largo de 75 años, quedando en la actualidad una liga compuesta por 30 franquicias, situadas en Estados Unidos y Canadá.

El origen de la NBA sucedió en 1946, donde en una reunión impulsada por los propietarios de pabellones que buscaban una liga que se disputara en las noches que tenían cerrados sus recintos, se instauraron las bases de la competición y las ciudades de la misma.

Esta competición funciona de forma que primero se disputa la temporada regular, donde se clasifican los mejores equipos de cada conferencia y después avanzan a los playoffs, una serie de eliminatorias donde se decide el campeón de la liga.

Temporada Regular

Durante siete meses, las 30 franquicias se enfrentan en una larga temporada regular. En ella se determinan los equipos que compiten en los Playoffs y los que participan en el play-in, así como el orden de elección para gran parte del Draft del próximo año, decidido en última instancia por la Lotería del Draft.

Cada franquicia disputa un total de 82 partidos en la temporada regular de la NBA y dentro de estos 82 encuentros, los equipos se enfrentan 4 veces con los equipos de su misma división, 3 o 4 veces con el resto de equipos de su conferencia y 2 veces con los equipos de la otra conferencia.

Play-in

Antes de comenzar los Playoffs, se compiten los Play-in, donde cada conferencia organiza dos series a un partido que enfrentan al 7º con el 8º clasificado y al 9º con el 10º. El ganador de la primera serie consigue la séptima plaza de los Playoffs, mientras que el perdedor pasa a ser el equipo local ante el vencedor del partido entre el 9º y 10º; ese segundo choque entre perdedor del primer encuentro y ganador del segundo sale el último puesto a los Playoffs.

Playoffs

Una vez se han decidido las ocho mejores franquicias de cada conferencia, estas deben superar cuatro series al mejor de 7 partidos para proclamarse campeonas:

- Primera ronda.
- Semifinales de Conferencia.
- Finales de Conferencia.
- Finales de la NBA.

En las tres primeras series, los ocho clasificados de cada conferencia compiten por ser el mejor del Este y el Oeste. En las Finales, los ganadores de cada conferencia luchan por el título.

Los enfrentamientos se establecen por orden clasificatorio:

| | | | |
|----------|----------|----------|----------|
| 1º vs 8º | 4º vs 5º | 3º vs 6º | 2º vs 7º |
|----------|----------|----------|----------|

Conferencias y Divisiones

La NBA se divide en dos conferencias de 15 franquicias, Este y Oeste, que a su vez se fraccionan en seis diferentes divisiones de 5 equipos cada una.

Se pueden observar en las siguientes tablas de cada conferencia, las distintas divisiones con sus respectivos equipos:

| División Atlántico | División Central | División Sureste |
|--------------------|---------------------|--------------------|
| Boston Celtics | Chicago Bulls | Atlanta Hawks |
| Brooklyn Nets | Cleveland Cavaliers | Charlotte Hornets |
| New York Knicks | Detroit Pistons | Miami Heat |
| Philadelphia 76ers | Indiana Pacers | Orlando Magic |
| Toronto Raptors | Milwaukee Bucks | Washington Wizards |

Tabla 3.1: Equipos de la Conferencia Este.

| División Noroeste | División Pacífico | División Suroeste |
|------------------------|-----------------------|----------------------|
| Denver Nuggets | Golden State Warriors | Dallas Mavericks |
| Minnesota Timberwolves | Los Angeles Clippers | Houston Rockets |
| Oklahoma City Thunder | Los Angeles Lakers | Memphis Grizzlies |
| Portland Trail Blazers | Phoenix Suns | New Orleans Pelicans |
| Utah Jazz | Sacramento Kings | San Antonio Spurs |

Tabla 3.2: Equipos de la Conferencia Oeste.

3.2. API

Una API [31] (Application Programming Interfaces, Interfaz de Programación de Aplicaciones), es un conjunto de reglas, protocolos y herramientas que permiten la comunicación entre diferentes sistemas de software. Actúa como un intermediario que permite que las aplicaciones se comuniquen entre sí y compartan datos y funcionalidades de manera eficiente y segura.

Las APIs son esenciales para la integración de sistemas y la colaboración entre aplicaciones. Permiten que los desarrolladores accedan a recursos y servicios de una aplicación de software de forma controlada y estructurada, sin necesidad de conocer los detalles de implementación subyacentes.

Existen diferentes tipos de APIs:

- **API abiertas:** son interfaces de programación de aplicaciones de código abierto a las que se puede acceder con el protocolo HTTP.
- **API de socios:** conectan a socios comerciales estratégicos.
- **API internas:** permanecen ocultas de los usuarios externos.
- **API compuestas:** combinan múltiples API de datos o servicios.

Se han desarrollado protocolos API que facilitan el intercambio estandarizado de información, como son: **SOAP** (Protocolo simple de acceso a objetos), **XML-RPC** (llamada a procedimiento remoto XML), **JSON-RPC** o **REST** (Transferencia de estado representacional).

3.3. Protocolo HTTP

El protocolo HTTP [50] (Hypertext Transfer Protocol) es el protocolo fundamental de comunicación en la World Wide Web. Su principal objetivo es permitir la transferencia de información, como texto, gráficos, sonido, vídeo y otros archivos multimedia, entre un cliente (por lo general, un navegador web) y un servidor.

HTTP opera bajo un modelo de solicitud-respuesta, donde los clientes web envían solicitudes HTTP a los servidores web para recuperar recursos específicos. Cada solicitud incluye un método:

- **GET:** solicita la recuperación de un recurso específico.
- **POST:** envía datos al servidor para su procesamiento.
- **PUT:** actualiza un recurso existente en el servidor o crea uno si no existe.
- **DELETE:** elimina el recurso especificado en el servidor.
- **HEAD:** similar a GET, pero solicita solo los encabezados del recurso sin su cuerpo.

Los servidores web responden a estas solicitudes con respuestas HTTP que contienen el recurso solicitado.

3.4. Random Forest

Random Forest (Bosque Aleatorio) [30] es un algoritmo de Machine Learning desarrollado por Leo Breiman y Adele Cutler, que combina múltiples árboles de decisión para generar un resultado único. Este método es popular debido a su flexibilidad y facilidad de uso, y es aplicable tanto en problemas de clasificación como de regresión.

Un árbol de decisión toma decisiones basadas en una serie de preguntas secuenciales, donde cada respuesta guía hacia la siguiente pregunta o decisión

final. Sin embargo, los árboles de decisión individuales pueden ser propensos a sesgos y sobreajustes. Para mitigar estos problemas, el bosque aleatorio utiliza múltiples árboles no correlacionados.

El aprendizaje por conjunto combina varios clasificadores para mejorar la precisión de las predicciones. Un método común es la agregación bootstrap, donde se toman múltiples muestras con reemplazo del conjunto de datos de entrenamiento y se entrenan modelos de forma independiente. La combinación de las predicciones de estos modelos, mediante promedio (regresión) o voto mayoritario (clasificación), reduce la varianza y mejora la precisión.

El algoritmo random forest extiende el método de agregación bootstrap añadiendo aleatoriedad en la selección de características para cada árbol, lo que reduce la correlación entre los árboles y mejora la precisión. Utiliza tres hiperparámetros principales: el tamaño de nodo, el número de árboles y el número de características muestreadas. Para la validación, se utiliza la muestra OOB (Out-Of-Bag), que no se incluye en el entrenamiento de cada árbol.

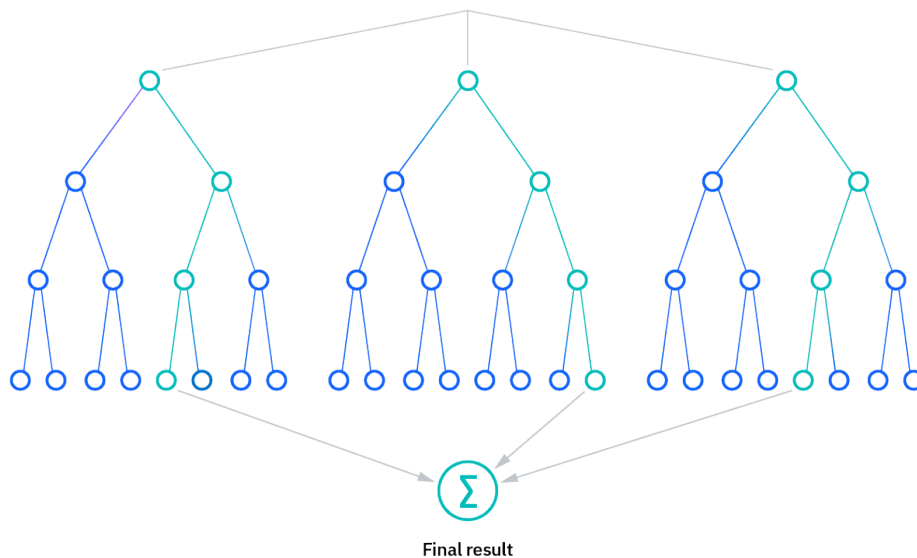


Figura 3.1: Algoritmo Random Forest

Ventajas:

- **Reducción del riesgo de sobreajuste:** La combinación de múltiples árboles no correlacionados promedia sus resultados, disminuyendo la varianza y el error de predicción.
- **Flexibilidad:** Puede manejar tareas de regresión y clasificación con alta precisión y es útil para estimar valores faltantes en los datos.
- **Evaluación de la importancia de características:** Facilita la identificación de la importancia de las variables en el modelo mediante métricas como la importancia de Gini y la disminución media de la exactitud (MDA).

