

TRABAJO FIN DE GRADO

GRADO EN INGENIERÍA INFORMÁTICA

# **WebApp para el análisis del rendimiento futbolístico y predicción de resultados**

Unai de la Mata Vicente

San Fernando, septiembre 2025

TRABAJO FIN DE GRADO

GRADO EN INGENIERÍA INFORMÁTICA

# **WebApp para el análisis del rendimiento futbolístico y predicción de resultados**

CODIRECTOR: Iván Ruíz Rube  
CODIRECTOR: Juan Antonio Caballero Hernández  
AUTOR: Unai de la Mata Vicente

San Fernando, septiembre de 2025



# Declaración personal de autoría

---

Unai de la Mata Vicente con DNI 49560645s, estudiante del Grado de Ingeniería Informática en la Escuela Superior de Ingeniería de la Universidad de Cádiz, como autor de este documento académico titulado “WebApp para el análisis del rendimiento futbolístico y predicción de resultados” y presentado como Trabajo Final de Grado

## DECLARO QUE

Es un trabajo original, que no copio ni utilizo parte de obra alguna sin mencionar de forma clara y precisa su origen tanto en el cuerpo del texto como en su bibliografía y que no empleo datos de terceros sin la debida autorización, de acuerdo con la legislación vigente. Asimismo, declaro que soy plenamente consciente de que no respetar esta obligación podrá implicar la aplicación de sanciones académicas, sin perjuicio de otras actuaciones que pudieran iniciarse.

Aclaro además que, durante la elaboración del documento, he utilizado la herramienta ChatGPT únicamente como apoyo para la revisión lingüística y corrección de estilo y de código fuente, manteniendo en todo momento la autoría, el contenido original y la responsabilidad académica del trabajo.

En Puerto Real, a septiembre de 2025

Fdo: Unai de la Mata Vicente



# Agradecimientos

---

A **Ainara**, mi novia y mi compañera de vida, gracias por ser parte de este camino y hacerlo todo más bonito. Por aguantarme y ayudarme a superar los malos momentos y por ser la primera persona con la que celebro y disfruto cada pequeño logro. Sin ti esto no hubiera sido posible.

A **mi familia**: mis padres Gorka y Susana, y mi hermana Nerea. Gracias por estar siempre ahí cuando me hace falta, por apoyarme, aconsejarme y por motivarme a levantarme cuando caigo.

A **todos mis amigos**, por darme ánimos cuando lo he necesitado y por alegrarse de mis logros como si fueran suyos.

A **mi familia política**: mis cuñaditos Andrea y Lito, mis suegros Cristóbal y María José, mis abuelos y abuelas adoptivas Cristóbal, Félix, Isa y Toñi y mis tíos adoptivos Noelia, Isa y Jesús; por hacerme sentir siempre como uno más de la familia, por interesarse siempre por mis avances en la carrera y por darme ánimos para llegar al final.

A **Zoe**, mi perra, porque aunque no entiendas nada y no sepas las crisis y agobios que ha sufrido tu dueño durante todo este tiempo, tu simple presencia hacía todo más fácil.

A **Paco**, a **Laura**, a **David** y a toda persona con la que he compartido buenos momentos durante esta carrera. Habéis sido una ayuda muy necesaria.



# Resumen

---

Los datos siempre han sido una fuente de información interesante y durante estos últimos años gracias al avance de la tecnología hemos sabido sacar más provecho de ellos y usarlos para perfeccionar nuestros objetivos. En el ámbito deportivo, no hay club profesional que no se fije en ellos para intentar dar con la clave que les permita mejorar y obtener buenos resultados.

Durante este proyecto, desarrollaré una aplicación web que contendrá el análisis de datos y estadísticas de un equipo de fútbol. En mi caso, usaré de ejemplo el Cádiz CF pero se proporcionará al usuario la posibilidad de cambiar de equipo a analizar. Gracias a este análisis podremos visualizar gráficas y fijarnos en los puntos fuertes y débiles sobre los que el equipo debe trabajar para obtener los objetivos esperados, así como podremos realizar y observar simulaciones que nos permitirán predecir el resultado final que se obtendrá en ciertas situaciones, basándonos en los datos que tenemos.

Usaremos el lenguaje R, muy usado en el ámbito del análisis de datos, así como la herramienta OpenRefine que nos permitirá procesar y limpiar estos datos. Finalmente usaremos el framework Shiny, compatible con R, para la creación de la WebApp.

**Palabras clave:** Cadiz cf, Big Data, fútbol, análisis, R, predicción.





# Índice general

---

Índice de figuras	x
-------------------	---

Índice de tablas	xi
------------------	----

## I Prolegómeno 1

### 1. Introducción 2

1.1. Motivación . . . . .	2
---------------------------	---

1.2. Alcance y Objetivos . . . . .	2
------------------------------------	---

1.3. Glosario de términos . . . . .	3
-------------------------------------	---

1.4. Organización del documento . . . . .	4
---	---

### 2. Antecedentes 7

2.1. Contexto . . . . .	7
-------------------------	---

2.2. Estado de la técnica . . . . .	7
-------------------------------------	---

### 3. Plan de gestión de proyecto 11

3.1. Metodología de desarrollo . . . . .	11
--	----

3.2. Tecnologías . . . . .	12
----------------------------	----

3.3. Planificación del proyecto . . . . .	13
---	----

3.4. Organización . . . . .	14
-----------------------------	----

3.5. Costes . . . . .	14
-----------------------	----

3.6. Riesgos . . . . .	14
------------------------	----

3.7. Gestión de la configuración . . . . .	15
--	----

## II Desarrollo 16

### 4. Requisitos del Sistema 17

4.1. Objetivos de Negocio . . . . .	17
-------------------------------------	----

4.2. Objetivos del Sistema . . . . .	17
--------------------------------------	----

4.3. Requisitos funcionales . . . . .	18
---------------------------------------	----

4.4. Requisitos no funcionales . . . . .	18
--	----

4.5. Requisitos de información . . . . .	19
--	----

4.6. Reglas de negocio . . . . .	19
----------------------------------	----

### 5. Análisis del Sistema 23

5.1. Modelo Conceptual . . . . .	23
----------------------------------	----

5.2. Modelo de Casos de Uso . . . . .	23
---------------------------------------	----

5.2.1. Actor Principal . . . . .	23
----------------------------------	----

5.2.2. Casos de Uso . . . . .	23
-------------------------------	----

5.3. Modelo de Interfaz de Usuario . . . . .	26
--	----

<b>6. Diseño del Sistema</b>	<b>31</b>
6.1. Arquitectura del Sistema . . . . .	31
6.1.1. Diseño de alto nivel . . . . .	31
6.2. Diseño Físico de Datos . . . . .	34
6.2.1. Tablas de datos . . . . .	34
6.3. Diseño de la Interfaz de Usuario . . . . .	37
6.3.1. Principios de diseño . . . . .	37
6.3.2. Estilo Visual . . . . .	38
6.3.3. Componentes visuales . . . . .	38
6.3.4. Prototipos . . . . .	38
<b>7. Construcción del Sistema</b>	<b>43</b>
7.1. Entorno de Construcción . . . . .	43
7.2. Código Fuente . . . . .	43
7.3. Gráficas . . . . .	45
7.4. Scripts de Base de datos . . . . .	46
<b>8. Pruebas del Sistema</b>	<b>49</b>
8.1. Estrategia . . . . .	49
8.2. Pruebas Unitarias . . . . .	49
8.3. Pruebas de Integración . . . . .	50
8.4. Pruebas de Sistema . . . . .	50
8.4.1. Pruebas Funcionales . . . . .	50
8.4.2. Pruebas No Funcionales . . . . .	51
8.5. Pruebas de Aceptación . . . . .	52
<b>9. Despliegue del Sistema</b>	<b>55</b>
9.1. Arquitectura Física . . . . .	55
9.1.1. Elementos hardware . . . . .	55
9.1.2. Elementos software . . . . .	55
9.1.3. Comunicación con el usuario . . . . .	55
9.2. Instrucciones de despliegue . . . . .	56
9.2.1. Requisitos previos . . . . .	56
9.2.2. Inventario de componentes . . . . .	56
9.2.3. Procedimientos de instalación . . . . .	57
9.2.4. Pruebas de implantación . . . . .	58
9.3. Instrucciones para la operación del sistema y mantenimiento del nivel de servicio . . . . .	58
9.3.1. Operación diaria . . . . .	58
9.3.2. Copias de seguridad (backups) . . . . .	58
9.3.3. Descarga de datos . . . . .	58
9.3.4. Mantenimiento preventivo . . . . .	59
9.3.5. Seguridad . . . . .	59
9.3.6. Documentación y solución de posibles incidencias . . . . .	59

<b>III</b>	<b>Epílogo</b>	<b>60</b>
<b>10.</b>	<b>Conclusiones</b>	<b>61</b>
10.1.	Objetivos alcanzados . . . . .	61
10.2.	Lecciones aprendidas . . . . .	61
10.3.	Trabajo futuro . . . . .	61
	<b>Información sobre Licencia</b>	<b>65</b>
	<b>Bibliografía</b>	<b>67</b>
<b>IV</b>	<b>Anexos</b>	<b>69</b>
<b>A.</b>	<b>Manual del desarrollador</b>	<b>70</b>
A.1.	Introducción . . . . .	70
A.2.	Preparación del entorno de trabajo . . . . .	70
A.3.	Consideraciones generales sobre el desarrollo . . . . .	71
A.4.	Instrucciones para construcción . . . . .	72
<b>B.</b>	<b>Manual de usuario</b>	<b>75</b>
B.1.	Introducción . . . . .	75
B.2.	Uso del sistema . . . . .	75

# Índice de figuras

---

3.1. Modelo de Desarrollo en Cascada . . . . .	11
3.2. Diagrama de Gantt . . . . .	13
5.1. Modelo Conceptual . . . . .	23
5.2. Modelo Casos de Uso . . . . .	26
5.3. Mockup página de inicio . . . . .	27
5.4. Mockup página de análisis . . . . .	27
5.5. Mockup página de predicción . . . . .	28
5.6. Mockup página de glosario . . . . .	28
6.1. Diagrama de contexto . . . . .	32
6.2. Diagrama de Contenedores . . . . .	33
6.3. Diagrama de Componentes . . . . .	34
6.4. Prototipo de la página de inicio . . . . .	39
6.5. Prototipo de la página de análisis . . . . .	39
6.6. Prototipo de la página de predicción . . . . .	40
6.7. Prototipo de la página de glosario . . . . .	40
B.1. Inicio . . . . .	76
B.2. Análisis . . . . .	76
B.3. Predicción . . . . .	77
B.4. Glosario . . . . .	77
B.5. Cambio de equipo . . . . .	78

# Índice de tablas

---

3.1. Coste recursos humanos . . . . .	14
3.2. Coste recursos materiales . . . . .	14
6.1. Tabla de Partidos . . . . .	35
6.2. Tabla de Tiros . . . . .	36
6.3. Tabla de Datos Jugador . . . . .	36
6.4. Tabla de Tiros Jugador . . . . .	37
6.5. Tabla de Goles . . . . .	37
8.1. Verificación de requisitos funcionales . . . . .	51
8.2. Verificación de requisitos no funcionales . . . . .	52
8.3. Resultados de pruebas de aceptación . . . . .	53



# Parte I

## Prolegómeno



# 1. Introducción

---

A continuación, se describe la motivación del presente proyecto y su alcance. También se incluye un glosario de términos y la organización del resto de la presente documentación.

## 1.1. Motivación

En este proyecto uniré dos de mis mayores intereses, el fútbol y los datos. Desde hace varios años me ha resultado interesante cómo la parte dedicada al análisis de datos y el Big Data del deporte está influyendo en el fútbol, una disciplina que tradicionalmente se había basado más en la experiencia y el juego antes de dejar paso a este análisis. He querido indagar más en ella para descubrir las ventajas que nos puede aportar este análisis y cómo los resultados obtenidos de estos estudios pueden ser útiles para mejorar el rendimiento de un club de fútbol.

El uso del Big Data abarca múltiples áreas: desde el análisis del rendimiento de un futbolista concreto hasta el análisis táctico del equipo en general o de los próximos rivales. Hoy en día existen empresas como Opta Sports, StatsBomb, Wyscout o InStat que se dedican a la venta de estos datos para ayudar en su trabajo tanto a miembros del cuerpo técnico de un club como a los medios de información deportivos.

Hoy en día podría decirse que no hay equipo en las grandes ligas que no haga uso de este análisis tanto para mejorar su rendimiento como para tomar mejores decisiones en cuanto a fichajes, alineaciones, tácticas empleadas, etc.

A lo largo de este proyecto desarrollaré una WebApp adaptada al usuario final para que el aficionado promedio pueda visualizar y entender mejor este aspecto del fútbol, viendo información relevante tras el análisis de datos, usando como ejemplo los datos del Cádiz CF debido a su cercanía con nuestra universidad y conmigo mismo. El usuario, además de análisis, podrá visualizar predicciones de resultados esperados basados en datos de temporadas anteriores y, en el caso de desearlo, podrá seleccionar otro equipo para analizar.

## 1.2. Alcance y Objetivos

Como he comentado anteriormente, para este proyecto he decidido centrarme en los datos del Cádiz CF como ejemplo, más concretamente en los de la temporada 2023-2024, ya que al ser la última temporada del equipo en primera división finalizada, puedo tener todos los datos completos tanto individuales de cada jugador como colectivos de todo el equipo.

Igualmente, se añadirá la posibilidad de cambiar de equipo y cargar sus respectivos datos para ser analizados, así como de cargar datos de temporadas anteriores del equipo y combinarlas para que se tengan en cuenta a la hora de hacer los análisis y predicciones.

