



Escuela Técnica Superior de Ingeniería Informática

Trabajo de Fin de Grado

Grado en Ingeniería Informática - Tecnologías Informáticas

RECOGIDA DE DATOS EN BALONCESTO CON STACK MEAN

Realizado por

JAIRO FERNÁNDEZ CARMONA

Dirigido por

JOSÉ ANTONIO TROYANO JIMÉNEZ

Departamento

LENGUAJES Y SISTEMAS INFORMÁTICOS

Curso 2019/2020

Agradecimientos

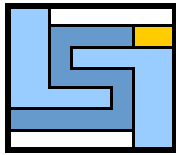
Quiero agradecer a varias personas el apoyo y ayuda prestado durante el transcurso de estos años en la facultad de Ingeniería Informática y el desarrollo de este trabajo de fin de grado.

En primer lugar, mencionar a mi tutor, José Antonio Troyano Jiménez, ya que la oferta de un trabajo de fin de grado tan interesante y atractivo ha servido de acicate durante los meses en los que se ha desarrollado el proyecto, ya que aúna mis dos grandes pasiones, la informática y el baloncesto.

Agradecer también la confianza y ánimos recibidos de parte de mi familia, con especial mención a mi madre, por ser un refuerzo básico y referente, sin su apoyo nunca podría haber llegado a ser Ingeniero Informático.

Así mismo no puedo obviar el apoyo de mis amigos, ha sido incondicional desde el primer momento, destacando a Rafa, cuya ayuda y comprensión han sido indispensables durante estos años.

Para finalizar, es imposible no mencionar a mis compañeros de carrera Kike, Carlos, Balboa y Juan, que desde el primer día han sido un ejemplo de camaradería indescriptible, no hay palabras para agradecer su compañía y las experiencias vividas en todo este tiempo, sabiendo que esta amistad no finalizará aún habiendo cumplido nuestro objetivo académico.



Resumen

Contexto

Este trabajo pretende realizar una aplicación online para equipos de baloncesto en la que se pueda realizar un registro estadístico de los acontecimientos de partidos en tiempo real, poniendo especial atención en la posterior visualización de estos datos mediante tablas y la comparación de jugadores y equipos a nivel estadístico.

Además de lo expuesto anteriormente la aplicación permitirá las acciones necesarias para cumplir este objetivo como son la creación y edición de los perfiles de jugadores, equipos, clubs y ligas.

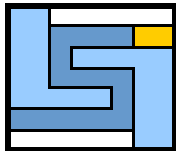
La interfaz desarrollada en el proyecto será lo más intuitiva posible para facilitar su uso al máximo, intentando que la accesibilidad sea total para cualquier tipo de usuario.

Resultados

El proyecto desarrollado ha dado como resultado un portal web de registro y análisis de estadísticas de baloncesto que ha cumplido con los objetivos técnicos y académicos planteados al diseñarlo.

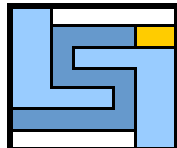
Índice general

1. Introducción.....	1
1.1 Conceptualización	2
1.2 Contexto	2
1.3 Objetivos técnicos	3
1.4 Objetivos personales o académicos	4
2. Metodología	6
2.1 Planificación.....	7
2.2 Resumen temporal del proyecto.....	7
2.3 Resumen y temporización de hitos	8
3. Estimación de costes	10
3.1 Costes de personal	11
3.2 Costes materiales	11
3.2.1 Costes de hardware	11
3.2.2 Costes de software	12
4. Estado del arte.....	13
4.1 Opinión inicial.....	14
4.2 Comparativa de aplicaciones existentes	14
4.2.1 Boxscore for basketball.	14
4.2.2 Basketball Stats Assistant	18
4.2.3 Breakthrough Stats App.....	22
4.3 Necesidades y opiniones de entrenadores y jugadores	24
4.4 Conclusiones relativas al estado del arte	25
5. Elicitación de requisitos	27