Desventajas:

- **Procesamiento lento:** Manejar grandes conjuntos de datos puede ser lento debido al procesamiento individual de cada árbol.
- **Requerimientos de recursos:** Necesita más recursos para almacenar y procesar grandes volúmenes de datos.
- **Complejidad:** Las predicciones de un bosque aleatorio son más difíciles de interpretar en comparación con un único árbol de decisión.

3.5. NoSQL

NoSQL [32] (Not Only SQL), es un enfoque de diseño de base de datos que permite almacenar y consultar datos sin seguir el modelo relacional tradicional utilizado en las bases de datos SQL.

Proporciona otras opciones para organizar datos de muchas maneras utilizando cualquiera de estos modelos de datos primarios:

- Almacén de pares clave-valor.
- Almacén de documentos.
- Almacén distribuido en columnas
- Almacén de grafos
- Almacén en memoria

Cada tipo de base de datos NoSQL presenta cualidades para casos de uso específicos. Sin embargo, todas comparten las siguientes ventajas: Rentabilidad, Flexibilidad, Réplica y Velocidad.

4. Técnicas y herramientas

4.1. Angular

Angular [12] es un potente framework de JavaScript ideal para construir aplicaciones frontend modernas de nivel medio o alto. Especializado en aplicaciones de una sola página (SPA) y aplicaciones web progresivas (PWA), ofrece una sólida base para desarrollar aplicaciones escalables y optimizadas, fomentando las mejores prácticas y un estilo de codificación modular y coherente.

El desarrollo en Angular se realiza utilizando TypeScript, un superset de JavaScript que ofrece ventajas adicionales como tipado estático y decoradores, aunque es posible desarrollar con JavaScript siguiendo las guías y recomendaciones de la comunidad.

He usado este framework en este proyecto por todas las características que nos ofrece y porque considero que es uno de los mejores frameworks para el desarrollo de aplicaciones web. Para generar mi proyecto Angular se ha necesitado Angular CLI.

Angular CLI

Angular CLI [13] es una herramienta fundamental dentro del ecosistema de Angular. Es una interfaz de línea de comandos (Command Line Interface, CLI) proporcionada por el equipo de Angular para facilitar el desarrollo de aplicaciones con este framework.

Esta herramienta facilita el proceso de inicio de cualquier aplicación Angular al generar rápidamente la estructura básica de archivos y carpetas necesarios, junto con numerosas herramientas configuradas. Durante el

desarrollo, Angular CLI ayuda en la creación de componentes y ofrece asistencia en diversas tareas. También es útil en etapas de producción y pruebas, facilitando la preparación de archivos para el servidor.

Para instalar Angular CLI, se requiere tener NodeJS en el sistema operativo. Angular CLI se instala a través de npm, el gestor de paquetes de NodeJS.

4.2. Firebase

Firebase [40] es una plataforma en la nube (de Google), que está diseñada para facilitar el proceso de creación, desarrollo y gestión de aplicaciones web y móviles.

Una de las principales ventajas de Firebase es su capacidad para trabajar con diferentes plataformas, incluyendo iOS, Android y web, lo que proporciona a los desarrolladores una mayor flexibilidad y eficiencia en el desarrollo de aplicaciones multiplataforma. Esto significa que los desarrolladores pueden utilizar las mismas herramientas y servicios para construir y gestionar aplicaciones para diferentes sistemas operativos, lo que reduce significativamente el tiempo y los recursos necesarios para el desarrollo.

Firebase ofrece una amplia gama de herramientas y servicios, que simplifican y aceleran el proceso de desarrollo de aplicaciones, por esto lo he usado en mi aplicación. Entre estas herramientas, se han utilizado las siguientes:

Authentication

Firebase ofrece un sistema de autenticación de usuarios seguro y fácil de usar que permite a los desarrolladores gestionar la identidad de los usuarios de la aplicación. Los desarrolladores pueden permitir que los usuarios se autenticuen utilizando diferentes métodos, como correo electrónico y contraseña, redes sociales (como Google, Facebook y Twitter) o números de teléfono.

Firestore

Cloud Firestore [49] es una base de datos NoSQL en la nube que destaca por su flexibilidad y escalabilidad. Permite almacenar y sincronizar datos de forma eficiente para aplicaciones en clientes, servidores y dispositivos móviles.

Hosting

Firebase ofrece un servicio de hosting que permite a los desarrolladores alojar sus aplicaciones web de forma rápida y sencilla. Firebase proporciona certificados de seguridad SSL y HTTP2 de forma automática y gratuita para cada dominio, lo que garantiza la seguridad de las aplicaciones alojadas en su plataforma.

Storage

Cloud Storage [21] para Firebase es un servicio de almacenamiento de objetos que utiliza la infraestructura rápida y segura de Google Cloud, ideal para desarrolladores de apps que necesitan almacenar y entregar contenido generado por usuarios, como fotos y videos. En mi caso, almaceno mis archivos csv y mi modelo random forest.

Cloud Functions

Google Cloud Functions [6] es la solución de Google para el procesamiento sin servidores, ideal para crear aplicaciones controladas por eventos. Desarrollado conjuntamente por los equipos de Google Cloud Platform (GCP) y Firebase, permite a los desarrolladores conectar lógicas de diferentes servicios de GCP mediante la detección y respuesta a eventos.

Para desarrolladores de Firebase, Cloud Functions para Firebase ofrece una forma de extender e integrar el comportamiento del producto con código del servidor, proporcionando ejecución rápida y confiable en un entorno completamente administrado, sin necesidad de gestionar servidores o infraestructura.

Esta funcionalidad de firebase la he implementado en mi proyecto con un uso de backend, en el que realizo en lenguaje Python, las predicciones de partidos.

4.3. TypeScript

TypeScript [16] es un lenguaje que compila a JavaScript, ampliando sus capacidades y herramientas, lo que lo convierte en un "superset" de JavaScript. Diseñado para el desarrollo de aplicaciones robustas, destaca por detectar tempranamente problemas comunes durante el desarrollo de sitios web.

Una de sus características más destacadas es la incorporación de tipos de datos estáticos, lo que permite una tipificación opcional pero recomendada, a diferencia de JavaScript, que carece de tipado estático. Esto convierte a TypeScript en un lenguaje fuertemente tipado, similar a lenguajes empresariales como Java o C#.

Además de la tipificación estática, TypeScript ofrece otras utilidades como genéricos y decoradores, presentes en lenguajes de programación avanzados. En conjunto, estas características convierten a TypeScript en una herramienta sólida que mejora la experiencia de los desarrolladores en su trabajo diario, dotando a JavaScript de características que lo acercan a lenguajes más avanzados.

4.4. HTML y CSS

HTML y CSS son complementarios, ya que uno define el contenido y el otro la presentación. En la actualidad, el uso conjunto de ambos es fundamental para crear páginas web visualmente atractivas y funcionales.

HTML

HTML [15] (Hypertext Markup Language), es el lenguaje fundamental utilizado para crear páginas web. Se basa en etiquetas o marcas que definen el contenido de una página web, incluyendo elementos como encabezados, párrafos, enlaces, etc.

CSS

CSS [14] (Cascading Style Sheets, Hojas de estilo en cascada), es un lenguaje esencial para el diseño web, utilizado para definir la apariencia y el estilo de los elementos en una página web. Junto con HTML, que define el contenido, CSS permite controlar la presentación visual, incluyendo aspectos como la disposición, forma, espaciado y color de los elementos.