Procesaré y analizaré estos datos para sacar conclusiones de aspectos clave del juego del equipo durante la temporada, para ver las carencias y detectar en qué falló el equipo y qué aspectos pueden mejorarse.

Como se ha comentado previamente, se incluirá un apartado de “predicciones” en el que se podrán visualizar los resultados esperados de distintos aspectos del juego, teniendo en cuenta los datos de dos o más temporadas anteriores.

Para el desarrollo de la aplicación usaré el lenguaje R y su framework integrado Shiny que permite realizar aplicaciones web con este lenguaje.

Todos los análisis y predicciones contarán con explicaciones y conclusiones que ayuden al usuario promedio a entender las gráficas y visualizar en qué aspectos del juego el equipo ha rendido por debajo de lo esperado.

Además, el usuario contará en la propia web con un glosario de términos que le ayudará en todo momento a entender todo lo que se le presenta sin necesidad de conocimientos previos.

### 1.3. Glosario de términos

- **CF:** Club de fútbol.
- **RF:** Requisito funcional.
- **RNF:** Requisito no funcional.
- **RN:** Regla de negocio.
- **RI:** Requisito de información.
- **Big Data:** Conjunto de técnicas y tecnologías destinadas a gestionar y analizar grandes volúmenes de datos, permitiendo descubrir patrones, tendencias o tomar decisiones informadas.
- **WebApp:** Aplicación web accesible desde un navegador que permite al usuario interactuar con funcionalidades sin necesidad de instalar software.
- **xG (Expected Goals):** Métrica avanzada que estima la probabilidad de que un disparo termine en gol, considerando la posición, el ángulo y otras variables del disparo.
- **Visualización de datos:** Representación gráfica de la información para facilitar la comprensión de patrones y tendencias.

## 1.4. Organización del documento

La estructura que se sigue en este documento es la siguiente:

1. **Introducción:** Se define la motivación, el alcance y los objetivos de mi proyecto.
2. **Antecedentes:** Un análisis del entorno y el contexto en el que se desarrolla el proyecto, así como ejemplos de Webs ya existentes.
3. **Plan de gestión del proyecto:** Se describe la metodología del desarrollo, las tecnologías usadas, la planificación, la organización, los costes y los riesgos.
4. **Requisitos del sistema:** Se detallan los requisitos funcionales y no funcionales, los requisitos de información y las reglas de negocio.
5. **Análisis del sistema:** Se muestran diversos modelos y diagramas: el modelo conceptual, modelo de casos de uso, modelo de interfaz de usuario y modelo de comportamiento.
6. **Diseño del sistema:** Se detalla la arquitectura del sistema, presentando el diseño físico de datos y el diseño de la interfaz de usuario.
7. **Construcción del sistema:** Se describe el entorno de construcción, así como la organización del código fuente.
8. **Pruebas del sistema:** Se describen los distintos tipos de pruebas realizadas.
9. **Despliegue del sistema:** Se describe la arquitectura física del sistema con sus correspondientes instrucciones para el despliegue.
10. **Conclusiones:** Se detallan los objetivos alcanzados, las lecciones aprendidas y el posible trabajo a realizar en el futuro.





## 2. Antecedentes

---

### 2.1. Contexto

Hoy en día no es difícil obtener datos y estadísticas del rendimiento de un equipo. Tampoco es raro ver a analistas y expertos discutir estos datos para intentar sacar sus conclusiones y “predecir” el fútbol, siendo este un deporte que no es una ciencia cierta debido a eventos inesperados y a su naturaleza impredecible. Esto ha provocado que muchos aficionados e incluso futbolistas o técnicos renieguen de estas estadísticas y de las herramientas para analizarlas, argumentando que “esto no es fútbol”. El debate sobre el uso del Big Data está ya muy presente en este deporte, y aunque es innegable lo que estas herramientas pueden aportar [AlfonsoX \(2024\)](#), hay quien las rechaza y prefiere un fútbol más tradicional y libre de estudios. [Segurola \(2024\)](#).

Mi objetivo con este proyecto es acercar estas estadísticas a los aficionados de a pie, simplificándolas y ayudándoles a ver cómo pueden aprender de ellas para analizar el rendimiento de su equipo y predecir futuros resultados en base a ellas. La aplicación web estará diseñada siguiendo principios de usabilidad y diseño centrado en el usuario, con un enfoque que permita explorar, comparar y entender los datos de forma progresiva. El objetivo no es reemplazar el análisis tradicional del fútbol, sino complementarlo a través del uso adecuado de datos, acercando al aficionado y superando el obstáculo de la complejidad y el desconocimiento.

Al ofrecer visualizaciones interpretables y acompañarlas de explicaciones, se busca mejorar el conocimiento del aficionado y revalorizar el uso de las estadísticas en el análisis del fútbol.

### 2.2. Estado de la técnica

Actualmente, existen varias webs que ofrecen datos y estadísticas de los distintos equipos de fútbol que participan en cada competición. Desde datos más concretos a datos más generales.

Entre ellas podemos encontrar la web Fbref ([FBRef \(2022\)](#)), que estaré usando en mi proyecto para obtener los datos a analizar. He decidido usar esta web para obtener los datos, ya que contiene multitud de estadísticas organizadas en tablas y divididas por diversos aspectos como estadísticas globales de la competición, estadísticas de tiros, estadísticas de los jugadores individuales, etc.

Otra fuente a considerar sería la propia web oficial de La Liga, que también tiene una sección para ver estos datos en un apartado que nos ofrece estadísticas tanto básicas como avanzadas de los mejores jugadores de la competición. Sin embargo, no ofrece un análisis tan profundo de cada equipo, cosa que sí haremos nosotros.

En la propia web oficial del Cádiz CF, nos encontramos con una sección de estadísticas. Sin embargo, contiene muy poca información, al ofrecer únicamente datos generales sin centrarse en la profundidad analítica, por lo que podría ser complementada con nuestro proyecto. Además, al tratarse de la web del Cádiz, es poco escalable ya que solo contiene datos y estadísticas de dicho equipo.

El uso de Big Data ha ido ganando terreno también en los medios de comunicación, no es raro encontrarse en periódicos deportivos artículos con titulares como: “¿Cuánta posibilidad tiene cada equipo de descender? El superordenador nos da los porcentajes” ([Garcia-Planas \(2025\)](#)). Estos son simples artículos puntuales que, aún pudiendo ser un poco sensacionalistas, pueden acercar más esta ciencia al público general y hacen que sea más fácil para el aficionado promedio normalizar su presencia en el ámbito del fútbol.

A nivel profesional, el análisis de datos se aplica en decisiones estratégicas como:

- **Scouting de jugadores:** Detectar talentos o fichajes potenciales en base a métricas objetivas. Plataformas como SkillCorner ([SAS \(2022\)](#)) permiten acceder a bases de datos con miles de futbolistas evaluados según indicadores cuantitativos (pases completados, duelos ganados, xG, xA, etc.), lo que facilita la identificación de perfiles adecuados al estilo de juego deseado por el club.
- **Análisis táctico:** Estudio del comportamiento del equipo propio o rival mediante mapas de calor, secuencias de pases o zonas de presión. El análisis de datos permite descomponer el juego en elementos objetivos que ofrecen información valiosa sobre el comportamiento colectivo.
- **Prevención de lesiones:** El control de la carga física mediante datos biométricos y GPS (distancia recorrida, aceleraciones, sprints, cambios de ritmo) permite prevenir lesiones y gestionar adecuadamente la fatiga.
- **Optimización del rendimiento:** Análisis del impacto de ciertas decisiones tácticas o alineaciones en función del rival. Por ejemplo, analizar si un determinado sistema táctico (4-4-2, 3-5-2, etc.) rinde mejor frente a ciertos estilos de juego rivales, o si determinadas combinaciones de jugadores funcionan mejor y aumentan los xG (goles esperados) y diferentes estadísticas que influyan positivamente en el rendimiento del equipo.

Este proyecto propone el desarrollo de una WebApp enfocada al aficionado promedio, ayudándolo a entender esta nueva ciencia y permitiendo que pueda disfrutar del análisis del rendimiento de su equipo. Mientras que otras fuentes se centran principalmente en los datos individuales de jugadores estrella o en estadísticas globales sin contexto, nuestra WebApp proporcionará un análisis detallado y específico. Usando como ejemplo los datos del Cádiz CF, proporcionaré un análisis claro y detallado de su rendimiento, combinando datos avanzados con gráficas y tablas interpretables.

Dividiré los datos en varias categorías diferenciales: Tiros, Resultados, Goles, etc. Cada categoría contará con una serie de análisis específicos, incluyendo gráficos interpretables. Ofrecerá además explicaciones y textos de ayuda para que los usuarios aprendan a sacar conclusiones interpretando los gráficos generados con las estadísticas del equipo.

Se ofrecerán también predicciones de cada una de las categorías, teniendo en cuenta temporadas anteriores. El usuario podrá además elegir las temporadas que prefiera para tener en cuenta datos de aquella o aquellas elegidas, tanto para los análisis como para las predicciones.

Además, el usuario tendrá la posibilidad de cambiar de equipo para elegir otros datos de otro equipo a analizar.





## 3. Plan de gestión de proyecto

---

En esta sección se describen todos los aspectos relativos a la gestión del proyecto: metodología, organización, costes, planificación, riesgos y aseguramiento de la calidad.

### 3.1. Metodología de desarrollo

Para el desarrollo de este proyecto se utiliza la metodología en cascada (Figura 3.1) , ya que nos ofrece un enfoque lineal con la posibilidad de iterar en cada fase antes de pasar a la siguiente. Además, esta metodología nos permite corregir errores descubiertos en fases posteriores.

1. **Requisitos:** En esta fase se analizan los requisitos del proyecto, incluyendo los objetivos y las funcionalidades requeridas.
2. **Diseño:** Basándonos en los requisitos previamente definidos, se desarrolla el diseño del sistema.
3. **Implementación:** Durante esta fase se desarrolla la programación, teniendo en cuenta las especificaciones definidas anteriormente.
4. **Verificación o Pruebas:** Durante esta fase se busca encontrar y corregir errores de nuestro software, así como asegurar que se cumplen los requisitos.
5. **Mantenimiento:** Se corrigen los posibles errores que vayan surgiendo, y se realizan ajustes y mejoras para que la aplicación pueda seguir en funcionamiento.

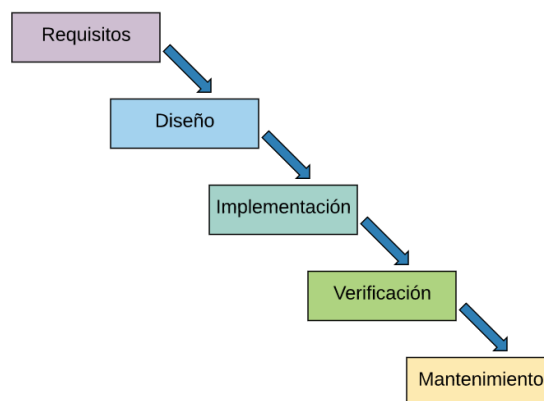


Figura 3.1: Modelo de Desarrollo en Cascada

## 3.2. Tecnologías

En esta sección justificaré las razones de las tecnologías empleadas, cada una destinada a cumplir una función específica respecto al proceso del análisis de datos, pero con capacidad para complementarse perfectamente entre sí.

### ■ OpenRefine

OpenRefine ([OpenRefine \(2023\)](#)) es una aplicación de escritorio de código abierto usada para la limpieza y el procesamiento de datos. Es una aplicación muy útil y fácil de usar, que permite corregir fallos de formato en los datos descargados, así como simplificar o eliminar columnas de una tabla específica.

Esta herramienta permite trabajar con el formato .csv tanto para la importación como para la exportación de datos, por lo que funciona bien con los datos que obtenemos de una página específica y nos permitirá exportarlos tras ser limpiados y procesados en un formato sencillo para su posterior análisis.

Los primeros datos usados los limpiaré manualmente haciendo uso de esta aplicación para, posteriormente, usar scripts que descarguen, limpien y procesen los datos de manera automática.

### ■ Lenguaje R y R Studio

R ([foundation \(1993\)](#)) es un lenguaje de programación enfocado al análisis estadístico, por lo que se ha considerado que es ideal para este proyecto. Hoy en día, se trata de uno de los lenguajes más utilizados en la investigación científica, siendo muy importante en campos como el análisis de datos o machine learning.

Haremos uso también de R Studio, un entorno de desarrollo integrado (IDE) para este lenguaje específico que facilita su uso dándonos una manera más visual y sencilla de organizar nuestro código y ejecutarlo.

### ■ Shiny

Shiny ([R.studio \(2012\)](#)) es un framework web que nos permite la creación de WebApps compatibles con el lenguaje R. Esta herramienta de código abierto nos facilita la creación de nuestra aplicación web.

### ■ OverLeaf

Overleaf ([Overleaf \(2013\)](#)) es una plataforma que permite la creación y edición de documentos LaTeX de manera más intuitiva y sencilla, lo que nos será útil para el desarrollo de la documentación del proyecto.

### ■ Github

Github ([Github \(2008\)](#)) nos ofrece un sistema de control de versiones y colabo-

ración en proyectos software. Esto nos permite realizar un seguimiento eficiente de los cambios realizados a lo largo del proyecto, actualizando e identificando estos cambios con descripciones una vez se realizan, así como mantener un respaldo de todo nuestro código para evitar pérdidas.

- **Balsamiq**

Balsamiq ([Balsamiq \(2008\)](#)) es una aplicación de escritorio que permite crear mockups o prototipos de interfaces web de forma rápida y visual. Su principal utilidad es ofrecer una representación preliminar del diseño de la página, facilitando así la comprensión de cómo se verá y funcionará la web en el navegador antes de comenzar su desarrollo real.