5.1 Introducción	28
5.2 Participantes en el proyecto	28
5.3 Descripción del Sistema Actual	29
5.4 Objetivos del sistema	30
5.5 Catálogo de requisitos del sistema	33
5.5.1 Requisitos de información	33
5.5.2 Requisitos funcionales	44
5.5.3 Requisitos No Funcionales	92
6. Diseño e implementación	96
6.1 Stack tecnológico	97
6.2 Desarrollo	98
6.2.1 MongoDB Atlas	98
6.2.2 Seguridad mediante tokens	99
6.2.3 Hosting de imágenes en Cloudinary	99
6.2.4 Creación de gráficos con Echarts	100
7. Manual de usuario	101
7.1 Login	102
7.2 Partidos	102
7.2.1 En Vivo	103
7.2.2 Boxscore	107
7.2.3 Play by Play	108
7.2.4 Crear Partido	109
7.3 Jugadores	110
7.3.1 Crear Jugador	110
7.3.2 Consultar Jugadores	111
7.3.3 Perfil Jugador	111
7.4 Equipos	113

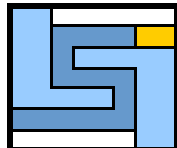
7.4.1 Perfil de Equipo	113
7.5 Estadísticas	114
8. Conclusiones	116
8.1 Evaluación del proyecto	117



Índice de figuras

4.1. Interfaz de jugadores en partido en <i>Boxscore for basketball</i>	15
4.2. Interfaz de selección de acción de jugador en <i>Boxscore for basketball</i>	16
4.3. Interfaz de resumen de partido en <i>Boxscore for basketball</i>	16
4.4. Pantalla de boxscore de partido en <i>Boxscore for basketball</i>	17
4.5. Pantalla de partido en <i>Basketball Stats Assistant</i>	19
4.6. Estadísticas de jugador en <i>Basketball Stats Assistant</i>	19
4.7. Registro de tiros de un jugador en <i>Basketball Stats Assistant</i>	20
4.8. Ventajas en el marcador de un partido en <i>Basketball Stats Assistant</i>	21
4.9. Líderes estadísticos del partido en <i>Basketball Stats Assistant</i>	21
4.10. Boxscore del partido en <i>Basketball Stats Assistant</i>	22
4.11. Esquema de interfaz de partido en <i>Breakthrough Stats App</i>	23
4.12. Interfaz de partido en <i>Breakthrough Stats App</i>	23
5.1. Diagrama de subsistemas.	46
5.2. Diagrama de Casos del Subsistema Gestión de Usuarios.....	47
5.3. Diagrama de Casos del Subsistema Gestión de Partidos.....	48
5.4. Diagrama de Casos del Subsistema Gestión de Clubs.....	49
5.5. Diagrama de Casos del Subsistema Gestión de Equipos.	50
5.6. Diagrama de Casos del Subsistema Gestión de Ligas.	51
7.1. Login.	102
7.2. Pantalla de Partidos.....	103
7.3. Pantalla de Partido: En Vivo.	104
7.4. Pantalla de Partido: Registrar Jugada.	105
7.5. Pantalla de Partido: Shotchart.	106
7.6. Pantalla de Partido: Banquillo.....	107
7.7. Pantalla de Partido: Boxscore.....	108

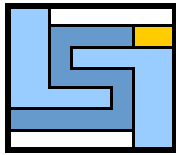
7.8. Pantalla de Partido: Play by Play.....	109
7.9. Crear Partido.....	110
7.10. Crear Jugador.....	110
7.11. Consultar Jugadores.....	111
7.12. Perfil de Jugador.....	112
7.13. Perfil de Jugador: Estadísticas de tiro.....	112
7.14. Perfil de Jugador: Comparador de estadísticas.....	113
7.15. Perfil de Jugador: Comparador de heptágonos.....	113
7.16. Perfil de Equipo.....	114
7.17. Perfil de Equipo: Estadísticas de temporada.....	114
7.18. Tabla comparativa de Jugadores.....	115



Índice de cuadros

2.1. Tabla de resumen temporal del proyecto.....	7
2.2. Tabla de resumen de hitos.	8
5.1. Participante: Jairo Fernández Carmona	28
5.2. Participante: José Antonio Troyano Jiménez	29
5.3. Organización: Departamento de Lenguajes y Sistemas Informáticos	29
5.4. Objetivos relacionados con la interfaz de recopilación estadística de un partido.	31
5.5. Objetivos relacionados con mostrar análisis estadístico de jugadores y equipos.	31
5.6. Objetivos relacionados con proporcionar herramientas para el análisis estadístico.	32
5.7. Objetivos relacionados con permitir cuentas de usuario.	33
5.8. IRQ-0001: Requisitos de Información sobre Jugadores.....	35
5.9. CRQ-0001: Restricción sobre Unicidad de Identificador de Jugador.....	35
5.10. IRQ-0002: Requisitos de Información sobre Clubs.	36
5.11. CRQ-0002: Restricción sobre Unicidad de Identificador de Club.	37
5.12. IRQ-0003: Requisitos de Información sobre Partidos.	38
5.13. CRQ-0003: Restricción sobre Unicidad de Identificador de Equipo.	39
5.14. IRQ-0005: Requisitos de Información sobre Equipos.....	40
5.15. CRQ-0005: Restricción sobre Unicidad de Identificador de Partido.	40
5.16. IRQ-0006: Requisitos de Información sobre Usuarios.	41
5.17. CRQ-0006: Restricción sobre Unicidad de Identificador de Usuario.	42
5.18. IRQ-0007: Requisitos de Información sobre Ligas.....	43
5.19. CRQ-0007: Restricción sobre Unicidad de Identificador de Liga.....	44
5.20. ACT-0001: Actor Administrador.	44
5.21. ACT-0002: Actor Empleado.	45
5.22. ACT-0003: Actor Usuario.....	45