4.5. Visual Studio Code

Visual Studio Code [23] (VS Code) es un popular editor de código desarrollado por Microsoft, que se destaca por su versatilidad y potencia. Es de código abierto y está disponible para Windows, GNU/Linux y macOS. Este editor cuenta con una integración sólida con Git, soporte para depuración de código y una amplia variedad de extensiones que permiten escribir y ejecutar código en diversos lenguajes de programación.

Considerado como el entorno de desarrollo más utilizado según una encuesta realizada por Stack Overflow en mayo de 2021, con un impresionante 71.06 % de adopción, VS Code ofrece una serie de características que lo hacen tan popular entre los desarrolladores.

Entre las características destacadas de Visual Studio Code se encuentran:

- **Multiplataforma:** Disponible en Windows, GNU/Linux y macOS, lo que lo hace accesible para una amplia gama de usuarios.
- **IntelliSense:** Proporciona sugerencias de código inteligentes y completado automático, lo que agiliza la escritura del código.
- **Depuración:** Permite detectar y corregir errores en el código de manera eficiente, lo que facilita el proceso de desarrollo.
- **Control de versiones:** Compatible con Git, lo que facilita la gestión de cambios en el código y la colaboración en proyectos.
- **Extensiones:** Ofrece una gran variedad de extensiones que permiten personalizar el editor y agregar funcionalidades adicionales, como soporte para diferentes lenguajes de programación, temas personalizados y conexiones con otros servicios.

He utilizado Visual Studio Code en mi proyecto debido a su eficiencia, amplia gama de extensiones y excelente integración con herramientas como Git, lo que ha mejorado mi productividad y facilitado el desarrollo.

4.6. Bootstrap

Bootstrap [38] es una biblioteca de herramientas de código abierto diseñada para facilitar el desarrollo de sitios y aplicaciones web. Utilizando HTML y CSS como base, Bootstrap proporciona una amplia gama de elementos de diseño, como formularios, botones y menús, que se adaptan a diferentes dispositivos y tamaños de pantalla.

Aunque se utiliza principalmente para el desarrollo web con HTML, CSS y JavaScript, Bootstrap a menudo se conoce como un "marco CSS", ya que su enfoque simplifica el diseño y la implementación. Esto hace que sea común encontrar código escrito en CSS al utilizar Bootstrap, además de una amplia biblioteca en este lenguaje.

Gracias a su capacidad para simplificar y acelerar el desarrollo de sitios web responsivos, Bootstrap se ha convertido en una herramienta popular entre desarrolladores front-end y profesionales del diseño web. Por esto mismo decidí incorporar Bootstrap en mi proyecto.

4.7. Angular Material

Angular Material [39] es un módulo de Angular que simplifica el desarrollo de aplicaciones web al ofrecer una amplia gama de componentes de interfaz de usuario predefinidos. Este módulo está diseñado para permitir la creación rápida y sencilla de aplicaciones visualmente atractivas. Se basa en Material Design, un estándar de diseño desarrollado por Google en 2014, que se centra en la creación de interfaces consistentes y atractivas.

He integrado Angular Material en mi proyecto de Angular porque proporciona varias ventajas significativas como:

- Ofrece diseños predefinidos que pueden implementarse fácilmente, lo que ahorra tiempo en el proceso de diseño.
- Al estar integrado de forma nativa con Angular, su uso es intuitivo y se adapta perfectamente a las funcionalidades del framework.
- Angular Material incluye formularios con validación incorporada, lo que simplifica el desarrollo y garantiza la integridad de los datos.

4.8. Chart.js

Chart.js [45] es una biblioteca de código abierto en JavaScript que destaca por su facilidad de uso y es perfecta para representar datos en forma de gráficos, como barras, circulares o líneas. Es ideal para aquellos que tienen conocimientos básicos de JavaScript y HTML.

Chart.js ofrece las siguientes ventajas:

- Es de código abierto.
- No necesita ninguna dependencia externa ni librería adicional, solo JavaScript y HTML.
- Se puede integrar fácilmente con cualquier framework, como Angular, Vue o React.
- Cuenta con una gran comunidad de usuarios, soporte y una documentación completa.
- Su uso no requiere de conocimientos avanzados.

Decidí implementar Chart.js por todas sus ventajas y lo he utilizado para mostrar gráficamente los análisis de las estadísticas de los jugadores y las probabilidades que un equipo tiene de ganar contra otro equipo de la NBA.

4.9. GitHub

GitHub [2] es una plataforma en la nube que alberga un sistema de control de versiones (VCS) llamado Git. Este sistema permite a los desarrolladores colaborar y realizar cambios en proyectos compartidos, mientras registran detalladamente su progreso.

Control de versiones

El control de versiones es una herramienta esencial que ayuda a rastrear y gestionar los cambios realizados en un archivo o conjunto de archivos. Es utilizado principalmente por ingenieros de software para controlar el código fuente, permitiéndoles analizar y revertir cambios sin consecuencias graves.

Git

Git es un sistema de control de versiones distribuido y de código abierto, ampliamente utilizado en todo el mundo. Permite a los miembros del equipo gestionar el código fuente y su historial de cambios de forma independiente, utilizando herramientas de línea de comandos. Git ofrece la funcionalidad de ramas de características, que permiten a los desarrolladores trabajar en su propio repositorio local aislado del código antes de fusionar los cambios con la rama principal del proyecto.

Antes de comenzar el proyecto, decidí utilizar GitHub como plataforma para registrar mis avances en un repositorio remoto.

4.10. Trello

Trello [51] es una herramienta de gestión de proyectos diseñada para facilitar la colaboración y el seguimiento de tareas de manera visual y colectiva. Esta plataforma está diseñada para simplificar la organización de información y tareas para equipos de trabajo.

Su función principal es proporcionar una interfaz intuitiva que permite a los usuarios acceder y organizar la información relacionada con proyectos, planes de trabajo y metas. Algunas de sus características clave incluyen:

- Organización visual de la información para una mejor comprensión.
- Gestión de tareas, tanto pequeñas como grandes, de manera eficiente.
- Herramientas creativas, como la lluvia de ideas, para fomentar la colaboración.
- Ayuda en la definición y seguimiento de objetivos y planes de trabajo.
- Seguimiento del progreso en la realización de un plan.
- Acceso compartido a los planes de trabajo para diferentes usuarios.

He utilizado la herramienta Trello por todas sus características y su diseño visual e intuitivo, que me facilita la gestión del proyecto.

4.11. Zotero

Zotero [11] es un gestor de referencias bibliográficas multiplataforma, libre, abierto y gratuito desarrollado por el Corporation for Digital Scholarship y el Roy Rosenzweig Center for History and New Media. Su propósito es simplificar la recopilación y administración de recursos para investigaciones.

Esta herramienta me ha ayudado para gestionar todas las referencias bibliográficas que he consultado durante la documentación del proyecto y la investigación de ciertos conceptos y dudas al desarrollar la aplicación web.

4.12. LaTeX

LaTeX [52] es una poderosa herramienta utilizada para crear documentos profesionales con una presentación impecable. A diferencia de los procesadores de texto tradicionales como Word o Libre Office, LaTeX funciona de manera única: utiliza comandos para formatear contenidos complejos, especialmente aquellos relacionados con las matemáticas, como fracciones, subíndices, matrices y derivadas.

Este sistema, pronunciado como "lah-tech." o "lay-tech", permite una amplia gama de aplicaciones. Los documentos en LaTeX son archivos de texto sin formato que incluyen comandos LaTeX para expresar la estructura y el formato del documento.

Overleaf

Overleaf [10] es una herramienta online de publicación y redacción colaborativa en línea que hace que todo el proceso de redacción, edición y publicación de documentos científicos sea mucho más rápido y sencillo. Overleaf ofrece la comodidad de un editor LaTeX fácil de usar con colaboración en tiempo real y la salida completamente compilada que se produce automáticamente en segundo plano mientras escribe.

Decidí utilizar LaTeX como editor de texto por la profesionalidad que aporta a la documentación del proyecto y Overleaf como herramienta de edición por su comodidad y facilidad de uso.

4.13. RapidAPI

RapidAPI [47] es una plataforma que facilita la conexión entre desarrolladores y proveedores de servicios web (APIs). Permite a los desarrolladores descubrir, utilizar y gestionar una amplia gama de APIs de forma centralizada.

Los desarrolladores pueden encontrar y acceder a APIs de diferentes proveedores, lo que les permite integrar diversas funcionalidades y datos en sus aplicaciones de manera rápida y sencilla. RapidAPI ofrece una variedad de herramientas para simplificar el proceso de desarrollo, incluyendo documentación detallada, pruebas en línea, generación de código y análisis de uso. Además, proporciona características para administrar suscripciones, facturación y seguridad de las APIs utilizadas.