- **Draw.io**

Draw.io ([Drawio \(2000\)](#)) es una página que nos ofrece la posibilidad de realizar modelos y diagramas de manera online, sencilla y gratuita. En nuestro proyecto nos servirá para realizar los distintos modelos (conceptual, casos de uso, etc.) que requiere la documentación.

### 3.3. Planificación del proyecto

En la Figura 3.2 , se puede visualizar el diagrama de Gantt que muestra los tiempos empleados en cada fase del proyecto, incluyendo el tiempo de aprendizaje para cada herramienta. Podemos apreciar también la gestión y planificación del tiempo y los recursos, así como de la organización seguida para la realización de las distintas tareas.

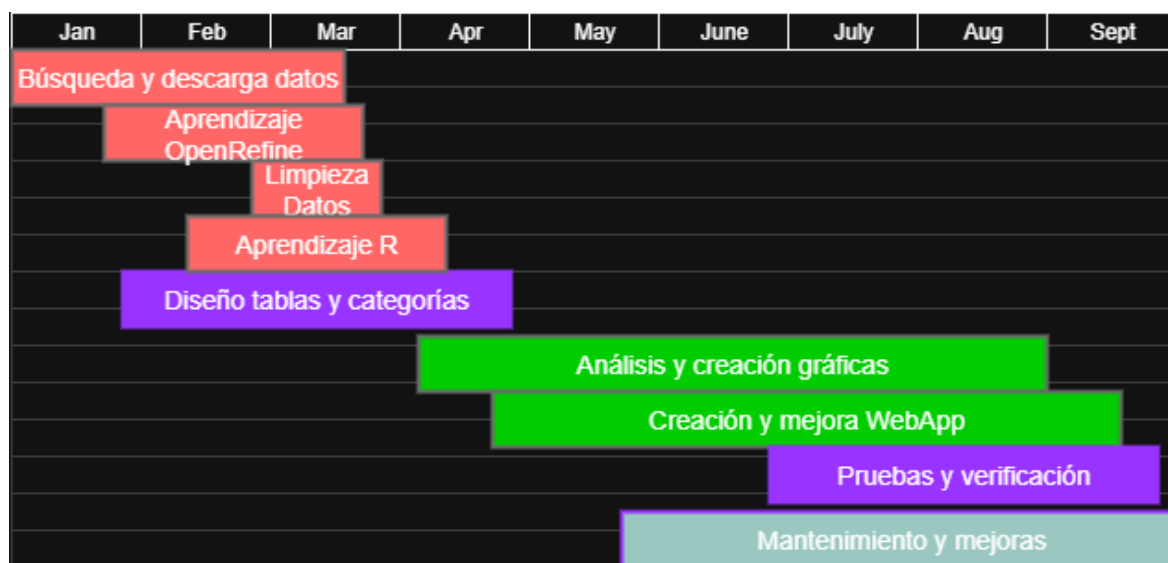


Figura 3.2: Diagrama de Gantt

## 3.4. Organización

Como herramientas principales se utilizan un ordenador de sobremesa y un portátil tanto para la programación como para el resto de tareas a realizar durante el proyecto, permitiendo así el trabajo en distintas ubicaciones de manera flexible dependiendo de las circunstancias.

## 3.5. Costes

Para el cálculo de los costes, tendremos en cuenta dos tipos: costes humanos y costes materiales.

- **Costes Humanos:** Para calcularlos tomaremos en cuenta el sueldo medio de un ingeniero junior en España por hora. [talent \(2025\)](#) El cálculo tendrá en cuenta que se trabajará aproximadamente 20 horas semanales a lo largo de 5 meses. En la Tabla 3.1 vemos este cálculo desglosado.

Horas	Coste	Total
450	13€/hora	5850€

Tabla 3.1

*Coste recursos humanos*

- **Costes Materiales:** Para los costes materiales tendremos en cuenta el gasto por uso de electricidad, de wifi y el equipo utilizado (amortización durante el tiempo de trabajo). Usaremos un gasto aproximado para la electricidad y el wifi teniendo en cuenta lo que consume un equipo propio de un informático por hora. En la Tabla 3.2 podemos ver este cálculo desglosado.

Equipo	Electricidad	Wifi	Total
100€	25€	135€	260€

Tabla 3.2

*Coste recursos materiales*

## 3.6. Riesgos

Los principales riesgos analizados e identificados para este proyecto han sido los siguientes:

- **Tiempo de desarrollo excesivo:** Como en cualquier proyecto, la falta de tiempo o la necesidad de tiempo extra para terminarlo es un posible riesgo. Para evitarlo hemos realizado un seguimiento adecuado del proyecto para tener capacidad de reaccionar a los posibles imprevistos, desglosándolo en varios hitos.
- **Modificación de requisitos:** Los requisitos pueden sufrir variaciones y/o la aparición de otros nuevos a lo largo del desarrollo del proyecto debido al descubrimiento de nuevas necesidades del usuario o a problemas técnicos, es por ello que hay que ser capaz de adaptarse a estos cambios.
- **Problemas de seguridad:** La seguridad siempre es un aspecto a tener en cuenta a la hora del desarrollo de una aplicación web, ya que problemas como vulnerabilidades en el software o ataques malware pueden tener consecuencias si no son tratados con cautela para prevenirlos.

### 3.7. Gestión de la configuración

Como se ha comentado previamente, el uso de Github será esencial para realizar el control de las versiones del código fuente. Se dispondrá de un repositorio propio que será actualizado de manera constante, garantizando que el código refleje los cambios más recientes y permitiendo revertir a versiones anteriores en caso de ser necesario.

Para minimizar errores y problemas, las actualizaciones del repositorio se realizarán únicamente después de que cada nueva implementación haya sido probada, tanto a nivel unitario como integrando la aplicación en su conjunto.

El repositorio donde se alojará la aplicación se encuentra en el siguiente enlace: <https://github.com/UnaiZoat/TFG-UNAI-CADIZ>

# Parte II

## Desarrollo

## 4. Requisitos del Sistema

---

En este capítulo se presentan los objetivos y el catálogo de requisitos del nuevo sistema informático.

### 4.1. Objetivos de Negocio

El desarrollo de nuestra WebApp persigue alcanzar los siguientes objetivos de negocio:

- **Mayor accesibilidad para el análisis de datos futbolísticos:** Facilitar el acceso a estadísticas y análisis avanzados para el aficionado promedio, sin necesidad de conocimientos técnicos o experiencia previa en el ámbito del análisis de datos. Nuestro objetivo es ayudar a que entiendan y vean las ventajas del Big Data en el fútbol.
- **Mejorar la experiencia del aficionado:** Proponer una nueva forma de seguir a su equipo, complementando la información tradicional con gráficos y análisis que ayuden a comprender el rendimiento.
- **Fomentar el conocimiento en Big Data:** Ayudar a aficionados e interesados en el análisis de datos deportivos a que empiecen a entender esta ciencia, convirtiendo la WebApp en una herramienta didáctica que sirva como introducción.

### 4.2. Objetivos del Sistema

El objetivo principal es desarrollar una WebApp interactiva que permita a los usuarios acceder, visualizar e interpretar datos estadísticos sobre el rendimiento de un equipo de fútbol, tomando en principio como ejemplo los datos del Cádiz CF en la temporada 2023/2024. Los objetivos generales del sistema son:

- **Visualización clara y accesible de datos:** Permitir al usuario ver datos relevantes del rendimiento del equipo, acompañados de gráficas intuitivas que faciliten la interpretación de estos. Para cumplir este objetivo, cada gráfica incluirá textos explicativos que ayuden a entenderlas y sacar conclusiones de cada análisis.
- **Organización de la información:** Agrupar datos obtenidos de distintas fuentes y estructurarlos en distintas categorías para facilitar su consulta.
- **Interfaz amigable:** Diseñar una interfaz pensada para el aficionado promedio, fácil de navegar e incluyendo explicaciones, leyendas y un glosario para que cualquiera pueda entender el análisis de estos datos.



## 4.3. Requisitos funcionales

### RF1

El sistema debe ofrecer gráficos interactivos para visualizar y sacar conclusiones de las distintas estadísticas.

### RF2

El sistema debe mostrar debajo de cada gráfico una pequeña explicación y un texto orientativo para ayudar al usuario a sacar conclusiones de cada análisis.

### RF3

Las conclusiones y explicaciones deben ser escritas en un lenguaje no técnico, que permita a cualquier usuario comprender su significado sin necesidad de tener conocimientos previos en estadística.

### RF4

El sistema debe permitir la incorporación de nuevos datos, actualizando los correspondientes gráficos tras la selección por parte del usuario de un nuevo equipo y/o temporada.

### RF5

El sistema debe incluir una página de inicio con una pequeña introducción sobre el contenido de la web, el significado de Big Data y su utilidad. Estará escrito en lenguaje claro y no técnico.

### RF6

El sistema debe ofrecer la posibilidad de 'predecir' las distintas categorías ofreciendo en cada una, una predicción de los resultados esperados mediante gráficas y explicaciones. Para ello, se tendrán en cuenta los datos de las temporadas anteriores.

### RF7

El sistema debe ofrecer la posibilidad de elegir la temporada o temporadas de las que se quieren tener en cuenta los datos a analizar y predecir.

## 4.4. Requisitos no funcionales

### RNF1

La interfaz debe estar diseñada para que cualquier usuario, aunque no tenga conocimientos técnicos o estadísticos, pueda entender y navegar por la aplicación de forma intuitiva.

### RNF2

Un nuevo usuario sin experiencia previa debe poder entender el funcionamiento básico del sistema gracias a la página de inicio explicativa y a un diseño claro.

### **RNF3**

El sistema debe manejar correctamente errores por falta de datos, evitando bloques o gráficas fallidas.

### **RNF4**

Los gráficos deben estar optimizados para poder cargar adecuadamente sin congelar la interfaz de usuario.

### **RNF5**

Un usuario podrá consultar el glosario en todo momento para descubrir y comprender de manera clara cualquier concepto de la aplicación que no entienda en primera instancia.

## **4.5. Requisitos de información**

### **RI1**

La información será clasificada en seis grandes bloques: resultados, tiros, goles a favor, goles en contra, tiros por jugador y tiros en contra. Esta información será accesible desde la interfaz del sistema.

### **RI2**

El sistema debe generar o mostrar textos explicativos relacionados con los datos representados en los gráficos, incluyendo conclusiones básicas que ayuden al usuario a entender lo que está viendo. Estos textos deberán ser simples y sencillos de entender para cualquier usuario.

### **RI3**

El sistema no gestionará datos personales de usuarios, ya que no requiere login ni registro, asegurando así el cumplimiento básico de protección de datos (RGPD).

## **4.6. Reglas de negocio**

### **RN1**

La WebApp es de acceso libre y no requiere ningún tipo de inicio de sesión ni de registro. Por tanto, no se gestionan políticas de autenticación ni almacenamiento de información privada, en cumplimiento del RGPD.

### **RN2**

Solo se mostrarán visualizaciones generadas a partir de datos verificados y provenientes de fuentes confiables. No se admitirán valores calculados sin justificación estadística ni interpretaciones no respaldadas.

### **RN3**

Toda conclusión escrita o interpretación debe emplear un lenguaje sencillo, evitando tecnicismos estadísticos, para que el aficionado medio pueda entender la información sin necesidad de experiencia previa o conocimientos en el campo de la estadística.

#### **RN4**

La aplicación solo debe mostrar estadísticas de competiciones oficiales (Liga, Copa del Rey, etc.). No se incluirán partidos amistosos ni de de pretemporada para mantener coherencia con las fuentes oficiales de datos.





## 5. Análisis del Sistema

---

Este capítulo cubre el análisis del sistema de información a desarrollar, haciendo uso del lenguaje de modelado UML.

### 5.1. Modelo Conceptual

Aquí podemos ver el modelo conceptual 5.1 de nuestro proyecto:

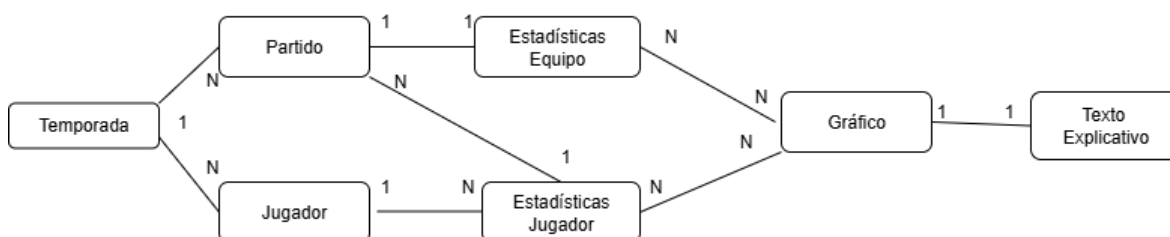


Figura 5.1: Modelo Conceptual

En él, podemos observar cómo salen varios gráficos de las diversas estadísticas del equipo y de los jugadores. Estos conjuntos de estadísticas son tomados de cada partido y de cada jugador, respectivamente. Además, cada uno de estos gráficos irá acompañado de un texto explicativo.

También, podemos observar que se tomarán en cuenta los datos de jugadores y partidos de una sola temporada.

### 5.2. Modelo de Casos de Uso

#### 5.2.1. Actor Principal

El actor principal será el **Usuario** visitante de la WebApp, el cual no necesita estar registrado.

#### 5.2.2. Casos de Uso

- CU1 – Navegar por las diferentes categorías de estadísticas

**Actor:** Usuario

**Precondiciones:** El usuario ha accedido a la WebApp.

**Postcondiciones:** El sistema muestra la categoría seleccionada (partidos, goles, tiros...).

**Escenario principal:**

- 1.El usuario accede a la página de análisis o de predicción.
- 2.Selecciona una categoría desde el menú (por ejemplo, “Tiros”).
- 3.El sistema carga los gráficos correspondientes a esa categoría.