5.23: UC-0001: Alta de Usuario.	53
5.24: UC-0002: Baja de Usuario.	54
5.25: UC-0003: Identificación de Usuario.	55
5.26: UC-0004: Edición de Usuario.	57
5.27: UC-0005: Consulta de Usuario.	58
5.28: UC-0006: Registro de Partido.	60
5.29: UC-0007: Editar Partido.	61
5.30: UC-0008: Borrar Partido.	62
5.31: UC-0009: Añadir acción.	63
5.32: UC-0010: Borrar acción.	65
5.33: UC-0011: Consultar Partido.	66
5.34: UC-0012: Añadir Club.	67
5.35: UC-0013: Editar Club.	69
5.36: UC-0014: Borrar Club.	70
5.37: UC-0015: Consultar Club.	71
5.38: UC-0016: Añadir Equipo.	72
5.39: UC-0017: Editar Equipo.	74
5.40: UC-0018: Borrar Equipo.	75
5.41: UC-0019: Consultar Equipo.	76
5.42: UC-0020: Comparar Jugadores.	77
5.43: UC-0021: Comparar Equipos.	79
5.44: UC-0022: Comparar Jugadores en Eje de Coordenadas.	80
5.45: UC-0023: Comparar Equipos en Eje de Coordenadas.	82
5.46: UC-0024: Añadir Jugador.	83
5.47: UC-0025: Editar Jugador.	84
5.48: UC-0026: Borrar Jugador.	86
5.49: UC-0027: Consultar Jugador.	87
5.50: UC-0028: Añadir Liga.	88
5.51: UC-0029: Editar Liga.	90

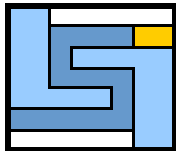


5.52: UC-0030: Borrar Liga.....	91
5.53: UC-0031: Consultar Liga.	92
5.54: NFR-0001: Accesibilidad.	93
5.55: NFR-0002: Calidad.....	93
5.56: NFR-0003: Seguridad.....	94
5.57: NFR-0004: Uso de Software de Actualidad.	95

Capítulo 1

1. Introducción

En este capítulo pretendemos comenzar el proyecto marcando unas pautas que deben servirnos para definir las líneas generales del mismo, dando un contexto de los softwares enfocados a la recopilación y cálculo de estadísticas en el baloncesto moderno y definiendo los objetivos principales del trabajo.



1.1 Conceptualización

El proyecto que vamos a realizar es el Trabajo de Fin de Grado, asignatura de 12 créditos, según la conversión de créditos ECTS que sigue la Escuela Técnica Superior de Ingeniería Informática de la Universidad de Sevilla el trabajo tendrá una duración aproximada de 300 horas por parte del alumno.

A continuación pasamos a definir los roles de este proyecto, que son:

- **Cliente:** el tutor del Trabajo de Fin de Grado, José Antonio Troyano Jiménez.
- **Jefe de proyecto:** el alumno Jairo Fernández Carmona. El trabajo del jefe de proyecto es conseguir el objetivo inicial marcado para este trabajo, siendo en nuestro caso la realización de una aplicación que facilite la recopilación en tiempo real de datos deportivos en partidos de baloncesto.
- **Patrocinadores:** incluimos aquí a la Escuela Técnica Superior de Ingeniería Informática y por ende a la Universidad de Sevilla, que nos proveen de los recursos necesarios para la realización de este proyecto.
- **Interesados:** cualquier persona o entidad que pueda beneficiarse de nuestro proyecto.
- **Usuarios:** todas las personas que vayan a utilizar nuestra aplicación, el perfil de usuario puede ser alguien que trabaja para un club de baloncesto y quiere recopilar datos de los encuentros disputados para analizar posteriormente las estadísticas y mejorar el rendimiento de los jugadores o equipos implicados.