Tras investigar durante bastante tiempo plataformas para acceder a APIs, llegué a la conclusión de que esta herramienta es la mejor, por su amplia variedad de APIs y por su forma rápida y sencilla de utilizarlas.

4.14. Microsoft Copilot

Microsoft Copilot [5] es una innovadora herramienta de chat desarrollada por Microsoft que aprovecha la inteligencia artificial para sostener conversaciones. Este asistente virtual está siempre conectado a internet, lo que le permite proporcionar información actualizada y referencias web relevantes en respuesta a consultas.

Una de las características más destacadas de Microsoft Copilot es su integración con DALL-E3, una IA generativa capaz de crear imágenes a partir de texto. Además, al igual que Gemini de Google, Copilot se integra en varias herramientas de Microsoft 365, como Outlook, Excel, Word y PowerPoint, lo que permite aprovechar la IA en diversas tareas y aplicaciones de productividad.

Gracias a esta herramienta pude hacer realidad el logotipo que tenía en mente para la aplicación StatsGlowMind, por su capacidad de creación de imágenes a partir de un texto.

4.15. Canva

Canva [20] es una plataforma de diseño gráfico y composición de imágenes que se lanzó en 2012. Ofrece herramientas en línea para crear diseños personalizados, tanto para uso personal como profesional. Utiliza un modelo freemium, lo que significa que puedes acceder de forma gratuita a sus servicios básicos, pero también ofrece opciones avanzadas mediante suscripción.

Puedes crear una variedad de diseños, incluyendo logos, posters, tarjetas de visita, flyers, portadas, invitaciones, folletos, calendarios, encabezados para correos electrónicos y publicaciones para redes sociales, entre otros.

En esta plataforma he realizado el diseño del nombre de la aplicación StatsGlowMind.

4.16. Flaticon

Flaticon [7] es una destacada fuente de iconos en Internet, conocida por su calidad y variedad. Su modelo de negocio freemium ofrece una licencia gratuita que permite usar una gran cantidad de iconos en proyectos, siempre y cuando se atribuya al autor en el proyecto final. Esta opción resulta atractiva para los creadores de sitios web, ya que proporciona acceso a una amplia gama de iconos sin coste alguno.

Para los iconos de la aplicación he utilizado esta herramienta que cuenta con una gran variedad de iconos.

4.17. Freepik

Freepik [19] es un buscador que ofrece más de 10 millones de recursos gráficos de alta calidad, incluyendo fotos, vectores, ilustraciones y archivos PSD. Esta plataforma te permite encontrar fácilmente contenido gráfico para tus proyectos creativos, tanto personales como profesionales. Además, Freepik facilita la edición de estas creaciones, ya que muchos elementos están disponibles en formatos compatibles con Adobe Illustrator, lo que permite a los diseñadores personalizarlos según la imagen de marca de su empresa.

Para las imágenes que componen el main de la aplicación utilicé este buscador.

4.18. Draw.io

Draw.io [36] es un software gratuito para la creación y edición de diagramas, disponible tanto offline como online a través de un navegador web. Ofrece integración con diversas plataformas y programas.

He usado esta herramienta para generar todos los diagramas de la documentación de los anexos.

4.19. Figma

Figma [33] es una herramienta de diseño de interfaces dirigida a diseñadores web, UX y UI, ideal para crear sitios web y aplicaciones. Destaca por ser una alternativa completa y accesible a programas como Sketch, con funciones avanzadas y multiplataforma.

Esta herramienta me ayudó a generar el prototipo de la interfaz de la aplicación.

4.20. Postman

Postman [34] es una herramienta de colaboración y desarrollo diseñada para que los desarrolladores interactúen y prueben servicios web y aplicaciones. Ofrece una interfaz gráfica intuitiva que facilita el envío de solicitudes a servidores web y la recepción de las respuestas correspondientes.

Con esta herramienta pude realizar las pruebas de petición POST a mi función *predict* de Cloud Functions y comprobar su correcto funcionamiento.

5. Aspectos relevantes del desarrollo del proyecto

Este apartado recoge los aspectos más interesantes del desarrollo del proyecto, incluyendo desde el inicio del proyecto y formación, hasta el desarrollo de software, diseño de la aplicación y resolución de problemas.

Inicio de proyecto

El inicio de mi proyecto se centró en una combinación de mis intereses personales y profesionales. Elegí desarrollar una aplicación web sobre estadísticas de la NBA porque siento una gran pasión por el baloncesto y, en particular, por la NBA. Este interés personal se complementó perfectamente con mi deseo de profundizar en el desarrollo web, utilizando Angular, un framework que considero líder en la creación de aplicaciones web dinámicas y robustas.

En el proceso de investigación sobre cómo desarrollar el proyecto, descubrí Firebase, una plataforma de desarrollo de aplicaciones web que me proporcionó una serie de servicios útiles como base de datos, almacenamiento en la nube, autenticación de usuarios, funciones en la nube y despliegue de la aplicación. Este descubrimiento fue crucial, ya que Firebase ofrecía todo lo necesario para gestionar los usuarios y los datos estadísticos de la NBA de manera eficiente y segura.

Además, este proyecto me ofrecía la oportunidad de trabajar con APIs externas, análisis de datos y Machine Learning, áreas que me interesan profundamente. También me ha permitido mejorar tanto mis habilidades de diseño web como de resolución de problemas.

Formación

Como la mayoría de las herramientas empleadas para desarrollar el proyecto no se han visto en la carrera, he estado investigando bastante acerca de estas tecnologías por lo que me dediqué a realizar varios cursos formativos y ver video tutoriales, que me han servido para comprender mejor estas herramientas.

Angular

Angular es un framework que conocía, pero que no he llegado a utilizar en un proyecto personal, por lo que decidí utilizarlo en este proyecto y profundizar en su aprendizaje. Para ello, realicé varios cursos que me permitieron adquirir una comprensión más sólida y avanzada de esta herramienta. Entre los cursos más destacados se encuentran:

- **Tour of Heroes application and tutorial [1]:** Este tutorial oficial de Angular proporciona una introducción práctica y detallada al framework. A través de este curso, aprendí los conceptos fundamentales de Angular, incluyendo la creación de componentes, servicios y el enrutamiento básico.
- **Fundamentos de Angular (OpenWebinars) [42]:** Con este curso comprendí mejor los conceptos fundamentales de Angular y me profundicé en los fundamentos clave de Angular.
- **Consumir APIs externas en Angular (OpenWebinars):** Este curso me enseñó a integrar APIs externas dentro de una aplicación Angular. Aprendí a realizar peticiones HTTP utilizando el módulo HttpClient de Angular, manejar respuestas y errores, y estructurar servicios.
- **Diseño web con Material Design para Angular (OpenWebinars):** En este curso, adquirí conocimientos sobre cómo utilizar Angular Material para crear interfaces de usuario modernas y atractivas. Aprendí a implementar componentes de Angular Material, como botones, formularios, tablas y diálogos, ...
- **Personalización de temas en Angular Material (OpenWebinars):** Este curso complementó mis conocimientos sobre Angular Material, enseñándome a personalizar y tematizar los componentes. Aprendí a crear y aplicar temas personalizados, gestionar paletas de colores y utilizar las herramientas de Angular Material.

Firestore

Firestore fue otra tecnología clave en el desarrollo de este proyecto. Para familiarizarme con esta plataforma, también realicé varios cursos y tutoriales:

- **Crear y Configurar un Proyecto en Angular y Firestore [17]:** Con este video tutorial aprendí como integrar Firestore y Angular, y como crear y configurar una aplicación web que pueda utilizar todas las funcionalidades de Firestore.
- **Crear un Login con Firestore en Angular [8]:** Este video tutorial explica como implementar la funcionalidad de autenticidad de Firestore, en un proyecto Angular. Para realizar el registro, login y logout de usuarios en la aplicación.
- **Deploy Angular a Firestore Hosting [9]:** En este video tutorial comprendí cómo hacer el despliegue de la aplicación de Angular en Firestore paso a paso, utilizando Firestore Hosting.
- **Curso de Firestore y Angular (OpenWebinars):** Este curso mejoró mis conocimientos y me enseñó a configurar y utilizar los servicios de Firestore, como la base de datos, almacenamiento en la nube y cloud functions.

Desarrollo

En esta sección se describen los aspectos más relevantes del desarrollo del proyecto, que han sido clave para desarrollar la aplicación web.