■ **CU2 – Visualizar gráficos**

**Actor:** Usuario.

**Precondiciones:** El usuario ha accedido a una categoría válida.

**Postcondiciones:** El usuario ve gráficos interactivos relacionados con la categoría.

**Escenario Principal**

1. El usuario selecciona una categoría.
2. El sistema muestra uno o más gráficos interactivos relacionados.
3. El usuario puede interactuar con el gráfico (zoom, descarga,etc).
4. El usuario puede elegir otra temporada de otro año para cargar esos datos y visualizar la gráfica correspondiente.

■ **CU3 - Leer explicaciones y conclusiones de los gráficos**

**Actor:** Usuario.

**Precondiciones:** El gráfico ha sido generado correctamente.

**Postcondiciones:** El usuario ve y entiende una breve explicación textual bajo el gráfico.

**Escenario Principal:**

1. El usuario visualiza un gráfico.
2. Debajo del gráfico, el sistema muestra un texto interpretativo generado manual o automáticamente.
- 3.El usuario lee la explicación para entender mejor la información mostrada.

■ **CU4 – Consultar introducción sobre Big Data y fútbol**

**Actor:** Usuario.

**Precondiciones:** El usuario accede a la página de inicio.

**Postcondiciones:** El sistema muestra el texto introductorio.

**Escenario Principal:**

1. El usuario abre la web.

2. El sistema le presenta una breve introducción sobre qué es Big Data, cómo se aplica al fútbol y qué ofrece esta WebApp.

■ **CU5 – Consultar glosario**

**Actor:** Usuario

**Precondiciones:** El usuario no entiende un concepto.

**Postcondiciones:** El sistema muestra el glosario.

**Escenario Principal:**

1. El usuario se encuentra con un concepto que no entiende.
2. El usuario accede al glosario.
3. El sistema le presenta un glosario de términos dónde podrá buscar y consultar aquel que no entiende.

■ **CU6 – Seleccionar nuevo equipo**

**Actor:** Usuario

**Precondiciones:** El usuario desea cambiar de equipo para analizar y predecir.

**Postcondiciones:** El sistema se actualiza con los datos del nuevo equipo.

**Escenario Principal:**

1. El usuario desea cambiar de equipo.
2. El usuario accede al botón “cambiar de equipo”.
3. El sistema le presenta un menú desplegable con varios equipos a elegir.
4. El usuario elige un equipo y confirma su elección. Tras esto cierra la aplicación y vuelve a entrar.
5. El sistema muestra todas las secciones con los datos actualizados.



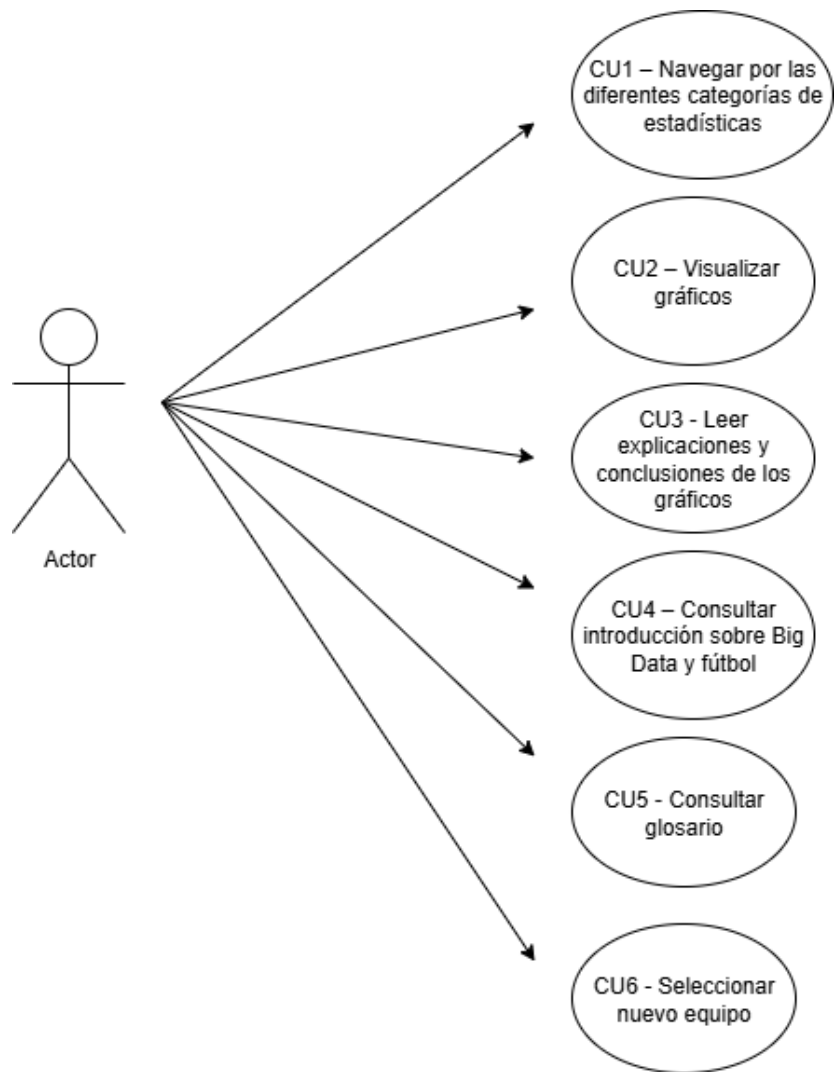


Figura 5.2: Modelo Casos de Uso

### 5.3. Modelo de Interfaz de Usuario

En esta sección podemos ver los mockups de baja fidelidad de las distintas páginas de la aplicación: inicio [5.3](#) , análisis [5.4](#), predicción [5.5](#) y glosario. [5.6](#)



Figura 5.3: Mockup página de inicio

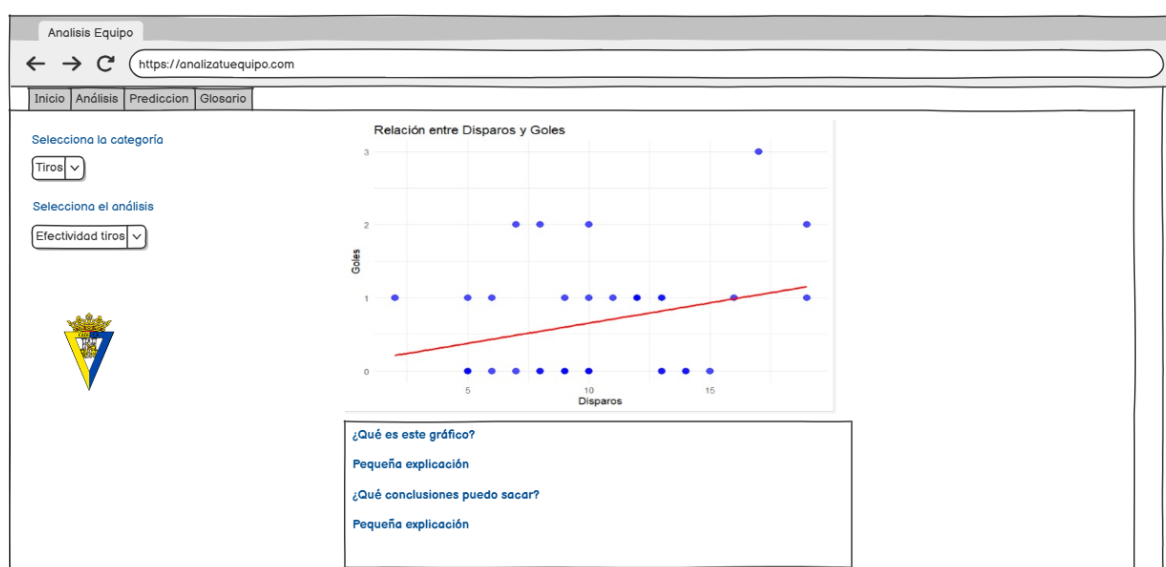


Figura 5.4: Mockup página de análisis

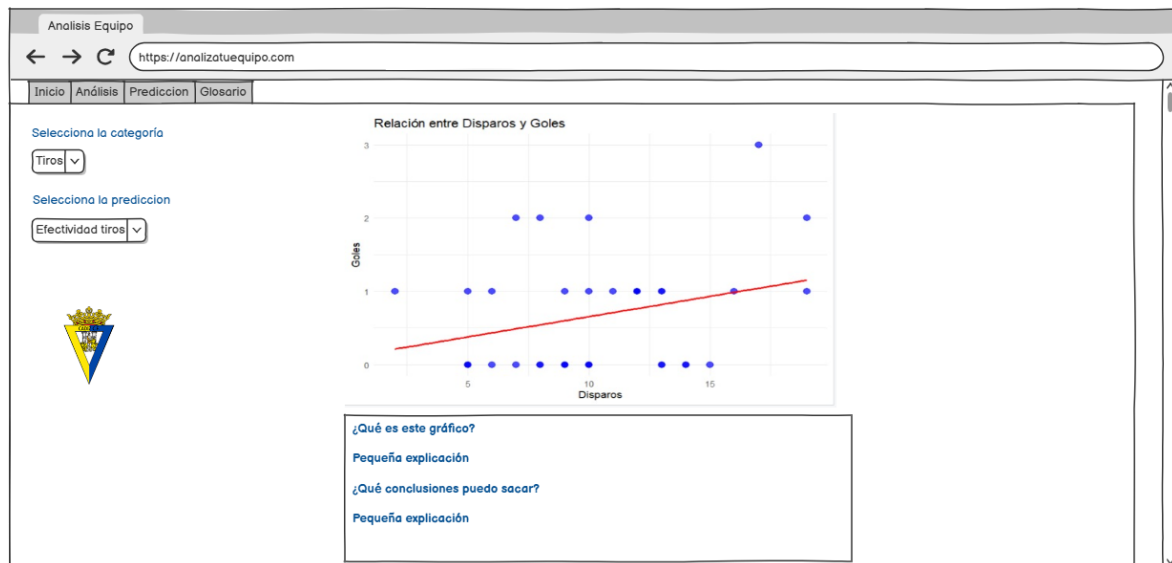


Figura 5.5: Mockup página de predicción



Figura 5.6: Mockup página de glosario





## 6. Diseño del Sistema

---

En este capítulo se recoge el diseño de la arquitectura del sistema informático, el diseño físico de datos y el diseño detallado de la interfaz de usuario.

### 6.1. Arquitectura del Sistema

En esta sección se define la arquitectura general del sistema, especificando la infraestructura tecnológica necesaria para dar soporte al software y la estructura de los componentes que lo forman. Nos centraremos, en este caso, en la arquitectura lógica o de desarrollo del software, dejando para un capítulo posterior la arquitectura física o de despliegue.

#### 6.1.1. Diseño de alto nivel

Para describir la arquitectura del sistema, se ha seguido el modelo C4, el cual nos permite representar visualmente la arquitectura del software en distintos niveles de abstracción. A continuación, se presentan los tres niveles más relevantes para este proyecto:

- **Diagrama de contexto:**

En este diagrama [6.1](#) podemos ver cómo el sistema interactúa con su entorno:

- **Usuario final:** Representa a cualquier persona interesada en consultar estadísticas de fútbol de manera visual e intuitiva. Esta persona puede ser un analista o aficionado al análisis de datos o un aficionado de fútbol promedio que no cuente con conocimientos previos en el campo de la estadística y/o el análisis de datos.
- **Sistema o WebApp de análisis de datos (el sistema):** permite acceder, visualizar e interpretar datos de rendimiento de un equipo de fútbol
- **Fuente externa de datos:** sitio web desde el cual se extraen los datos estadísticos, ya sea mediante descarga directa o scraping. En nuestro caso será principalmente la web FBRef y los datos se descargarán de manera automática mediante varios scripts (uno para cada categoría) y un script maestro que se encargue de llamar a cada uno de estos scripts indicando el año y el equipo del que se quieren obtener los datos.

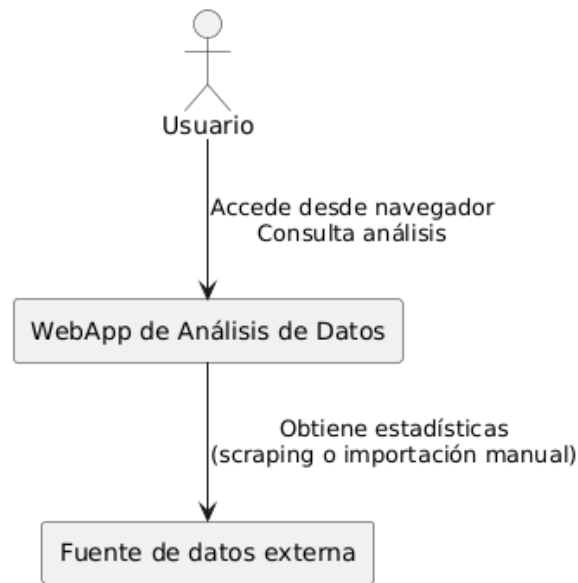


Figura 6.1: Diagrama de contexto

#### ■ Diagrama de contenedores:

Este diagrama 6.2 se enfoca en describir los diferentes contenedores que componen el sistema y cómo interactúan entre ellos.

- **Frontend (UI Shiny):** Aplicación web interactiva desarrollada en R con el framework Shiny. Permite a los usuarios navegar, visualizar gráficos y leer interpretaciones.
- **Servidor Shiny (lógica de servidor):** Se encarga de procesar los datos, generar las visualizaciones y preparar los textos explicativos.
- **Contenedor de datos (almacenamiento local o base de datos):** Los datos pueden estar almacenados en archivos .csv o en una base de datos ligera, y se actualizan manualmente o mediante scripts.