1.2 Contexto

En los últimos años hemos podido comprobar como la estadística aplicada al deporte ha evolucionado de forma notable, cada vez disponemos de más fuentes de información al respecto (como medios de comunicación especializados o personas individuales) y existen más empresas dedicadas a la recopilación y venta de forma profesional de estos datos, cuyos clientes, equipos profesionales o semiprofesionales demandan con mayor asiduidad, siendo ya un tema de total actualidad y en el que cualquier seguidor, sin necesidad de dedicarse a ello profesionalmente, puede realizar un acercamiento para estar al tanto de las últimas novedades.

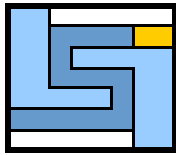
Como hemos comentado, cada vez hay más empresas dedicadas a la venta de estos datos, para que los deportistas o los empleados de la entidad a la que pertenecen analicen estos datos, con el objetivo de mejorar la actuación individual y colectiva en el ámbito deportivo. Una de las dificultades por las que realizar la recopilación de estos datos es ardua es la profundidad y variedad de datos que se manejan en el deporte actual, haciendo mención especial al deporte en el que nos enfocaremos en este proyecto, el baloncesto, que cuenta con una amplia gama de datos y fórmulas mediante las cuales se trata de medir el rendimiento de jugadores y equipos. Podemos citar como ejemplos de esta variedad el *boxscore* más tradicional (tabla en la que recogemos datos simples como los puntos anotados por un jugador, las asistencias dadas o los rebotes capturados) como datos que se utilizan desde hace varias generaciones, y algunos datos más actuales, que necesitan de ciertos cálculos matemáticos, de la llamada *estadística avanzada* aplicada al baloncesto, como puede ser el porcentaje de rebotes de un jugador (rebotes capturados por el jugador sobre el total de rebotes mientras el jugador está en pista).

Podemos entender con facilidad entonces la importancia de softwares en los que podamos registrar las acciones cotidianas de un partido de forma rápida y sencilla y que tales acciones sean guardadas y tratadas por el sistema para mostrar finalmente los datos requeridos por los distintos interesados.

1.3 Objetivos técnicos

A continuación hablaremos de los objetivos a cumplir por el proyecto desde un punto de vista técnico y que ampliaremos en el capítulo 4 con la elicitación de requisitos.

Analizando todo lo dicho anteriormente imbuimos que, aunque existe gran variedad de softwares que ofertan la recopilación y trata de datos estadísticos en baloncesto estos no tienen un coste bajo, ya que están diseñados para equipos con cierto presupuesto y categoría. Por lo que este proyecto tiene la intención de desarrollar un software que englobe todas estas tareas antes mencionadas para cualquier equipo con poco presupuesto, que no pueda permitirse la compra de estos sistemas y que no necesite de una profundidad estadística comparable a un equipo profesional o de una categoría deportiva muy elevada.



Con vistas a finalizar este proyecto de forma satisfactoria podemos identificar una serie de objetivos que nos ayudarán a marcar una línea de referencia durante el desarrollo del software, los cuales definimos a continuación:

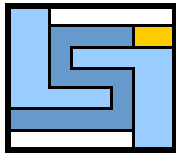
- **Objetivo 1:** Estudiar el *estado del arte* de las aplicaciones existentes con la finalidad deseada.
- **Objetivo 2:** Una vez realizado el estudio analizar tales aplicaciones para decidir qué características específicas debería comprender nuestra aplicación, teniendo en cuenta que deberíamos ofrecer una aplicación que mejore a las que sean gratuitas y que ofrezca una ventaja real respecto a las mismas.
- **Objetivo 3:** Teniendo asentada la idea base de nuestro software, hablar con posibles usuarios de nuestra aplicación para definir con mayor exactitud las necesidades reales que debe comprender nuestro software.
- **Objetivo 4:** Desarrollar la aplicación teniendo en mente que debe realizarse con el menor coste posible (cumpliendo, como ya dijimos en el punto anterior, que debemos crear un sistema de calidad), para que así, los equipos con poco presupuesto puedan implementarla con facilidad.