Integración de APIs externas

Una de las claves del desarrollo fue la integración de APIs externas para obtener datos actualizados sobre la NBA. Esta integración permitió a la aplicación proporcionar información en tiempo real sobre clasificaciones, partidos, y estadísticas de jugadores y equipos.

Para seleccionar la API a utilizar, se eligió la API-NBA [46], una API gratuita que encontré en la plataforma de RapidAPI, debido a su extensa base de datos y capacidad para proporcionar datos en tiempo real. También se uso la API oficial de la NBA para recoger información que no tenía la anterior API, los datos de los playoffs y las estadísticas de los líderes de la temporada de la NBA. A continuación, se muestran las URLs que se han utilizado para realizar peticiones a las APIs:

```
// Rutas de la API
private apiUrls = {
  standings: 'https://api-nba-v1.p.rapidapi.com/standings',
  leagueLeaders: 'https://stats.nba.com/stats/leagueleaders',
  games: 'https://api-nba-v1.p.rapidapi.com/games',
  playoffs: 'https://stats.nba.com/stats/playoffbracket',
  teams: 'https://api-nba-v1.p.rapidapi.com/teams',
  seasonGames: 'https://api-nba-v1.p.rapidapi.com/games',
  gameStats: 'https://api-nba-v1.p.rapidapi.com/games/statistics',
  teamPlayers: 'https://api-nba-v1.p.rapidapi.com/players',
  playerStats: 'https://api-nba-v1.p.rapidapi.com/players/statistics'
};

// Claves de la API
private headers = new HttpHeaders({
  'X-RapidAPI-Key': API_KEY,
  'X-RapidAPI-Host': 'api-nba-v1.p.rapidapi.com',
});
```

Figura 5.1: URL de las APIs.

Después de seleccionar las APIs, se desarrolló el servicio StatsService en Angular para manejar las peticiones HTTP a la API, asegurando que los datos se reciban y procesen de manera eficiente y segura.

Finalmente, los datos recibidos de la API, se almacenaron en Firestore, la base de datos NoSQL de Firebase, lo que facilitó su gestión y acceso rápido desde la aplicación.

Implementación de Chart.js

Otro aspecto relevante en el desarrollo del proyecto fue la implementación de Chart.js para la visualización de datos y poder analizarlos mediante gráficos. Chart.js es una librería de JavaScript que permite crear gráficos interactivos y dinámicos, facilitando la representación visual de estadísticas y datos complejos de una manera atractiva para los usuarios.

Se utilizaron diversos tipos de gráficos para representar diferentes conjuntos de datos:

- **Gráficos de Líneas:** Para representar la evolución del rendimiento de un jugador a lo largo de la temporada.

- **Gráficos de Pastel:** Para visualizar distribuciones porcentuales, como los tiros de campo o probabilidad de victoria entre dos equipos.
- **Gráficos de Barras:** Para mostrar la comparativa de la diferencia de puntos cuando un jugador está en la cancha.

Desarrollo del Modelo Random Forest

Uno de los aspectos más importantes del proyecto fue la implementación de un modelo de Machine Learning, específicamente el algoritmo Random Forest, para predecir los resultados de los partidos de la NBA. Este modelo se diseñó para analizar una amplia variedad de datos estadísticos y generar predicciones sobre los posibles ganadores de futuros enfrentamientos.

El desarrollo del modelo incluyó varios pasos críticos:

- **Recopilación de datos:** Se utilizó la API-NBA para recoger datos históricos de la temporada y en tiempo real sobre los partidos de la NBA. Estos datos que están almacenados en Firebase Firestore, se recopilaron en el proyecto Angular y se exportaron en un archivo csv utilizando la librería Papa Parse [43] para su posterior uso.
- **Entrenamiento del Modelo:** Se entrenó el modelo Random Forest utilizando el anterior archivo csv, para ello se creó un Jupyter Notebook en el cual descargó el archivo csv de Firebase Storage, cargó el csv en un DataFrame y finalmente entrenó el modelo con los datos recopilados y lo almacenó en Firebase Storage.

```
# Entrenar el modelo con las estadísticas de los partidos históricos
X = data.drop(columns=['result', 'game_id', 'home_id', 'visitor_id'])
y = data['result']
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X, y)

# Guardar el modelo en Firebase Storage
model_blob = bucket.blob('random_forest_model.pkl')
model_data = pickle.dumps(model)
model_blob.upload_from_string(model_data)

print("Modelo entrenado y guardado en Firebase Storage")
```

Figura 5.2: Entrenamiento del Modelo.

- **Pruebas:** Se realizaron pruebas exhaustivas para evaluar la precisión del modelo y se hicieron ajustes adicionales para optimizar su capacidad.

- **Implementación en la Aplicación:** El modelo se integró en la aplicación web utilizando Firebase Cloud Functions, lo que permitió crear la función *predict* para realizar predicciones en tiempo real y proporcionar recomendaciones a los usuarios sobre los equipos con mayor probabilidad de ganar. Este paso fue el más complicado ya que hubo problemas de dependencias de Python.
- **Petición post a la función *predict*:** Una vez desplegada la función *predict* en Cloud Functions se realizaron pruebas de peticiones post utilizando la herramienta Postman, una vez comprobado que todo funciona correctamente se generó el servicio PredictionService en Angular para realizar la petición post a la función.

```
export class PredictionService {  
  constructor(private http: HttpClient) { }  
  
  predict(homeTeamId: number, visitorTeamId: number): Observable<any> {  
    const body = {  
      home_team_id: homeTeamId,  
      visitor_team_id: visitorTeamId  
    };  
    return this.http.post<any>('https://us-central1-statsglowmindtfg.cloudfunctions.net/predict', body);  
  }  
}
```

Figura 5.3: Servicio PredictionService.

- **Mostrar predicción:** Para mostrar las probabilidades de victoria de un partido, se ha implementado Chart.js en el componente Predict-Component de Angular; en él se seleccionan los equipos mediante un select de Angular Material, se pulsa en el botón *Predecir Partido* y tras varios segundos de procesamiento, muestra los resultados de la predicción en un gráfico de pastel.

Diseño

El diseño del proyecto ha sido uno de los aspectos fundamentales para asegurar una interfaz de usuario atractiva y funcional. Para lograr esto, se han utilizado varias herramientas y tecnologías de diseño web como Bootstrap y Angular Material.

Bootstrap lo he utilizado para gestionar los estilos y asegurar la responsividad de la aplicación. Y he implementado Angular Material para insertar componentes preconstruidos, como botones, formularios, tablas, diálogos, ...; así como personalizar un tema para ajustar la apariencia de la aplicación

a la paleta de colores usada, la cual se compone de los siguientes colores: Azul marino (#002649), naranja (#FE5B3B) y blanco (#FFFFFF).

Aparte de estas dos bibliotecas, también se ha utilizado la herramienta Flaticon para mejorar la interfaz de la aplicación con iconos y la herramienta Freepik para mostrar imágenes de fondo en los componentes cards de la página principal de la aplicación. A continuación, se presentan los autores de los siguientes iconos e imágenes utilizados:

- **Icono Login y Analysis:** Autor berkahicon [3].
- **Icono Games y Playoffs:** Autor Kreev Studio [48].
- **Icono Outlook, Standings y GitHub:** Autor Pixel perfect [44].
- **Icono Leaders y Google:** Autor Freepik [25].
- **Icono Teams:** Autor kmg design [37].
- **Icono Predict:** Autor bsd [4].
- **Icono Logout:** Autor Icon Hubs [29].
- **Icono NBA:** Autor Fliqker [22].
- **Icono LinkedIn:** Autor Google [28].
- **Imagen Cards Main:** Diseñado por Freepik [26].
- **Imagen Cards NBA:** Diseñado por Freepik [27].

Resolución de problemas

Durante el desarrollo del proyecto, se presentaron varios desafíos técnicos que fueron solucionados cuidadosamente. A continuación, se detallan algunos de los problemas más significativos.

Problema con las peticiones a la API

Inicialmente, el proyecto enfrentó problemas debido a la gran cantidad de peticiones realizadas a la API de la NBA, ya que al ser una API gratuita solo permitía 10 peticiones por minuto y 100 al día. Para evitar esto, había que seleccionar un servicio de pago que te ofrecía muchas más peticiones diarias.

Para resolver este problema evitando costos, se tuvo que reducir las peticiones minimizando las peticiones innecesarias y agrupando múltiples peticiones en una sola cuando era posible. También se programaron actualizaciones periódicas para refrescar los datos en intervalos definidos en lugar de realizar peticiones continuas, lo que ayudó a gestionar mejor los límites de la API.