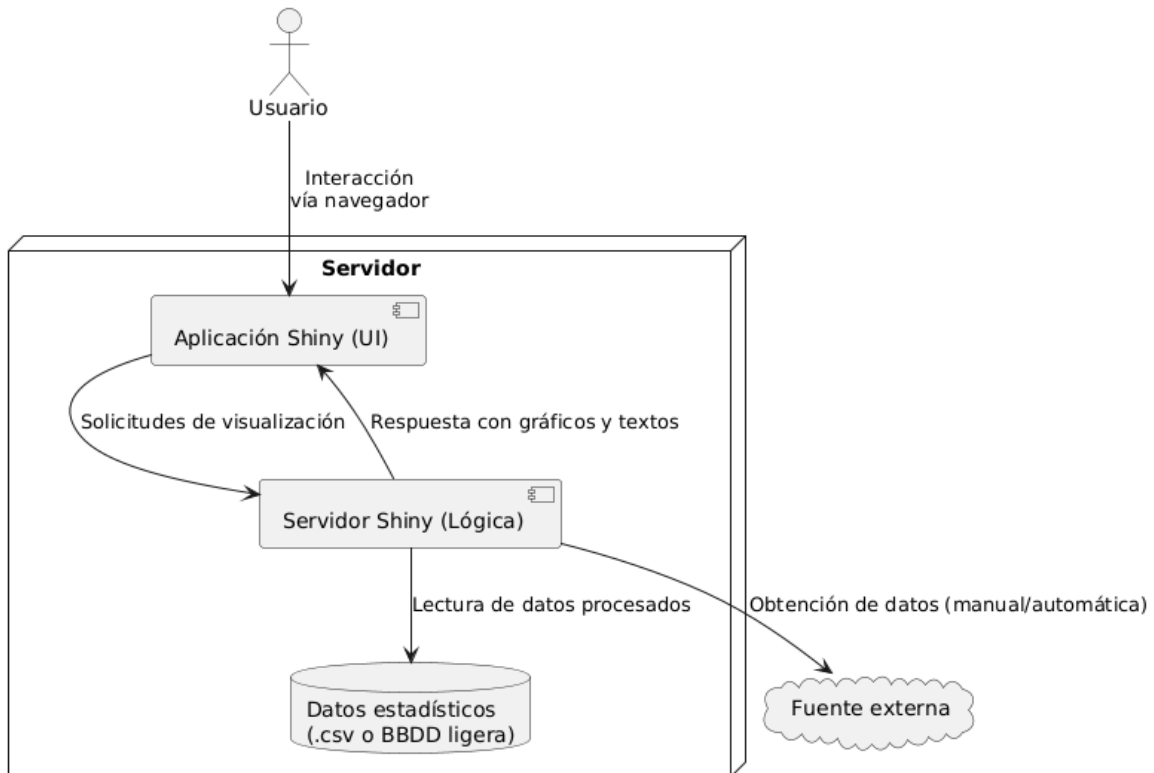


Figura 6.2: Diagrama de Contenedores

#### ■ Diagrama de componentes:

Este diagrama 6.3 detalla los principales módulos lógicos de la aplicación y su responsabilidad:

- Módulo de carga y preprocesado de datos: importa y limpia los datos de entrada.
- Módulo de visualización: genera los distintos tipos de gráficos interactivos.
- Módulo de explicación: asocia a cada gráfico una interpretación escrita en lenguaje no técnico.
- Módulo de navegación: gestiona la estructura de menús y acceso a las diferentes secciones.
- Módulo de introducción: presenta una página inicial con explicación sobre Big Data y su aplicación al fútbol.



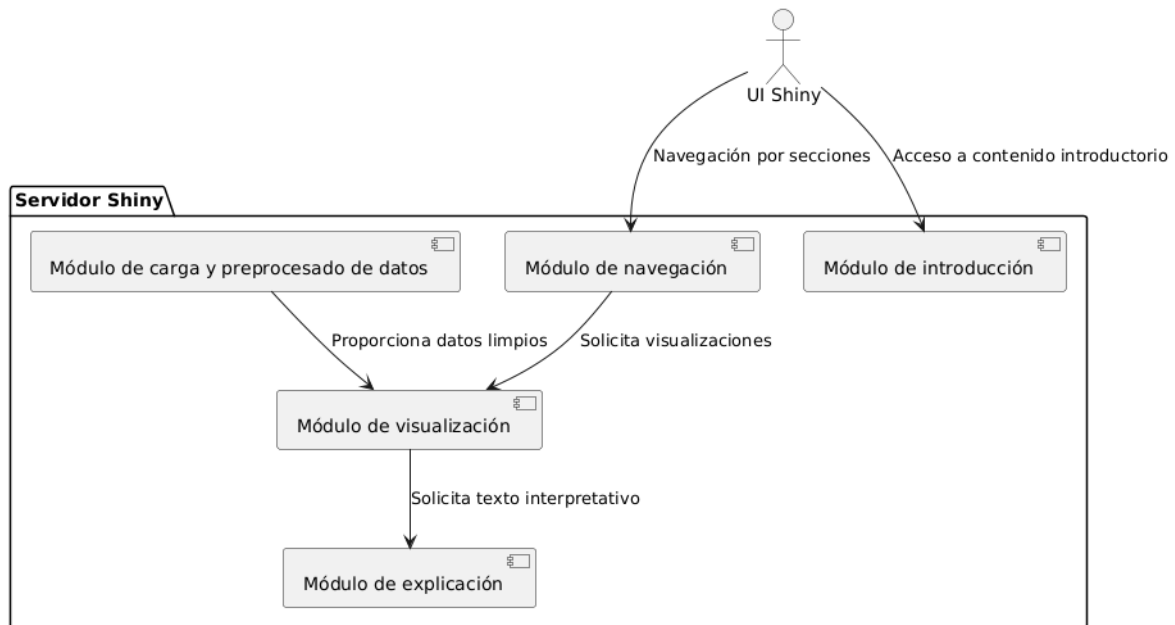


Figura 6.3: Diagrama de Componentes

## 6.2. Diseño Físico de Datos

Dado que la WebApp se está desarrollando en R con Shiny y no requiere grandes volúmenes de almacenamiento, se ha optado por una solución ligera basada en archivos .csv importados de manera automática mediante una serie de scripts desde fuentes externas como FBRef. Igualmente, existe la posibilidad de utilizar una base de datos ligera para mejorar la persistencia y gestión de datos.

### 6.2.1. Tablas de datos

Como se observa en el modelo conceptual, de cada temporada obtenemos las estadísticas de los jugadores. Además, de cada partido de dicha temporada, obtenemos las estadísticas del equipo. Dichas estadísticas se dividen en varias categorías, para las cuales se han usado las siguientes tablas:

- **Tabla de Partidos**

<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
id_partido	Integer	Identificador único del partido
id_equipo	Integer	Identificador único del equipo
Fecha	Date	Fecha del partido disputado
Jornada	Text	Jornada de liga del partido disputado
Local/Visitante	Text	Indicador de si el equipo era Local o Visitante
resultado	Text	Resultado global del partido (ej. 2-1)
goles_favor	Integer	Goles anotados por el equipo
goles_contra	Integer	Goles recibidos por el equipo
rival	Text	Nombre del equipo rival
xG	Double	Goles esperados para anotar
xGA	Double	Goles esperados para recibir
Posesión	Integer	Porcentaje de posesión del equipo
Capitán	Text	Capitán del equipo
Formación	Text	Formación del equipo
Formación rival	Text	Formación del rival

Tabla 6.1

*Tabla de Partidos*

## ■ Tabla de Tiros

<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
id_partido	Integer	Identificador único del partido
Equipo_atacante	Bool	Booleano que sirve como clave para identificar si los tiros son del equipo o de los rivales (para el equipo serían tiros en contra)
Fecha	Date	Fecha del partido disputado
Jornada	Text	Jornada de liga del partido disputado
Local/Visitante	Text	Indicador de si el equipo era Local o Visitante
resultado	Text	Resultado global del partido (ej. 2-1)
goles_favor	Integer	Goles anotados por el equipo
goles_contra	Integer	Goles recibidos por el equipo
Rival	Text	Nombre del equipo rival
Goles marcados	Integer	Goles marcados por el equipo
Disparos	Integer	Disparos realizados por el equipo
Disparos a puerta	Integer	Disparos a puerta realizados por el equipo
Porcentaje Disparos a Puerta	Integer	Porcentaje de disparos que han ido a puerta
Goles/Disparos	Double	Division de goles entre disparos realizados
Goles/Disparos a puerta	Double	Division de goles entre disparos a puerta realizados
Distancia	Double	Distancia media de los disparos
xG	Double	Goles esperados para anotar
xGnP	Double	Goles esperados para anotar sin contar penaltis
G - xG	Double	Goles menos goles esperados para anotar
G - xGnP	Double	Goles menos goles esperados para anotar sin contar penaltis

Tabla 6.2

*Tabla de Tiros*

#### ■ Tabla Datos Jugador

<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
id_jugador	Integer	Identificador único del jugador
Nombre	Text	Nombre del jugador
País	Text	Nacionalidad del jugador
Posición	Text	Posición/Posiciones en las que juega
Edad	Integer	Edad del jugador

Tabla 6.3

*Tabla de Datos Jugador*

#### ■ Tabla Tiros Jugador

<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
id_jugador	Integer	Identificador único del jugador para enlazar con su tabla de datos
Goles	Integer	Goles anotados por el jugador
Disparos	Integer	Disparos realizados por el jugador
Disparos a puerta	Integer	Disparos a puerta realizados por el jugador
Porcentaje Disparos a Puerta	Integer	Porcentaje de disparos que han ido a puerta
Disparos cada 90 min	Double	Disparos realizados por el jugador cada 90 minutos
Disparos a puerta cada 90 min	Integer	Disparos a puerta realizados por el jugador cada 90 minutos
Goles/Disparos	Double	Division de goles entre disparos realizados
Goles/Disparos a puerta	Double	Division de goles entre disparos a puerta realizados
Distancia	Double	Distancia media de los disparos
xG	Double	Goles esperados para anotar

Tabla 6.4

*Tabla de Tiros Jugador*

#### ■ Tabla Goles

<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
id_gol	Integer	Identificador único del gol
Equipo_atacante	Bool	Booleano que sirve como clave para identificar si los goles son del equipo o de los rivales (para el equipo serían tiros en contra)
Sede	Text	Indicador de si el equipo era Local o Visitante
Anotador	Text	Jugador responsable del gol
Parte del cuerpo	Text	Parte del cuerpo con la que se anotó el gol
Distancia	Double	Distancia desde la que se anotó el gol
Minuto	Integer	Minuto en el que se anotó el gol
Marcador	Text	Marcador final del encuentro

Tabla 6.5

*Tabla de Goles*

## 6.3. Diseño de la Interfaz de Usuario

### 6.3.1. Principios de diseño

Para el diseño de la interfaz de usuario se han seguido los siguientes principios generales:

- Claridad y simplicidad: Se prioriza la presentación clara de información, evitando la sobrecarga visual.

- **Consistencia:** Los elementos visuales (colores, botones, menús) se repiten y mantienen un diseño coherente en todas las secciones.
- **Accesibilidad y legibilidad:** Uso de colores contrastantes y tipografía legible, adaptada a pantallas de distintos tamaños.
- **Navegación intuitiva:** El menú superior permite acceder fácilmente a las secciones principales (Inicio, Análisis, Actualización de datos, Cambio de equipo).

### 6.3.2. Estilo Visual

En cuanto al estilo visual, la aplicación contará con los siguientes elementos:

- **Colores corporativos:** En este ejemplo se emplean los colores oficiales del Cádiz CF (amarillo y azul) como base del diseño. Si el usuario desea cambiar de equipo y selecciona uno que no sea el Cádiz CF, estos colores se actualizarán de manera automática por unos más neutros.
- **Tipografía:** Se utiliza una fuente sans-serif moderna para mejorar la legibilidad.
- **Iconografía:** Se incluyen iconos minimalistas e intuitivos para acciones como seleccionar o interactuar con los gráficos.

### 6.3.3. Componentes visuales

Los componentes visuales de la aplicación son los siguientes:

- **Menú de navegación superior:** Siempre visible y accesible, con indicadores visuales del apartado activo.
- **Caja de texto informativa:** Se usa en la página de inicio para introducir al usuario en el contenido, el concepto de Big Data y su utilidad. También se usará en la página de glosario para cada una de las definiciones.
- **Desplegables para elegir el tipo de análisis.**
- **Gráficos interactivos** que responden a la selección del usuario.
- **Explicación contextual** debajo del gráfico, en lenguaje accesible.
- **Botones de llamada a la acción (CTA):** Incluido en la pantalla de inicio (“Empieza el análisis”, “Ve al Glosario”, “Cambia de Equipo”).

### 6.3.4. Prototipos

Los prototipos que se presentan a continuación permiten ver de forma estimada el aspecto que tendrá la web con sus datos de ejemplo del Cádiz CF. Este aspecto cambiará cuando se decida elegir otro equipo a analizar, pasando la web a tener colores más neutros.