1.4 Objetivos personales o académicos

En este trabajo se pretendía no solo desarrollar un proyecto de calidad técnicamente hablando, sino también alcanzar un desarrollo tanto personal como académico-profesional que fuera perceptible al terminar el mismo. Como objetivos primarios de tal desarrollo podemos citar los siguientes.

- **Objetivo 1:** formarse sobre las bases de datos NoSQL (MongoDB en nuestro caso), comprendiendo su estructura, pros y contras a la hora de elegir entre SQL o NoSQL.
- **Objetivo 2:** desarrollar un front end utilizando AngularJS, al ser un framework demandado en la actualidad.

- **Objetivo 3:** realizar el trabajo de forma autónoma, autoimponiendo una metodología de trabajo que se ajuste a las preferencias personales y asegurara a la vez la motivación continua sobre el trabajo realizado.



Capítulo 2

2. Metodología

***P**ara la planificación del proyecto tendremos en cuenta el tiempo a emplear, el periodo disponible para realizar todas las tareas necesarias y el reparto de las mismas en distintas fases, obteniendo como resultado final el cómputo de horas trabajadas.*

2.1 Planificación

Para realizar este trabajo hemos utilizado la metodología ágil Scrum, estableciendo hitos (o sprints) con una duración de entre una o dos semanas cada uno, dependiendo de la cantidad de tareas previstas para el hito. Al final de cada hito se analizaba el cumplimiento (o no) de las distintas tareas y se comparaba el desempeño inicial estimado con el que finalmente se invertía, para así mejorar esta estimación de cara al siguiente hito. Las tareas que no se cumplieran en el hito previsto se posponían al siguiente, además de añadir las nuevas.

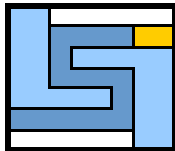
Para comenzar con la planificación de cada sprint nuestra primera tarea era registrar en el *Project Management* de nuestro proyecto en **Github** las tareas a realizar en el hito que iba a comenzar, estimando un número y complejidad de tareas que nos permitiese cumplir los plazos, registrando estas tareas como *issues* y utilizando etiquetas para distinguir estas tareas entre *feature*, *bug* o *documentation*. Una vez teníamos las tareas en nuestro *backlog* utilizábamos la integración de **Toggl** con Github para cronometrar el tiempo empleado en cada una.

2.2 Resumen temporal del proyecto

A continuación mostramos la temporización del proyecto. Cabe destacar que aunque la primera reunión tiene lugar el miércoles 19 de febrero de 2020 (fecha en la cual se da por iniciado el proyecto) no es hasta el mes de junio que se comienza el trabajo, coincidiendo con la finalización del curso lectivo, habiendo sido imposible compaginar ambos proyectos.

Fecha de inicio	19/02/2020
Fecha de fin	01/12/2020
Periodicidad de los hitos	1-2 semanas
Horas totales previstas	300 horas
Horas totales empleadas	297 horas

Cuadro 2.1: Tabla de resumen temporal del proyecto.



2.3 Resumen y temporización de hitos

Ya hemos mencionado en el apartado anterior la distribución temporal del trabajo en sprints, ahora vamos a definirlos y mostrar la temporización utilizada para cada uno.

Sprint 1	Creación del servidor y base de datos inicial.	15/06/20 a 28/06/20
Sprint 2	Creación de métodos en la base de datos.	10/08/20 a 26/08/20
Sprint 3	Inicio de la implementación del front end.	1/9/20 a 8/9/20 y 21/9/20 a 30/9/20
Sprint 4	Creación de los formularios, vistas de la web y servicios.	1/10/20 a 18/10/20
Sprint 5	Implementación de métodos de cálculo estadístico.	19/10/20 a 25/10/20
Sprint 6	Implementación de seguridad en servidor y web y finalización del front end.	26/10/20 a 1/11/20
Sprint 7	Redacción de la memoria del trabajo.	2/11/20 a 22/11/20

Cuadro 2.2: Tabla de resumen de hitos.

Detallamos a continuación el objetivo y las horas de trabajo de cada sprint. Las primeras semanas, para poder realizar un proyecto de calidad, hemos realizado un curso de NodeJS y otro de AngularJS, ambos en la plataforma de aprendizaje Udemy, la duración total de estos cursos ha superado las 50 horas.

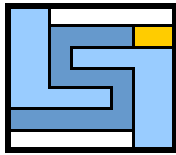
- **Sprint 1:** Investigación y aprendizaje de MongoDB, creación del servidor en Node utilizando Express, creación del documento de requisitos y del estudio del arte (59 horas, 4 minutos).