Problema con Firebase Cloud Functions

Integrar funciones escritas en Python dentro de un proyecto Angular desarrollado con TypeScript presentó varios problemas de dependencias y compatibilidad. Las soluciones implementadas fueron:

- **Configuración de Entorno:** Se configuró un entorno adecuado para ejecutar funciones Python dentro del ecosistema de Firebase, utilizando herramientas como virtualenv para manejar dependencias.
- **Despliegue en Firebase Cloud Functions:** Las funciones Python se desplegaron independientemente de la aplicación Angular, utilizando Firebase Cloud Functions para ejecutar código Python en la nube.
- **Configuración de Firebase Cloud Functions:** Al inicializar Firebase Cloud Functions, te permite elegir entre varios lenguajes de programación para implementar en tus funciones (JavaScript, TypeScript o Python), por lo que seleccioné la opción de Python para evitar más errores de compatibilidad.

Problema con Internacionalización i18n

Al intentar implementar la internacionalización del proyecto usando i18n y locales, se descubrió que cambiar dinámicamente el idioma no era posible con esta configuración. Esto es un problema para el usuario porque limita la flexibilidad y la experiencia del mismo.

Se encontró una solución, que fue el uso de la biblioteca ngx-translate, que ofrece una solución más flexible para la internacionalización en aplicaciones Angular ya que permite cambiar el idioma de manera dinámica, lo que mejoró significativamente la usabilidad de la aplicación.

6. Trabajos relacionados

En este apartado, se realiza un análisis de distintas aplicaciones web relacionadas con la temática del proyecto, es decir, Asociación Nacional de Baloncesto (NBA), dichas aplicaciones están compuestas de funcionalidades y características similares a las de StatGlowMind.

El objetivo de este análisis es recoger ideas, funcionalidades y características adicionales de otros proyectos similares, para ampliar el desarrollo del proyecto y mejorarlo. Por esto mismo, realizar este estudio fue uno de los primeros pasos que realicé al comenzar el proyecto.

6.1. Hoops Stats

Hoops Stats [35] es una aplicación web que principalmente muestra toda clase de estadísticas de cada uno de los equipos y jugadores de la NBA, a nivel histórico. Además de esto, también tiene las distintas funcionalidades:

- Muestra todos los resultados de los partidos y los partidos restantes de toda la temporada de la NBA.
- Informa de datos curiosos como las rachas y récords de cada equipo y en base a estadísticas calcula los jugadores y equipos más mejorados con respecto a la temporada pasada.
- Notifica tanto la clasificación de la temporada de cada conferencia y playoffs como información de cada equipo y sus respectivos jugadores que lo componen.

Todas estas funcionalidades están compuestas con tipos de filtrado como son: año de la temporada, posición del jugador, parte de la cancha, rookies, últimos partidos, mes, . . . , y mucho más

| No | Player | G | Min | Pts | Reb | Ass | Stl | Blk | To | Pf | Dreb | Oreb | Fgm-a | Pct | 3gm-a | Pct | Ftm-a | Pct | eff |
|----|-----------------------|----|------|------|------|------|-----|-----|-----|-----|------|------|-----------|------|----------|------|-----------|------|------|
| 1 | J. Embiid | 34 | 34.0 | 35.3 | 11.3 | 5.7 | 1.1 | 1.8 | 3.7 | 2.9 | 8.7 | 2.6 | 11.8-22.2 | .533 | 1.2-3.3 | .366 | 10.5-11.9 | .883 | 39.8 |
| 2 | N. Jokic | 67 | 34.4 | 26.0 | 12.3 | 9.0 | 1.2 | 0.9 | 2.9 | 2.5 | 9.4 | 2.9 | 10.2-17.6 | .582 | 1.0-2.9 | .354 | 4.5-5.5 | .825 | 38.1 |
| 3 | L. Doncic | 60 | 37.5 | 34.1 | 9.0 | 9.8 | 1.4 | 0.6 | 4.0 | 2.0 | 8.2 | 0.8 | 11.6-23.8 | .488 | 3.9-10.4 | .376 | 7.0-8.9 | .788 | 36.9 |
| 4 | G. Antetokounmpo | 64 | 35.0 | 30.8 | 11.2 | 6.4 | 1.2 | 1.0 | 3.4 | 2.9 | 8.7 | 2.5 | 11.4-18.6 | .616 | 0.5-1.8 | .292 | 7.4-11.2 | .665 | 36.4 |
| 5 | D. Sabonis | 68 | 35.9 | 20.0 | 13.7 | 8.3 | 0.9 | 0.6 | 3.3 | 3.1 | 10.1 | 3.6 | 8.1-13.2 | .609 | 0.4-1.1 | .411 | 3.5-4.9 | .705 | 33.7 |
| 6 | S. Gilgeous-Alexander | 67 | 34.4 | 30.9 | 5.6 | 6.4 | 2.1 | 0.9 | 2.1 | 2.5 | 4.8 | 0.9 | 10.9-20.1 | .544 | 1.3-3.5 | .376 | 7.7-8.8 | .875 | 33.5 |
| 7 | A. Davis | 65 | 35.5 | 24.4 | 12.3 | 3.7 | 1.2 | 2.4 | 2.0 | 2.3 | 9.3 | 3.0 | 9.3-16.8 | .554 | 0.3-1.2 | .269 | 5.5-6.7 | .813 | 33.2 |
| 8 | L. James | 60 | 35.1 | 25.5 | 7.2 | 8.1 | 1.2 | 0.6 | 3.2 | 1.2 | 6.3 | 0.9 | 9.6-18.0 | .532 | 2.2-5.4 | .406 | 4.1-5.5 | .746 | 29.5 |
| 9 | K. Durant | 62 | 37.3 | 27.7 | 6.7 | 5.3 | 0.9 | 1.3 | 3.3 | 1.8 | 6.2 | 0.5 | 10.1-19.3 | .525 | 2.2-5.3 | .417 | 5.3-6.2 | .853 | 28.6 |
| 10 | J. Tatum | 65 | 35.7 | 27.1 | 8.3 | 4.9 | 1.0 | 0.6 | 2.5 | 2.0 | 7.4 | 0.9 | 9.2-19.4 | .474 | 3.2-8.4 | .382 | 5.6-6.7 | .830 | 28.1 |
| 11 | D. Booker | 55 | 35.9 | 27.1 | 4.7 | 7.0 | 0.9 | 0.4 | 2.5 | 3.0 | 3.9 | 0.8 | 9.5-19.4 | .491 | 2.1-5.8 | .355 | 5.9-6.7 | .881 | 27.0 |
| 12 | D. Mitchell | 49 | 35.4 | 27.4 | 5.3 | 6.1 | 1.8 | 0.5 | 2.9 | 2.2 | 4.4 | 0.9 | 9.4-20.2 | .468 | 3.4-9.1 | .373 | 5.1-5.9 | .862 | 26.6 |
| 13 | T. Haliburton | 57 | 32.2 | 20.4 | 3.9 | 11.2 | 1.1 | 0.7 | 2.4 | 1.0 | 3.4 | 0.5 | 7.4-15.5 | .479 | 2.8-7.7 | .359 | 2.9-3.3 | .858 | 26.4 |
| 14 | V. Wembanyama | 61 | 29.1 | 20.7 | 10.4 | 3.5 | 1.3 | 3.5 | 3.5 | 2.2 | 8.2 | 2.2 | 7.5-16.2 | .464 | 1.7-5.1 | .323 | 4.0-5.0 | .805 | 26.2 |
| 15 | A. Sengun | 63 | 32.5 | 21.1 | 9.4 | 5.0 | 1.2 | 0.7 | 2.6 | 3.3 | 6.4 | 2.9 | 8.4-15.6 | .537 | 0.5-1.8 | .297 | 3.9-5.6 | .693 | 25.9 |

Figura 6.1: Estadísticas de los jugadores.

6.2. Basketball Reference

Esta aplicación, Basketball Reference [24], es mucho más completa que la anterior ya que tiene las mismas funcionalidades de estadísticas, información de equipos y jugadores, partidos, sistema de filtrado, datos curiosos (récor­ds, rachas, . . .), etc.

Y además contiene muchas más funcionalidades que aportan una gran interacción con el usuario:

- Informa sobre noticias y tendencias de la actualidad.
- Realiza un análisis de datos exhaustivo de los jugadores a base de estadísticas.
- Aparte de informar sobre la NBA, también muestra información de otras ligas de baloncesto como son ABA, WNBA, competiciones europeas y ligas internacionales.