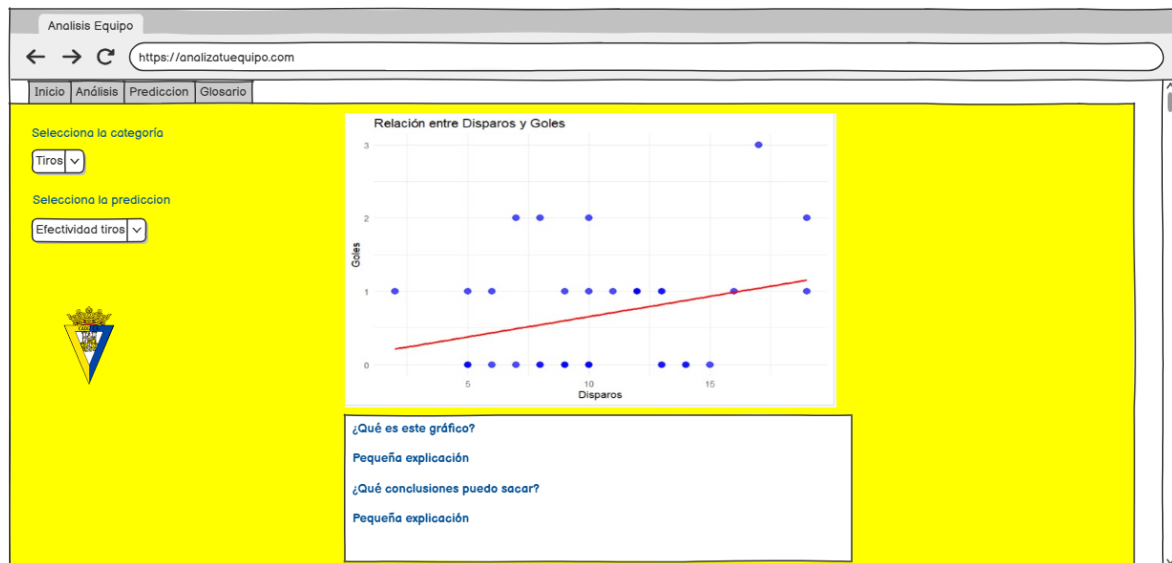


Figura 6.6: Prototipo de la página de predicción



Figura 6.7: Prototipo de la página de glosario







## 7. Construcción del Sistema

---

Este capítulo trata sobre todos los aspectos relacionados con la implementación del sistema en código, haciendo uso de un determinado entorno tecnológico.

### 7.1. Entorno de Construcción

Como he comentado anteriormente, este proyecto está basado en el lenguaje de programación R y en su IDE R Studio. Además, inicialmente, se usó la herramienta OpenRefine para la limpieza y procesado de datos.

### 7.2. Código Fuente

En el siguiente enlace se encuentra el repositorio de Git con todo el código fuente de la aplicación y todos los archivos relevantes: [TFG-UNAI-CADIZ](#).

Además del código fuente principal de la WebApp, implementado en un archivo llamado App.R , se han programado una serie de scripts (uno por cada categoría) con la función de descargar datos y estadísticas de manera automatizada, así como de limpiar y procesar estos datos.

También se ha programado un script maestro cuya función consiste en llamar a cada uno de estos scripts pasándole dos argumentos: el código del equipo del cual se quieren extraer los datos y los años de las temporadas a analizar. Estas llamadas al script se realizarán de manera automática teniendo en cuenta el equipo que elija el usuario. Por ejemplo, la siguiente llamada descargaría los datos del Cadiz cf de las temporadas 2022-2023 y 2023-2024:

**Rscript descarga\_datos\_anno.R ee7c297c 2022 2023**

El script maestro tiene la siguiente forma:

```
args <- commandArgs(trailingOnly = TRUE)
if (length(args) < 2) {
  stop("Debes proporcionar el código del equipo y al menos un año. -
  Ejemplo: Rscript descargar_datos.R ee7c297c 2023")
}
codigo_equipo <- args[1]
años <- as.integer(args[-1])

scripts <- c("descargaautomaticadatosresultados.R",
             "descargaautomaticadatostiros.R",
             "descargaautomaticadatostirosencontra.R",
             "descargaautomaticadatosgolesafavor.R",
             "descargaautomaticadatostopgolesadores.R",
```

```

"descargaautomaticadatosgolesencontra.R")
for (i in seq_along(annios)) {
  annio <- annios[i]
  cat(paste0("\n-Procesando-datos-para-el-annio-", annio, "-y-equipo-",
    codigo_equipo, "\n"))

  if (i > 1) {
    cat("Pausa-entre-annios-(15-segundos)... \n")
    Sys.sleep(15)
  }

  for (j in seq_along(scripts)) {
    script <- scripts[j]
    cat(paste0("-Ejecutando-", script, "-para-", annio, "\n"))
    source(script, local = TRUE)

    if (j < length(scripts)) {
      tiempo_pausa <- sample(5:10, 1)
      cat(paste0("-Pausa-de-", tiempo_pausa, "-segundos... \n"))
      Sys.sleep(tiempo_pausa)
    }
  }
}

writeLines(codigo_equipo, "equipo_actual.txt")

```

Mediante este scripts se llama a cada uno de los scripts que descargan datos de cada una de las categorías, indicando el código del equipo y las temporadas deseadas como argumentos. Además, se han añadido pausas entre cada script y entre temporadas para evitar colapsar la página FBRef o que nos bloquee por exceso de peticiones en poco tiempo. Al final del script se modifica también un documento de texto equipo\_actual.txt escribiendo el equipo del cual se han descargado los datos.

El código principal de la WebApp “App.R” consiste en dos partes, como cualquier código en R que use el framework shiny:

- **UI:** *User Interface* o interfaz de usuario, controla la apariencia de la app y de sus distintas páginas. Aquí controlaremos el aspecto visual de las páginas Inicio, Análisis, Predicción y Glosario.
- **Server:** Contiene las funciones y toda la lógica para que nuestra aplicación funcione como deseamos. Entre las funciones más importantes encontramos la generación de gráficos según categoría y elemento a analizar, la lógica para cambiar de equipo con su llamada al script necesaria, la generación de gráficos para las predicciones y el manejo de los distintos .csv que contienen todas las estadísticas con su correspondiente combinación en los casos que sea necesario.

## 7.3. Gráficas

En este apartado se recogen las distintas visualizaciones elaboradas a partir de los datos analizados. El objetivo de las gráficas es principalmente ayudar al usuario a ver de manera clara y sencilla qué conclusiones se pueden sacar de cada análisis: carencias, elementos a mejorar, puntos fuertes del equipo, etc.

Las distintas gráficas usadas han sido las siguientes:

### ■ Gráficos de barras

Este tipo de representación resulta adecuado para comparar cantidades discretas o categóricas. Se ha usado en los siguientes análisis:

- **Resultados en Casa vs Fuera y Goles en Casa vs Fuera:** permiten visualizar de manera clara las diferencias en el desempeño del equipo dependiendo de la condición de local o visitante.
- **Goles en contra en Casa vs Fuera:** complementa el análisis anterior, enfocándose en la vulnerabilidad defensiva.
- **Goles con cada parte del cuerpo y Goles en contra con cada parte del cuerpo rival:** ilustran la frecuencia de goles según la zona corporal empleada, lo que ayuda a identificar fortalezas ofensivas y carencias defensivas.

### ■ Histogramas

Los histogramas se utilizan para representar la distribución de variables continuas. Se ha usado en los siguientes análisis:

- **Posesión según Resultado:** refleja cómo se reparte el porcentaje de posesión en función del desenlace del partido.
- **Distribución de goles por minuto y Distribución de goles en contra por minuto:** permiten identificar los intervalos temporales en los que el equipo es más efectivo o más vulnerable.
- **Distancia de los Goles y Distancia de los Goles en contra:** muestran desde qué posiciones del campo se generan y se reciben los goles, aportando información sobre la tipología de las finalizaciones.
- **Distancia media de los tiros en contra:** facilita analizar si los rivales generan peligro desde fuera del área o principalmente en posiciones cercanas a portería.

### ■ Diagramas de dispersión

Los gráficos de dispersión son apropiados para explorar relaciones entre dos variables continuas. En algunos casos se añade una línea de regresión o referencia para facilitar la interpretación. Se ha usado en los siguientes análisis:

- **Relación entre Posesión y Goles Marcados y Relación entre Posesión y Goles en Contra:** permiten evaluar si existe correlación entre dominar la posesión y la capacidad ofensiva/defensiva.

- **Comparación entre xG y Goles Marcados:** compara los goles esperados con los realmente convertidos, evidenciando la eficacia o ineficacia en la definición.
- **Comparación de xG con y sin penaltis:** aísla el impacto de los penaltis para valorar la generación de ocasiones en juego abierto.
- **Relación entre Disparos y Goles:** mide la eficiencia de los tiros, comparando volumen y efectividad.
- **Relación entre Disparos Recibidos y Goles en Contra:** analiza la capacidad defensiva de limitar el peligro del rival más allá del número de disparos concedidos.
- **Comparación xG concedido vs Goles recibidos:** muestra si los goles encajados se corresponden con la calidad de las ocasiones concedidas.
- **Goles por disparo vs Goles por disparo a puerta:** permite analizar la precisión ofensiva general frente a la efectividad en disparos dirigidos a portería.
- **Relación entre Disparos cada 90min y Goles:** cuantifica la frecuencia de disparo como predictor de la producción goleadora.
- **Comparación de Goles vs xG (por equipo/jugador):** revela qué equipos o jugadores superan las expectativas y cuáles rinden por debajo de lo esperado.
- **Efectividad por edad:** analiza cómo varía la capacidad de finalización en función de la edad del jugador, observando posibles patrones evolutivos.

## 7.4. Scripts de Base de datos

Para la base de datos, se utiliza una carpeta en la que se almacenan todos los datos correspondientes de cada temporada divididos en las 6 categorías principales; un archivo por cada temporada, equipo y año: Resultados, Tiros, Tiros en Contra, Tiros de Jugador, Goles a Favor y Goles en Contra.

La nomenclatura seguida para estos archivos es la siguiente “Nombre equipo” + “Categoría” + “Año”.csv. Es decir, el archivo de la categoría Resultados del Cádiz CF de la temporada 2023-2024 tendrá el nombre: “CádizResultados2023.csv”.

Estos datos se encuentran en archivos .csv y están procesados y “limpios” para que cumplan el formato correcto para el funcionamiento de la app. Estos datos son descargados de manera automática mediante el script maestro previamente comentado.





## 8. Pruebas del Sistema

---

A continuación, se presentan los diferentes tipos de pruebas que se han llevado a cabo y los resultados obtenidos.

### 8.1. Estrategia

Las pruebas del sistema se han ido realizando y se realizarán de forma manual con cada actualización e implementación de un nuevo componente o modificación de uno ya existente. De esta manera, se consigue detectar errores de manera temprana y asegurarnos de la correcta integración de cada componente en la aplicación.

De igual manera, se comprobará que todos los datos se descarguen correctamente y se muestren las gráficas correspondientes de manera actualizada con cada uno de ellos, ya que, si los datos contienen errores o no se descargan adecuadamente, la aplicación no cumpliría su objetivo principal.

Una vez finalizada la aplicación, se realizará una prueba exhaustiva de todas las funcionalidades del sistema. El objetivo será comprobar que todo funciona correctamente y no quedan errores presentes en ninguna funcionalidad.

### 8.2. Pruebas Unitarias

Se han realizado pruebas unitarias con el objetivo de garantizar la fiabilidad y correcto funcionamiento de cada componente implementado. Se han probado todas las funcionalidades y características de cada componente para asegurar su correcto funcionamiento, desde casos de uso básicos hasta escenarios más complejos.

El procedimiento seguido para cada prueba fue el siguiente:

1. **Preparación del escenario de prueba:** carga de datos de ejemplo o configuración inicial del sistema.
2. **Ejecución del componente:** interacción directa con la interfaz o invocación de la función correspondiente.
3. **Observación del resultado:** análisis visual o comprobación de valores devueltos.
4. **Comparación con el resultado esperado:** validación de que la salida coincide con lo especificado.

Los escenarios a probar fueron los siguientes:

- Navegación entre las diferentes categorías de estadísticas.



- Navegación entre las diferentes categorías de predicciones.
- Visualización de gráficos interactivos y actualización tras la carga de nuevos datos.
- Generación automática de explicaciones, textos orientativos y conclusiones.
- Comprobación del comportamiento ante datos incompletos o inconsistentes.
- Actualización de temporadas a analizar, eligiendo distintas temporadas y combinándolas. Esta prueba se realiza tanto en el apartado de Análisis como en el de Predicción.

Las pruebas dieron como resultado un cumplimiento de lo esperado, tanto en situaciones simples como en escenarios con mayor complejidad de datos.

### 8.3. Pruebas de Integración

Las pruebas de integración se realizaron de forma manual, con el propósito de verificar la correcta interacción entre los distintos componentes del sistema, asegurando que la comunicación y el flujo de datos se desarrollan según lo especificado.

En este caso, cada prueba consistió en ejecutar escenarios que implican más de un módulo o funcionalidad trabajando de manera conjunta, observando que el resultado final fuera coherente y sin errores.

Los escenarios a probar fueron los siguientes:

- Carga de datos y actualización automática de gráficos asociados.
- Sincronización entre gráficos y textos explicativos.
- Navegación y persistencia de datos al cambiar de categoría.
- Coherencia en la presentación de estadísticas tras modificaciones de datos.

Las pruebas confirmaron que los módulos del sistema interactúan de manera correcta y que la información se actualiza sin ningún tipo de problema inesperado.

### 8.4. Pruebas de Sistema

En esta sección, se realizarán pruebas del sistema con el objetivo de verificar que la plataforma cumple con todos los requisitos funcionales y no funcionales establecidos.

#### 8.4.1. Pruebas Funcionales

Se realizaron pruebas manuales para verificar que el sistema cumple con todos los requisitos funcionales previamente definidos. Cada prueba consistió en simular el comportamiento de un usuario y comprobar que la funcionalidad se ejecuta de acuerdo con lo especificado.

### Procedimiento general

1. Identificación del requisito funcional a evaluar.
2. Ejecución de las acciones necesarias para ponerlo a prueba.
3. Observación del comportamiento del sistema.
4. Comparación del resultado obtenido con el esperado.
5. Observaciones y conclusiones.