- **Sprint 2:** Implementación de operaciones CRUD en el servidor y realización de cursos de NodeJS y AngularJS (70 horas, 57 minutos).
- **Sprint 3:** Implementación de formularios de creación y edición de entidades, primeros aspectos visuales de la web y corrección de esquemas y métodos de la base de datos (34 horas, 21 minutos).
- **Sprint 4:** Creación de vistas de la web, servicios y modelos de AngularJS (43 horas, 43 minutos).
- **Sprint 5:** Implementación de métodos de cálculos estadísticos de los equipos y jugadores (34 horas, 15 minutos).
- **Sprint 6:** Añadidos finales del apartado visual e implementación de la seguridad correspondiente a la API y a la web (18 horas, 31 minutos).
- **Sprint 7:** Redacción de la memoria de trabajo y análisis de objetivos y alcance (36 horas, 9 minutos).

Para poder comprender con mayor facilidad la división de estos hitos y observar las dificultades encontradas durante el transcurso del proyecto en forma de tiempo dedicado a cada tarea adjuntamos a continuación el link del proyecto en GitHub, donde podemos ver un desglose de las horas trabajadas, generado mediante Toggl.

[Enlace de GitHub.](#)

Podemos observar como las tareas correspondientes a la implementación de los métodos necesarios para registrar las acciones del partido (tarea #46 *Plays Methods*) y la continua actualización de los esquemas y métodos de la base de datos a medida que íbamos ampliando las propiedades a almacenar para su posterior trato estadístico (tarea #14 *Update database methods*) han sido con mucha diferencia las que han necesitado de más horas de trabajo y en las que hemos encontrado mayor dificultad (excluyendo la creación de documentación y la realización de cursos).



Capítulo 3

3. Estimación de costes

En el siguiente capítulo desglosaremos los costes de la plataforma realizada, que como ya mencionamos tiene como objetivo ser un software al alcance de equipos con poco presupuesto, por lo que como veremos se ha intentado abaratar su coste al máximo mientras se aseguraba su futura escalabilidad.

3.1 Costes de personal

Teniendo en cuenta la naturaleza del proyecto dividiremos el trabajo realizado en varios roles que serán un *Programador junior*, un *Analista* y un *Diseñador web*. El cálculo de los costes de personal lo podemos realizar consultando el XVII CONVENIO COLECTIVO ESTATAL DE EMPRESAS DE CONSULTORÍA Y ESTUDIOS DE MERCADO Y DE LA OPINIÓN PÚBLICA [1], en el que se especifica y regula el salario mínimo para cada uno de los roles antes enunciados.

Mostraremos el salario mínimo anual y lo desglosaremos para mostrar el coste total del proyecto.

- **Programador junior:** 14.800'66 €/año, **7'71 €/hora**.
- **Analista programador:** 22.993'74 €/año, **11'97 €/hora**.
- **Diseñador web:** 22.993'74 €/año, **11'97 €/hora**.

Desglosando las horas trabajadas en cada rol de la siguiente manera:

- **Programador junior:** 204 horas.
- **Analista programador:** 15 horas.
- **Diseñador web:** 78 horas.

Podemos calcular los costes totales:

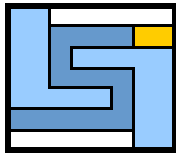
- **Programador junior:** 7'71 €/hora * 204 horas = **1.572'84€**.
- **Analista programador:** 11'97 €/hora * 15 horas = **179'55€**.
- **Diseñador web:** 11'97 €/hora * 78 horas = **933'66€**.

Lo que aumenta a un total de **2.686'05€**.

3.2 Costes materiales

3.2.1 Costes de hardware

Para la realización de este proyecto hemos utilizado un ordenador de sobremesa con un coste actual de 536'71€, un teclado con un coste de 80€, un monitor con un coste de 105€ y un ratón con un coste de 36€, lo que asciende a un total de **757'71€** en hardware.



3.2.2 Costes de software

En el apartado software, como ya explicamos en anteriores apartados, hemos intentado abaratar el coste lo máximo posible, eligiendo opciones que permitieran una escalabilidad futura del proyecto manteniendo el coste al mínimo.

Enumeramos a continuación las herramientas utilizadas en el trabajo.