- Utilizan las redes sociales, como por ejemplo YouTube, para realizar vídeos de todo tipo (podcast, noticias, resúmenes de partidos, entrevistas, highlight, ...).
- Contiene un juego interactivo para los usuarios, que consiste en adivinar jugadores que cumplan ciertas características.
- Por último, también tienen un blog en el que tratan de varios deportes y categorías y puedes subscribirte para comentarios, comunicarte con otros usuarios y consultar dudas.

En general esta aplicación es muy completa ya que contiene información y estadísticas de todo tipo para que la puedan utilizar los analistas y entrenadores, y además proporciona muchas funcionalidades útiles e interesantes para los fanáticos del baloncesto.

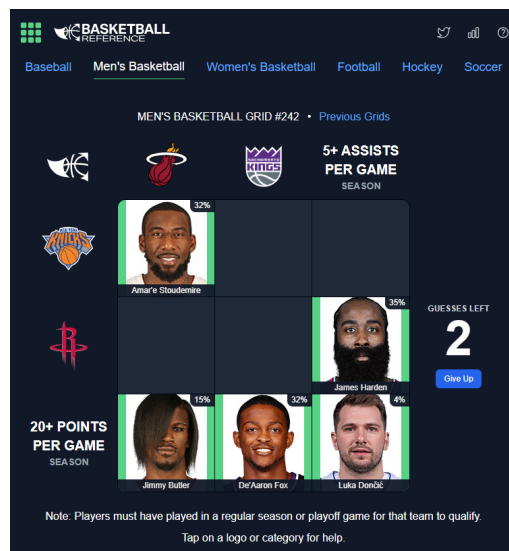


Figura 6.2: Juego interactivo.

6.3. PROBALLERS

PROBALLERS [18] es una mezcla de las anteriores aplicaciones, es decir, muestra estadísticas e información de manera detallada como en Hoop Stats; y muestra más tipos de datos, noticias, distintas ligas de baloncesto, e interactúa más con el usuario como lo hace Basketball Reference.

- Muestra una galería de fotos de cada jugador de la liga.
- Permite instalar widgets en tu dispositivo gratuitamente, sobre partidos de tu equipo favorito.
- Puedes realizar seguimientos de super estrellas de la NBA.
- Disponen de una aplicación de baloncesto disponible en iOS y Android.
- Tienen un diseño muy intuitivo y cómodo que aporta mucha más profesionalidad a la aplicación web.

ALL LEAGUES

NBA

NCAA

EURO LEAGUE

AUSTRALIA NBL

CHINA CBA

JAPAN B1 LEAGUE

LIGA ENDESA

BETCLIC ELITE

NBA

PORTADA

Portada

Lista de jugadores

Lista de equipos

Calendario

Clasificación

Líderes

Récords

SÁB

DOM

LUN

MAR

MIÉ

JUE

VIE

SÁB

DOM

LUN

MAR

16/3

17/3

18/3

19/3

20/3

21/3

22/3

23/3

24/3

25/3

26/3

PARTID.

13 PARTI.

2 PARTID.

8 PARTID.

5 PARTID.

8 PARTID.

7 PARTID.

9 PARTID.

9 PARTID.

4 PARTID.

11 PARTI.

NBA 2023-2024 PARTIDOS

Resultados y calendario

00:00

DET

102

BOS

129

Boxscore

00:00

TOR

103

OKL

123

Boxscore

01:00

SAN

97

MEM

99

Boxscore

01:00

GS

101

GS

101

Boxscore

NBA 2023-2024 LÍDERES

Los mejores promedios

Puntos por partido

Joel Embiid

Philadelphia 76ers

35.3

Rebotes por partido

Domantas Sabonis

Sacramento Kings

13.7

Asistencias por partido

Tyrese Haliburton

Indiana Pacers

11.2

Triplets por partido

Stephen Curry

Golden State

5.4

Figura 6.3: Página inicio.

6.4. Tabla comparativa

Tabla comparativa de las funcionalidades de las distintas páginas web sobre NBA:

| Funcionalidades | Stats Glow Mind | Hoops stats | Basketball Reference | Proballers |
|---|--------------------|----------------|-------------------------|------------|
| Noticias | | | X | X |
| Partidos y resultados | X | X | X | X |
| Clasificación y Playoffs | X | X | X | X |
| Equipos y Jugadores | X | X | X | X |
| Líderes | X | | | |
| Récords | | X | X | X |
| Análisis de datos | X | | X | X |
| Gráficos de datos | X | | | |
| Predicción de resultados | X | | | X |
| Estadísticas en tiempo real | X | X | X | X |
| Estadísticas históricas | | X | X | X |
| Estadísticas de otros deportes o ligas | | | X | X |
| Juego interactivo | | | X | |
| Diseño visual e intuitivo | X | | | X |
| Gratuito | X | X | X | |
| Chat y comunidad | | | | |
| Blog y artículos | | | X | |
| Anuncios | | X | | X |
| Redes sociales | | | X | X |
| Varios idiomas | X | | | X |
| Widget | | | | X |
| App para Android o iOS | | | | X |

Tabla 6.1: Tabla comparativa de funcionalidades

7. Conclusiones y Líneas de trabajo futuras

7.1. Conclusiones

Este proyecto me ha resultado muy útil tanto personalmente como profesionalmente. A lo largo del desarrollo de esta aplicación web dedicada a la NBA, he adquirido una serie de habilidades y conocimientos, marcados en los objetivos, que me han permitido crecer y madurar en diversas áreas.

Desde un punto de vista personal, trabajar en un proyecto que une mi pasión por la NBA con el desarrollo web ha sido muy gratificante. He podido profundizar en el análisis de datos deportivos, lo que me ha permitido obtener una comprensión más profunda de cómo se pueden utilizar las estadísticas para prever resultados y mejorar la experiencia de los aficionados al baloncesto.

En cuanto a las conclusiones técnicas, he aprendido a utilizar varias tecnologías que no se han visto en la carrera, como han sido Angular, TypeScript y Firebase. La experiencia con Angular me ha permitido dominar un framework de frontend muy potente, mientras que el uso de Firebase me ha proporcionado un conocimiento práctico sobre cómo gestionar bases de datos NoSQL, autenticar usuarios y desplegar funciones en la nube. La integración de Bootstrap y Angular Material me han permitido mejorar mis habilidades de diseño, mientras que la integración de Chart.js ha mejorado mis habilidades en la visualización de datos, pudiendo crear gráficos interactivos y dinámicos.

En resumen, este proyecto ha sido una gran experiencia para mi desarrollo técnico y profesional, proporcionándome una serie de competencias que serán valiosas en mi carrera profesional.

7.2. Líneas de trabajo futuras

El proyecto, aunque está completo y funcional, me hubiera gustado añadir más funcionalidades y mejoras en el desarrollo de la aplicación. A continuación, se presentan algunas líneas de trabajo futuras que podrían llevarse a cabo para continuar desarrollando y mejorando la aplicación.

Mejora del Análisis de Rendimiento con Algoritmos de Análisis de datos

El análisis de rendimiento de los jugadores actualmente tan solo muestra las estadísticas a lo largo de la temporada mediante gráficos, esto permite ver la evolución del jugador, pero esto se podría mejorar implementando algoritmos de análisis de datos y Machine Learning, como redes neuronales profundas o métodos de clustering para segmentar y analizar datos con mayor precisión.

Añadir más interacción con el usuario

Para aumentar la interactividad de la aplicación, se podría ampliar el perfil del usuario para incluir más detalles personales, como jugadores y equipos favoritos y así implementar un sistema de notificaciones que informe a los usuarios sobre cuando juega su equipo, seguimiento de sus jugadores favoritos, ..., además de actualizaciones importantes, como resultados de partidos en tiempo real, predicciones y noticias relevantes.

Creación de una Liga Fantasy de la NBA

Una línea futura muy atractiva, la cual tenía mucho interés en desarrollarla, sería la creación de una Liga Fantasy de la NBA integrada dentro de la aplicación. Esto permitiría a los usuarios crear y gestionar sus propios equipos, competir con otros usuarios y recibir puntuaciones basadas en el rendimiento real de los jugadores. La integración de esta funcionalidad aumentaría bastante la experiencia del usuario.

Agregar las estadísticas y récords históricos de cada temporada de la NBA

Incorporar estadísticas y récords históricos de la NBA permitiría a los usuarios acceder a una base de datos completa y detallada de la NBA. Esta funcionalidad podría incluir gráficos y comparativas de estadísticas históricas, permitiendo a los usuarios explorar cómo han cambiado los patrones de juego a lo largo del tiempo.