RF	Descripción	Método de prueba	Resultado esperado	Resultado obtenido	Cumple (Sí/No)
RF1	Navegación entre categorías de estadísticas	Recorrer todas las categorías desde el menú principal	El sistema permite cambiar de categoría sin errores y de forma inmediata	Funciona según lo esperado	Sí
RF2	Visualización de gráficos interactivos	Interactuar con todos los gráficos (hover, zoom, etc)	Los gráficos reaccionan a la interacción sin fallos	Se observa respuesta correcta en todos los gráficos	Sí
RF3	Explicación bajo cada gráfico	Comprobar cada gráfico y su texto asociado	Bajo cada gráfico aparece una explicación coherente	Todos los gráficos cuentan con su explicación	Sí
RF4	Lenguaje no técnico en explicaciones	Revisar el contenido textual de las explicaciones	El lenguaje es claro y sin tecnicismos	Se cumple en todas las explicaciones	Sí
RF5	Actualización con nuevos datos	Subir un nuevo archivo de datos y verificar cambios en gráficos	Los gráficos se actualizan correctamente	Los cambios se reflejan correctamente	Sí
RF6	Página de inicio con introducción	Abrir la página principal	Se muestra introducción con definición de Big Data en lenguaje claro	El contenido cumple lo especificado	Sí

Tabla 8.1

#### *Verificación de requisitos funcionales*

### 8.4.2. Pruebas No Funcionales

Estas pruebas pretenden comprobar el funcionamiento del sistema, con respecto a los requisitos no funcionales identificados previamente.

Las pruebas se han realizado de forma manual, simulando distintos escenarios de uso por parte de usuarios con diferentes niveles de experiencia técnica.

RNF	Descripción	Método de prueba	Resultado esperado	Resultado obtenido	Cumple (Sí/No)
RNF1	Interfaz intuitiva para cualquier usuario, sin conocimientos técnicos o estadísticos	Navegar por todas las secciones con un usuario sin experiencia	El usuario puede interactuar y encontrar funciones sin ayuda externa	El usuario navegó correctamente sin confusiones	Sí
RNF2	Comprensión básica del sistema gracias a la página de inicio y diseño claro	Probar con un usuario nuevo, pidiéndole que describa la aplicación tras leer la página de inicio	El usuario comprende el objetivo y funciones principales	El usuario entendió la funcionalidad y navegación	Sí
RNF3	Manejo de errores por falta de datos	Forzar la ausencia de datos en una sección	El sistema muestra mensajes claros sin fallos ni bloqueos	El sistema mostró mensajes de error adecuados y continuó funcionando	Sí
RNF4	Optimización de gráficos para carga fluida	Cargar gráficos con datasets de distinto tamaño	Los gráficos se cargan rápidamente sin congelar la interfaz	Carga fluida en todos los casos probados	Sí
RNF5	Consulta de glosario para aclarar conceptos	Buscar términos desconocidos en el glosario	El glosario explica los conceptos de forma comprensible	Todos los términos buscados fueron entendidos por el usuario	Sí

Tabla 8.2

*Verificación de requisitos no funcionales*

## 8.5. Pruebas de Aceptación

El objetivo de estas pruebas es demostrar que el producto está listo para el paso a producción; para ello, se toma en cuenta la opinión del cliente final, es decir, un usuario que acceda a nuestra página.

ID	Funcionalidad probada	Método de prueba	Resultado esperado	Resultado obtenido	Aceptado (Sí/No)
PA1	Navegación entre categorías de estadísticas	Cliente navega por la interfaz	Acceso fluido a todas las secciones sin errores	Navegación correcta, sin incidencias	Sí
PA2	Visualización de gráficos interactivos	Cliente abre y manipula gráficos en distintas secciones	Los gráficos se muestran y responden a interacciones	Gráficos visibles e interactivos según lo previsto	Sí
PA3	Textos explicativos claros y comprensibles	Cliente revisa las descripciones bajo los gráficos	Textos entendibles sin tecnicismos complejos	Textos comprendidos por el cliente sin ayuda externa	Sí
PA4	Actualización de datos y refresco de gráficos	Cliente añade datos nuevos y observa el cambio en las visualizaciones	Gráficos actualizados inmediatamente	Actualización en tiempo real sin errores	Sí
PA5	Página de inicio con introducción y explicación de Big Data	Cliente lee y evalúa la introducción	Texto claro y acorde al público objetivo	Cliente confirmó comprensión del contenido	Sí

Tabla 8.3

*Resultados de pruebas de aceptación*

Para cada una de estas pruebas se ha tomado en cuenta la opinión de distintos tipos de usuarios que han accedido a ayudarme a probar la web:

- **Usuario 1:** La primera usuaria ha sido Ainara, que cuenta con pocos conocimientos tanto informáticos como de fútbol. Por lo tanto, mediante su navegación por la web se ha evaluado cómo de intuitiva y fácil de entender es la interfaz. A pesar de tener algunas pequeñas dificultades al inicio debido a su falta de conocimientos, la usuaria ha podido navegar por la web y entender todo el contenido que se le presentaba aunque ha empleado más tiempo que los otros dos usuarios en navegar por la web.
- **Usuario 2:** El segundo usuario ha sido Gorka, un hombre adulto aficionado al fútbol, con conocimientos informáticos pero poco conocimiento sobre Big Data y análisis de datos. El usuario ha podido navegar por la web sin dificultades y entender todos los gráficos y análisis, con la ayuda del glosario para la comprobación de algunos términos que no sabía que significaban.
- **Usuario 3:** El tercer y último usuario ha sido Ángel, un hombre con conocimientos en Big Data, estadística y análisis de datos. Como era de esperar, este usuario no ha tenido ningún problema navegando por la web y entendiendo su contenido. Además, ha valorado positivamente la aplicación web y su utilidad.



## 9. Despliegue del Sistema

---

Este capítulo recoge la arquitectura física planteada para el sistema, las instrucciones para su despliegue y las instrucciones para la operación y mantenimiento del nivel de servicio.

### 9.1. Arquitectura Física

En este apartado se describen los componentes hardware y software que conforman el servidor, así como este se comunica con cualquier usuario de la plataforma.

#### 9.1.1. Elementos hardware

Para garantizar un rendimiento óptimo, el sistema se desplegará en un entorno de servidor ligero. Los requisitos mínimos recomendados para el servidor son:

- **CPU:** Procesador x86\_64 con al menos 2 núcleos.
- **RAM:** 4 GB de memoria (mínimo) / 8 GB (recomendado).
- **Almacenamiento:** 10 GB.
- **Red:** Conectividad estable a internet y acceso a puertos HTTP/HTTPS.

#### 9.1.2. Elementos software

Los componentes software del servidor serán los siguientes:

- **Sistema operativo:** Windows 10, Ubuntu Server 22.04 LTS o similar.
- **Lenguaje de programación:** R (versión 4.2 o mayor).
- **Framework Web:** Shiny para desarrollo de la aplicación web.
- **Servidor Web:** Shiny Server Open Source.
- **Dependencias de R:** Shiny, tidyverse, plotly, DT, entre otros.

#### 9.1.3. Comunicación con el usuario

La WebApp será accesible a través de un navegador moderno. El usuario interactuará mediante peticiones HTTP a la interfaz generada por Shiny, que ejecuta dinámicamente código R en el servidor y genera respuestas visuales (gráficos, tablas, textos).

## 9.2. Instrucciones de despliegue

A continuación, se describen las instrucciones de instalación del sistema sobre la infraestructura física descrita anteriormente.

### 9.2.1. Requisitos previos

Aquí se indican los prerrequisitos que deben estar cumplidos antes de instalar la aplicación:

- **Sistema operativo instalado y actualizado:** Ubuntu Server o Windows 10
- **Conectividad a internet estable**
- **R instalado (versión 4.2 o superior)**
- **Paquetes de R actualizados**
- **Navegador web moderno**

### 9.2.2. Inventario de componentes

Aquí se listan los elementos que forman parte del producto final (hardware + software), ya no como recomendación de requisitos, sino como inventario de lo que efectivamente compone el sistema:

#### Hardware

- **Servidor con CPU x86\_64 de 2 núcleos.**
- **4 GB de RAM (mínimo) / 8 GB (óptimo).**
- **10 GB de almacenamiento en disco.**

#### Software

- **Sistema operativo: Windows 10** (instalación base).
- **Lenguaje: R** ( v4.2 o superior)
- **Framework web Shiny.**
- **Paquetes de R incluidos** (instalados en la versión + otros necesarios como bslib, promises, future, rvest y writexl)
- **Navegador cliente compatible** (Chrome, Firefox o Edge en sus últimas versiones.)

## Carpeta final del proyecto

Como entregable del proyecto y acompañando a este PDF que contiene la documentación, se subirá a la plataforma de entrega un fichero Zip con los siguientes elementos:

- **App.R**: fichero escrito en el lenguaje R que contiene todo el código de la aplicación.
- **Carpeta Data**: carpeta que contiene los .csv utilizados, con el formato y nomenclatura descritos en el apartado 7.4 de este documento.
- **Carpeta Scripts**: carpeta que contiene los scripts empleados para la descarga de datos, comentados en el apartado 7.2 de este documento.

### 9.2.3. Procedimientos de instalación

Para el correcto despliegue de la aplicación, proceda según las siguientes instrucciones.

#### 1. Instalación de software base

##### a) Instalar R:

- Descargar desde la página oficial de CRAN: <https://cran.r-project.org>
- Seleccionar Download R for Windows (elegir Ubuntu en caso de usar ese sistema operativo) → base → instalar con la configuración por defecto.

##### b) Instalar Git

c) **Instalar RStudio (opcional, recomendado)** : Descargar desde: <https://posit.co/download/desktop/> e instalar con la configuración por defecto.

#### 2. Clonar el repositorio del proyecto

Abrir Git Bash o Powershell y ejecutar la siguiente instrucción:

```
git clone https://github.com/UnaiZoat/TFG-UNAI-CADIZ
```

#### 3. Ejecución de la aplicación

##### a) Desde R directamente:

- Abrir R desde el menú del sistema.
- Ubicarse en la carpeta del proyecto.
- Ejecutar la app: `runApp(app.R)`.

##### b) Desde RStudio (recomendado):

- Abrir el archivo `app.R`.
- Pulsar Run App en la parte superior derecha y open in browser en caso de querer visualizar la aplicación en nuestro navegador por defecto.



### 9.2.4. Pruebas de implantación

Una vez realizada la clonación y ejecución de la aplicación, bastará con acceder al puerto local en el que se abra la aplicación para comprobar su correcto funcionamiento. (normalmente debe ser `http://127.0.0.1:3838`).

Una vez abierto, podremos comprobar que todas las funciones de la aplicación nos devuelven el resultado esperado y todo se carga correctamente: ver los análisis de las distintas categorías, comprobar las predicciones, cambiar de equipo, etc.

Igualmente, durante la presentación del proyecto y cuando el administrador habilite su acceso, se podrá acceder a la WebApp en el siguiente enlace: [ShinyApps.io/AnalisisEquipo](https://ShinyApps.io/AnalisisEquipo)

## 9.3. Instrucciones para la operación del sistema y mantenimiento del nivel de servicio

Para asegurar la correcta operación de la aplicación y mantener un nivel de servicio estable, se deben seguir los siguientes procedimientos:

### 9.3.1. Operación diaria

- **Verificación de disponibilidad:** comprobar que la aplicación se encuentra accesible en el puerto configurado (por defecto: `http://127.0.0.1:3838`).
- **Supervisión de procesos:** confirmar que el servicio de R/Shiny se está ejecutando sin interrupciones.
- **Comprobación de logs:** revisar los registros de ejecución (archivo `app.log`) en busca de errores o advertencias.

### 9.3.2. Copias de seguridad (backups)

- **Código fuente:** Mantener actualizado el repositorio de Github con una versión estable de la aplicación.
- **Datos:** realizar copias periódicas de los datos actualizados (archivos `.csv`)

### 9.3.3. Descarga de datos

En el caso de que se quiera descargar nuevos datos de nuevos equipos o de nuevas temporadas, simplemente habrá que realizar la llamada al script con el código de equipo deseado y/o el año de la temporada a descargar. Por ejemplo, en el caso de querer descargar los datos del Cádiz CF de la temporada 2019, habrá que realizar la siguiente llamada:

## Rscript descarga\_datos\_anno.R ee7c297c 2019

Los códigos de cada equipo los podremos encontrar en la página Fbref en su link correspondiente. Por ejemplo, en un link cualquiera del Cádiz como este [Fbref Cádiz](#) podemos ver su código “ee7c297c” en la URL.

### 9.3.4. Mantenimiento preventivo

- **Actualización de dependencias:** revisar periódicamente la versión de R y de los paquetes necesarios (shiny, tidyverse, etc.), aplicando actualizaciones controladas.
- **Pruebas tras actualizaciones:** tras cada actualización de R o librerías, realizar las pruebas necesarias para comprobar el correcto funcionamiento de la aplicación y de todas sus funciones.
- **Monitoreo de rendimiento:** comprobar el tiempo de respuesta de la aplicación y el consumo de recursos (CPU, RAM).

### 9.3.5. Seguridad

- **Actualización del sistema operativo:** aplicar parches de seguridad al servidor donde corre la aplicación.
- **Revisión de vulnerabilidades:** verificar que las dependencias no tengan fallos de seguridad reportados, antes y después de actualizarlas.

### 9.3.6. Documentación y solución de posibles incidencias

- **Registro de incidencias:** Documentar cualquier error detectado, la causa y la solución aplicada.
- **Plan de recuperación:** Disponer de un procedimiento para restaurar rápidamente el sistema en caso de fallo (reinstalación desde el repositorio + restauración de backups).