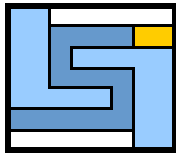
- **Visual Studio Code:** IDE de desarrollo gratuito y con múltiples extensiones adicionales.
- **MEAN stack:** el stack es gratuito en su totalidad.
- **Github:** versión gratuita.
- **MongoDB Atlas:** versión gratuita de la web en la nube que puede almacenar hasta 512MB.
- **Cloudinary:** versión gratuita del portal que nos habilita la subida y almacenamiento de imágenes mediante su API, permite la creación de un único usuario y un máximo de 25.000 subidas mensuales o un almacenamiento máximo de 25GB de imágenes.
- **Heroku:** versión gratuita que incluye el host y el dominio de nuestra aplicación.

Como podemos observar hemos realizado el proyecto con un coste en software mínimo y sin haber escatimado en calidad técnica a la hora de desarrollar el producto, si bien es cierto que en el caso de enfocar el trabajo a otro tipo de organizaciones deportivas los requisitos de rendimiento podrían aumentar y encontrar la necesidad de mejorar las prestaciones técnicas de los softwares.

Capítulo 4

4. Estado del arte

A continuación analizaremos el estado actual de aplicaciones que tengan como finalidad la recopilación de datos en partidos de baloncesto en tiempo real, detallando sus ventajas e inconvenientes, con el objetivo de ayudarnos a definir las características que debería incluir nuestra aplicación.



4.1 Opinión inicial

Para comenzar aportaremos nuestra opinión inicial sobre cual debe ser el objetivo de las aplicaciones del tipo especificado anteriormente para con una entidad deportiva relacionada con el baloncesto (como un equipo o club).

La aplicación debería aportar información útil para la mayoría del personal del quipo, tanto del cuerpo técnico como jugadores, a partir de los datos recopilados en los partidos disputados, sin olvidar además los datos más comúnmente utilizados en estos casos y de mayor acceso para el público en general. Para ello el sistema debe mostrar estos datos de forma clara, para que la interpretación de los mismos sea fácil y sirvan para analizar y mejorar el rendimiento colectivo e individual tanto técnica como tácticamente.

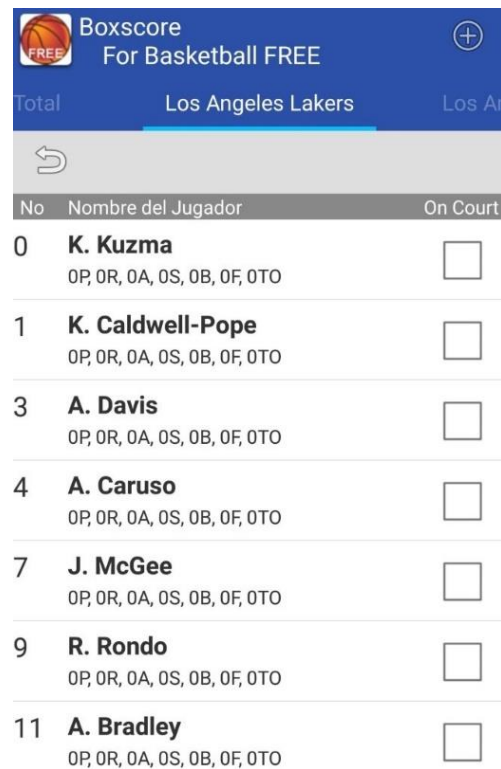
4.2 Comparativa de aplicaciones existentes

En este apartado analizaremos las distintas aplicaciones seleccionadas como ejemplos, desarrollando al final de cada sección sus puntos fuertes y débiles.

4.2.1 Boxscore for basketball.

Comenzamos con una aplicación gratuita para Android, que cumple con unos objetivos básicos y tiene una interfaz muy intuitiva y simple.

En la siguiente figura observamos la interfaz correspondiente a la lista de jugadores que pertenecen a un equipo determinado, dentro de la interfaz del partido. Podemos ver además un resumen bastante básico de estadísticas comunes (puntos, rebotes, asistencias, etc) debajo del nombre de cada jugador. El desarrollador permite seleccionar el quinteto de jugadores en pista con mucha facilidad mediante los respectivos *checkboxes* que se encuentran a la derecha de los nombres de jugadores.