Mejorar la Seguridad de la Aplicación

Aunque la seguridad y el manejo de errores ya están considerados en el proyecto actual, siempre se puede mejorar. Se podría implementar un sistema más robusto para la gestión de errores y excepciones, asegurando que la aplicación pueda manejar fallos inesperados sin comprometer la experiencia del usuario.

Bibliografía

- [1] Angular. Tour of heroes application and tutorial. <https://v17.angular.io/tutorial/tour-of-heroes>, June 2024.
- [2] Gustavo B. Qué es github y cómo empezar a usarlo. <https://www.hostinger.es/tutoriales/que-es-github>, April 2024.
- [3] berkahicon. Icono login y análisis. <https://www.flaticon.com/authors/berkahicon>, June 2024.
- [4] bsd. Icono predict. <https://www.flaticon.com/authors/bsd>, June 2024.
- [5] Laia Cardona. Microsoft copilot: qué es y cómo funciona. <https://www.cyberclick.es/numerical-blog/microsoft-copilot-que-es-y-como-funciona>, April 2024.
- [6] Google Cloud. Google cloud functions y firebase. <https://cloud.google.com/functions/docs/concepts/functions-and-firebase?hl=es-419#:~:text=Google%20Cloud%20Functions%20es%20la,y%20el%20equipo%20de%20Firebase.>, May 2024.
- [7] Lluís Codina. Flaticon: banco de iconos con licencia de uso gratuito. <https://www.lluiscodina.com/banco-de-iconos-flaticon/>, April 2024.
- [8] Garaje de ideas | Tech. Cómo crear un login con firebase en angular paso a paso. <https://www.youtube.com/watch?v=8VTxuIvMTlc>, June 2024.
- [9] Garaje de ideas | Tech. Deploy angular a firebase hosting. <https://www.youtube.com/watch?v=P4K59gr8sw0>, June 2024.

- [10] Biblioteca/CRAI de la Universidad Pablo de Olavide. Latex: redacción de documentos científicos. <https://guiasbib.upo.es/latex/Trabajando-con-Overleaf>, April 2024.
- [11] Biblioteca/CRAI de la Universidad Pablo de Olavide. Zotero: Qué es zotero. <https://guiasbib.upo.es/zotero>, April 2024.
- [12] desarrolweb.com. Angular. <https://desarrolloweb.com/home/angular>, April 2024.
- [13] desarrolweb.com. Angular cli. <https://desarrolloweb.com/articulos/angular-cli.html>, April 2024.
- [14] desarrolweb.com. Css. <https://desarrolloweb.com/home/css>, April 2024.
- [15] desarrolweb.com. Html. <https://desarrolloweb.com/home/html>, April 2024.
- [16] desarrolweb.com. Typescript. <https://desarrolloweb.com/home/typescript>, April 2024.
- [17] DotTech-ES. Cómo crear y configurar un proyecto en angular y firebase. <https://www.youtube.com/watch?v=E5iSsbPwXe8>, June 2024.
- [18] Benoit Dujardin. Proballers basketball stats. <https://www.proballers.com/es>, April 2024.
- [19] José Facchin. ¿qué es freepik y cómo funciona esta plataforma española para descargar recursos gráficos? <https://josefacchin.com/freepik/#que-es-freepik>, April 2024.
- [20] Yúbal Fernández. Qué es canva, cómo funciona y cómo usarlo para crear un diseño. <https://www.xataka.com/basics/que-canva-como-funciona-como-usarlo-para-crear-diseno>, April 2024.
- [21] Firebase. Cloud storage for firebase. <https://firebase.google.com/docs/storage?hl=es-419>, May 2024.
- [22] Fliqqr. Icono nba. <https://www.flaticon.com/authors/fliqqr>, June 2024.
- [23] Frankier Flores. Qué es visual studio code y qué ventajas ofrece. <https://openwebinars.net/blog/que-es-visual-studio-code-y-que-ventajas-ofrece/>, April 2024.

- [24] Sean Forman. Basketball reference. <https://www.basketball-reference.com/>, April 2024.
- [25] Freepik. Icono leaders y google. <https://www.flaticon.com/authors/freepik>, June 2024.
- [26] Freepik. Red de metal de baloncesto. https://www.freepik.es/foto-gratis/red-metal-baloncesto_2272987.htm#position=2, June 2024.
- [27] Freepik. Vista superior de baloncesto en clase de gimnasia. https://www.freepik.es/foto-gratis/vista-superior-baloncesto-clase-gimnasia_17539112.htm#position=4, June 2024.
- [28] Google. Icono linkedin. <https://www.flaticon.com/authors/google>, June 2024.
- [29] Icon Hubs. Icono logout. <https://www.flaticon.com/authors/icon-hubs>, June 2024.
- [30] IBM. ¿qué es un bosque aleatorio? <https://www.ibm.com/es-es/topics/random-forest>, May 2024.
- [31] IBM. ¿qué es una api? <https://www.ibm.com/es-es/topics/api>, April 2024.
- [32] IBM. ¿qué son las bases de datos nosql? <https://www.ibm.com/es-es/topics/nosql-databases>, April 2024.
- [33] impulsoh. Figma: ¿qué es y para qué sirve? <https://impulsoh.com/que-es-figma-y-para-que-sirve/>, May 2024.
- [34] Javier. ¿qué es postman? ¿cuáles son sus principales ventajas? <https://formadoresit.es/que-es-postman-cuales-son-sus-principales-ventajas/>, June 2024.
- [35] Jorje. Hoops stats - the ultimate source for nba stats. <http://www.hoopsstats.com/>, April 2024.
- [36] KeepCoding. ¿qué es draw.io? <https://keepcoding.io/blog/que-es-drawio/>, May 2024.
- [37] kmg design. Icono teams. <https://www.flaticon.com/authors/kmg-design>, June 2024.

- [38] Pablo Londoño. Qué es bootstrap, para qué sirve y cómo funciona. <https://blog.hubspot.es/website/que-es-bootstrap#:~:text=Bootstrap%20es%20una%20biblioteca%20de,a%20diferentes%20formatos%20de%20navegaci%C3%B3n.>, April 2024.
- [39] Ana Lisseth Juárez Marengo. Angular material: Primeros pasos. <https://community.listopro.com/angular-material-primeros-pasos/#:~:text=Angular%20Material%20es%20un%20m%C3%B3dulo,dise%C3%B1ar%20una%20aplicaci%C3%B3n%20m%C3%A1s%20estructurada.>, April 2024.
- [40] Sara López Mora. Firebase: qué es, para qué sirve, funcionalidades y ventajas. <https://digital55.com/blog/que-es-firebase-funcionalidades-ventajas-conclusiones/>, April 2024.
- [41] Spain ID NBA. Qué es la nba y cómo funciona. <https://spain.id.nba.com/noticias/nba-que-es-como-funciona>, April 2024.
- [42] OpenWebinars. Openwebinars. <https://openwebinars.net/>, June 2024.
- [43] Papa Parse. Papa parse. <https://www.papaparse.com/>, June 2024.
- [44] Pixel perfect. Icono outlook, standings y github. <https://www.flaticon.com/authors/pixel-perfect>, June 2024.
- [45] Laura Cercas Ramos. Cómo añadir gráficos en tu web con chart.js. <https://www.adictosaltrabajo.com/2022/07/01/como-anadir-graficos-en-tu-web-con-chart-js/>, April 2024.
- [46] RapidAPI. Api-nba. <https://rapidapi.com/api-sports/api/api-nba>, June 2024.
- [47] RapidAPI. What is rapidapi.com? <https://docs.rapidapi.com/docs/what-is-rapidapi>, April 2024.
- [48] Kreev Studio. Icono games y playoffs. <https://www.flaticon.com/authors/kreev-studio>, June 2024.
- [49] KeepCoding Team. ¿qué es cloud firestore? <https://keepcoding.io/blog/que-es-cloud-firestore/>, April 2024.
- [50] KeepCoding Team. ¿qué es el protocolo http? <https://keepcoding.io/blog/que-es-el-protocolo-http/>, April 2024.

- [51] Daniella Terreros. Qué es trello, para qué sirve y cómo funciona. <https://blog.hubspot.es/marketing/que-es-trello>, April 2024.
- [52] Carolina González Valenzuela. ¿qué es latex y cómo funciona esta útil herramienta para crear documentos? <https://computerhoy.com/tecnologia/latex-como-funciona-util-herramienta-crear-documentos-1165366>, April 2024.