## Parte III

### Epílogo

# 10. Conclusiones

---

En este último capítulo se detallan las lecciones aprendidas tras el desarrollo del presente proyecto y se identifican las posibles oportunidades de mejora sobre el software desarrollado.

## 10.1. Objetivos alcanzados

Este proyecto ha logrado cumplir con éxito los objetivos definidos en la fase inicial. Se ha podido visualizar y analizar el rendimiento del Cádiz en la temporada 2023-2024 ayundándonos de gráficas y sacando conclusiones de estas. Además, se ha ido más allá, añadiendo la opción de analizar temporadas anteriores e incluso predecir la siguiente temporada usando modelos predictivos así como la opción de cambiar de equipo a analizar.

## 10.2. Lecciones aprendidas

Durante el desarrollo de este proyecto, he vivido una experiencia enriquecedora y formativa que me ha permitido adentrarme y aprender valiosas lecciones del mundo del Big Data y el análisis de datos.

Principalmente, he aprendido a manejar mejor el lenguaje R, siendo esta la primera vez que desarrollaba un proyecto de mayor envergadura utilizando este lenguaje. He podido aprender y aprovechar las características que este lenguaje ofrece a la hora de desarrollar gráficas sobre los datos a analizar. Además, usando el framework Shiny he descubierto una manera de desarrollar aplicaciones webs enfocadas en el análisis de datos de manera sencilla y efectiva.

Por otro lado, también he aprendido conceptos sobre el análisis y la predicción de datos. He descubierto y ampliado mi conocimiento sobre conceptos típicos como regresión lineal, histogramas y modelos predictivos entre otros.

Además, he aprendido a organizar mi tiempo y a enfrentarme a los obstáculos que suponen un trabajo más largo y elaborado, como es un proyecto de fin de grado.

## 10.3. Trabajo futuro

Durante el desarrollo del proyecto se han identificado diversas áreas con potencial de mejora, ampliación o especialización que podrían ser abordadas en versiones futuras del sistema.

- **Acceder a datos y estadísticas de un jugador concreto:** Centrarse en el jugador que el usuario elija para poder ver su evolución a lo largo de una o varias temporadas.
- **Añadir la opción de comparar varios equipos:** Implementar la posibilidad de “enfrentar” a dos o más equipos para ver cual es más efectivo en distintos aspectos del juego.
- **Añadir la opción de comparar jugadores:** Al igual que con los equipos, sería interesante implementar la posibilidad de enfrentar dos o más jugadores del mismo o distintos, equipos para ver cuál es más efectivo en determinados aspectos del juego.
- **Ampliación del número de equipos disponibles:** Aunque finalmente se ha añadido la posibilidad de analizar más equipos a parte del Cádiz, se podría ampliar este número e incluso añadir equipos de otros países o de divisiones inferiores.
- **Traducción de la aplicación:** Se podría añadir la opción de ver la WebApp en otros idiomas como el inglés o francés, sobre todo si se terminan añadiendo equipos de dichas ligas.





# Información sobre Licencia

---

Este proyecto está sujeto a un Acuerdo de Licencia de Usuario Final (EULA). Está prohibida la redistribución del software o su uso en aplicaciones que no estén explícitamente autorizadas por la licencia. Los usuarios no pueden modificar, distribuir, o crear obras derivadas del software sin consentimiento explícito.





# Bibliografía

---

- AlfonsoX, U. (2024). El impacto del big data en el fútbol: innovación y estrategias.
- Balsamiq (2008). Balsamiq.
- Cadiz (1910). Cadiz cf.
- Docker (2013). Docker.
- Drawio (2000). Drawio.
- FBRef (2022). Resultados y partidos de cádiz en la 2022-2023.
- foundation, R. (1993). The r project for statistical computing.
- Fútbol con datos, G. C. (2023). Las principales plataformas proveedoras de datos de fútbol: ¿cuál es la mejor para ti?
- Garcia-Planas, Y. (2025). El superordenador dice quién ganará la eurocopa.
- Github (2008). Github.
- LaLiga (1928). Laliga.
- OpenRefine (2023). Openrefine.
- Overleaf (2013). Overleaf.
- R.studio (2012). Shiny framework.
- SAS, S. (2022). Skillcorner.
- Segurola, S. (2024). Kroos desprecia el imperio del ‘big data’.
- talent (2025). Ingeniero informático: salario promedio en españa, 2025.
- Vivanco, J. (2019). El modelo c4 de documentación para la arquitectura de software.



# Parte IV

## Anexos

# A. Manual del desarrollador

---

A continuación se recogen las instrucciones necesarias para evolucionar el software. Este manual está dirigido a los desarrolladores que pretenden extender o modificar el código fuente, con el fin de incorporar nuevas funcionalidades o modificar las ya existentes.

## A.1. Introducción

Este proyecto consiste en una aplicación web cuya finalidad es facilitar el análisis de datos y predicción de resultados de un equipo de fútbol para un aficionado promedio, sin la necesidad de que este cuente con conocimientos y/o estudios previos sobre estadística, análisis de datos, Big Data, etc.

Para unos primeros datos de ejemplo, hemos tomado los de la temporada 2023/2024 del Cádiz CF. No obstante, la aplicación cuenta con la posibilidad de elegir temporadas anteriores para analizar, así como cambiar de equipo.

El software desarrollado es una WebApp construida en el lenguaje R utilizando el framework Shiny, que permite desarrollar aplicaciones web interactivas directamente desde este lenguaje. La arquitectura del sistema es sencilla: se apoya en un archivo principal `app.R` que gestiona la aplicación Shiny, en una serie de scripts en R que contienen el procesamiento y análisis de los datos, en una carpeta de datos donde se guardan tanto los archivos originales como los procesados, y en una carpeta `www` que almacena recursos estáticos como imágenes de escudos en formato `.png`. Además, la aplicación hace uso de librerías de R ampliamente utilizadas en análisis de datos como `shiny`, `ggplot2` y `dplyr`, entre otras.

Como he comentado anteriormente, esta aplicación está pensada para que un aficionado promedio sin conocimientos previos en el área de análisis de datos pueda comprender y visualizar los resultados del análisis de cada una de las categorías de las que dispone la aplicación y entender qué conclusiones pueden sacarse para ayudar a mejorar el equipo. Además, se ha implementado una sección de predicción donde, según los resultados de temporadas anteriores, se podrá ver qué resultados futuros se esperan.

## A.2. Preparación del entorno de trabajo

Para preparar su entorno de trabajo, puede utilizar cualquier S.O. de su preferencia, siendo Windows 10 el usado durante el desarrollo y recomendado.

En cuanto a los requisitos de hardware, basta con disponer de un equipo con un procesador moderno de al menos dos núcleos, 8 GB de memoria RAM y conexión a

internet estable para poder instalar librerías y descargar dependencias. No se requieren tarjetas gráficas específicas ni hardware especializado, dado que el procesamiento de los datos y la generación de gráficos estadísticos se realizan de forma eficiente en un ordenador convencional.

En lo referente al software, se debe contar con el lenguaje R en su versión 4.2 o superior y con RStudio como entorno de desarrollo recomendado. Además, es necesario instalar las principales librerías empleadas en el proyecto, entre ellas shiny para la creación de la aplicación web, ggplot2 para la visualización de datos, dplyr para la manipulación de conjuntos de datos y otras complementarias según se vaya ampliando la funcionalidad del sistema. La instalación de estas librerías se realiza directamente desde R y están indicadas en el documento principal de la app.

El control de versiones del software se gestiona mediante Git, siendo recomendable clonar el repositorio del proyecto desde una plataforma como GitHub o GitLab. Esto permite mantener un historial de cambios y desplegar nuevas versiones de forma controlada.

La instalación en local del entorno de desarrollo es sencilla: una vez instalado R y RStudio, se clona el repositorio del proyecto y se abren los archivos en el entorno. El archivo principal app.R centraliza la ejecución de la aplicación, de modo que basta con abrirlo en RStudio y ejecutar el comando `runApp()` para lanzar la aplicación en un navegador web. Los datos utilizados por la aplicación se encuentran en la carpeta correspondiente y se cargan automáticamente al iniciar el proyecto, por lo que no se necesita configuración adicional.

En este caso concreto no es necesario instalar bases de datos ni servidores de aplicaciones adicionales, ya que Shiny incorpora un servidor integrado que permite la ejecución de la WebApp en local y, en caso de despliegue, puede ejecutarse directamente en un servidor Shiny remoto o empaquetarse en un contenedor Docker. Esta simplicidad en los requisitos facilita que cualquier desarrollador pueda instalar y ejecutar el proyecto sin dificultades técnicas más allá de la instalación inicial de R, RStudio y las librerías necesarias.

### A.3. Consideraciones generales sobre el desarrollo

Es importante seguir las siguientes pautas a la hora de modificar y extender el código fuente:

- Utilizar un sistema de control de versiones como Git para gestionar las modificaciones en el código.
- El código debe ampliarse de manera modular, mediante funciones en scripts separados o integrando módulos de Shiny en el documento app.R que permitan mantener el código limpio y fácilmente escalable.
- Por cada componente nuevo que se realice, se deberán hacer pruebas unitarias, de integración y de rendimiento para comprobar que el funcionamiento tanto de la nueva función como de la aplicación después de añadirla es correcto.

- Mantener una política de documentación continua, registrando en el repositorio las principales modificaciones, las dependencias adicionales necesarias y las instrucciones específicas para reproducir los resultados.

## A.4. Instrucciones para construcción

Para poder construir y ejecutar la aplicación de forma correcta en una máquina de desarrollo será necesario seguir la siguiente secuencia de pasos:

En primer lugar, se debe clonar o descargar el repositorio del proyecto desde el siguiente enlace de github <https://github.com/UnaiZoot/TFG-UNAI-CADIZ>. A continuación, se abre el proyecto en RStudio y se comprueba que están instaladas todas las librerías necesarias mediante el comando *install.packages()*. En caso de que falte alguna dependencia, esta deberá instalarse antes de ejecutar la aplicación.

Una vez preparado el entorno, la aplicación se construye y ejecuta abriendo el archivo principal app.R dentro de RStudio y ejecutando el comando `runApp()`. Esto inicia el servidor Shiny en local y abre automáticamente la aplicación en el navegador web predeterminado. De esta manera, el desarrollador puede interactuar con la aplicación, comprobar su funcionamiento y validar las modificaciones introducidas en el código.







## B. Manual de usuario

---

Las instrucciones de uso del software se detallan a continuación. Este manual se dirige al usuario final del software objetivo de este proyecto.

### B.1. Introducción

La aplicación desarrollada tiene como principal objetivo acercar el análisis de datos en el fútbol al aficionado promedio, permitiéndole comprender de manera sencilla aspectos clave del rendimiento de un equipo y de sus jugadores. El ámbito del proyecto se centra en el Cádiz CF durante la temporada 2023/2024, última en primera división, y utiliza datos tanto individuales como colectivos para mostrar comparativas, estadísticas y visualizaciones que ayudan a entender mejor el desempeño del club. Además, se incluye la opción de ver predicciones en distintas categorías para visualizar los resultados esperados, teniendo en cuenta los de temporadas anteriores.

El primer paso consiste en descargar o clonar el repositorio del proyecto desde la plataforma en la que se encuentre publicado, en este caso desde el repositorio [Github](#). Una vez obtenido, se debe abrir la carpeta del proyecto en RStudio y asegurarse de instalar las librerías necesarias. Para ello, basta con ejecutar en la consola de R el comando `install.packages()`.

Cuando todas las dependencias estén instaladas, el usuario debe abrir el archivo principal `app.R` y pulsar en el botón Run App de RStudio o bien ejecutar el comando `runApp()` en la consola. Esto iniciará automáticamente la aplicación en un navegador web.

### B.2. Uso del sistema

Una vez iniciada la aplicación, el usuario accederá a la interfaz principal en su navegador. Esta interfaz ha sido diseñada para ser intuitiva y fácil de manejar, incluso para personas sin conocimientos técnicos. En la página inicial se presenta una pequeña explicación sobre el contenido de la web y sobre qué es el Big Data, ayudando así al usuario a entender lo que se va a encontrar durante el uso de la aplicación. El usuario tendrá dos botones para acceder a la pestaña de Análisis o Cambiar de equipo, respectivamente.



Figura B.1: Inicio

Tras acceder a la pestaña “Análisis”, el usuario podrá navegar por varias categorías diferentes: Goles, Goles en Contra, Tiros de jugador, Resultados, Tiros y Tiros en Contra. En cada una de estas categorías podrá visualizar gráficas, explicaciones y conclusiones de distintos análisis. Además, se le da al usuario la posibilidad de elegir las temporadas que desea tomar en cuenta para los análisis, pudiendo elegir una o varias temporadas anteriores.



Figura B.2: Análisis

En la pestaña “Predicción”, el usuario podrá acceder a las mismas categorías que en “Análisis”. En cada una de estas categorías podrá ver los resultados esperados en distintos aspectos del juego, además de explicaciones y conclusiones. Igualmente, el usuario podrá volver a elegir qué temporadas usar para sólo tener en cuenta esos datos a la hora de realizar una predicción.



Figura B.3: Predicción

Además, el usuario podrá acceder a una pestaña “Glosario” donde encontrará varias definiciones de distintos términos relacionados con el Big Data o que se usan a lo largo de la WebApp y que un usuario sin conocimientos previos pueda llegar a no entender bien. De este modo, el usuario podrá consultar el glosario en todo momento en el caso de encontrarse con un concepto que no comprenda.

Figura B.4: Glosario

Por último, el usuario tendrá la posibilidad de cambiar de equipo. Al seleccionar esta opción, se le presentarán varios equipos conocidos de primera división y el usuario podrá elegir su preferido para analizar. Se volverá a abrir la aplicación y se mostrarán los datos actualizados del equipo elegido.



Figura B.5: Cambio de equipo