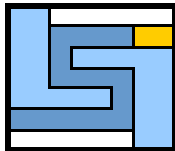
No	Nombre del Jugador	On Court
0	K. Kuzma OP, OR, OA, OS, OB, OF, OTO	<input type="checkbox"/>
1	K. Caldwell-Pope OP, OR, OA, OS, OB, OF, OTO	<input type="checkbox"/>
3	A. Davis OP, OR, OA, OS, OB, OF, OTO	<input type="checkbox"/>
4	A. Caruso OP, OR, OA, OS, OB, OF, OTO	<input type="checkbox"/>
7	J. McGee OP, OR, OA, OS, OB, OF, OTO	<input type="checkbox"/>
9	R. Rondo OP, OR, OA, OS, OB, OF, OTO	<input type="checkbox"/>
11	A. Bradley OP, OR, OA, OS, OB, OF, OTO	<input type="checkbox"/>

Figura 4.1: Interfaz de jugadores en partido en *Boxscore for basketball*.

La siguiente imagen se corresponde con la pantalla en la que podemos registrar una jugada (una canasta, ya sea anotada o fallada, asistencia, falta personal, etc).

Como parte positiva vemos como es una interfaz muy sencilla, ya que para registrar un suceso simplemente debemos pulsar en el nombre del jugador y seguidamente en el botón correspondiente al suceso que queremos añadir.

La parte negativa es que cada vez que registramos una acción la aplicación nos devuelve a la pantalla de jugadores (Figura 4.1), por lo que si deseamos añadir más de un suceso deberíamos pulsar en el botón correspondiente, volver a seleccionar el jugador de la lista y pulsar nuevamente en la acción elegida. En nuestra opinión se debería cambiar esta característica ya que puede aumentar el número de *clicks* necesarios para añadir varios sucesos realizados por un mismo jugador rápidamente, una posible solución sería mantener al usuario en la pantalla de selección de suceso y obligarle a pulsar en un botón de *atrás* cuando deseara volver a la lista de jugadores.

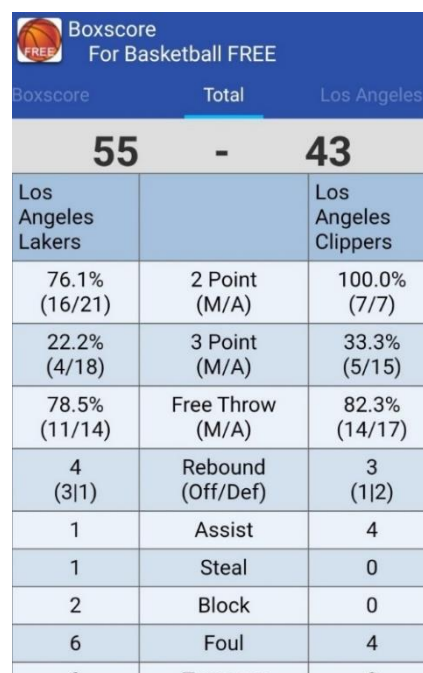


4.2. Comparativa de aplicaciones existentes



Figura 4.2: Interfaz de selección de acción de jugador en *Boxscore for basketball*.

A continuación vemos la pantalla de resumen de un partido en juego, con el marcador en la cima de la interfaz y una serie de estadísticas básicas de los equipos. Podemos observar como aunque esta pantalla cumple con la función que pretende la interfaz de resumen del partido carece de complejidad, al igual que el resto de pantallas.



Boxscore For Basketball FREE		
Boxscore	Total	Los Angeles L
55 - 43		
Los Angeles Lakers		Los Angeles Clippers
76.1% (16/21)	2 Point (M/A)	100.0% (7/7)
22.2% (4/18)	3 Point (M/A)	33.3% (5/15)
78.5% (11/14)	Free Throw (M/A)	82.3% (14/17)
4 (3/1)	Rebound (Off/Def)	3 (1/2)
1	Assist	4
1	Steal	0
2	Block	0
6	Foul	4
0	Turnover	0

Figura 4.3: Interfaz de resumen de partido en *Boxscore for basketball*.

Para finalizar vamos a ver la pantalla que puede que sea el atractivo principal de esta aplicación y es la correspondiente al *boxscore*. Como aspecto a favor de la aplicación y un hecho muy interesante es que desde este apartado tenemos la opción de subir las estadísticas para poder consultarlas más adelante, lo cual nos proporciona un link de acceso que hemos comprobado que sigue operativo meses después de su creación ([aquí](#) podemos ver un ejemplo). Por el contrario, un aspecto negativo concerniente a esta característica es que este *boxscore* es accesible por cualquier persona, ya que se aloja en la web oficial de la aplicación, la cual muestra una lista con todas las estadísticas subidas por los usuarios.



Boxscore For Basketball FREE

Boxscore Total

Subir Compartir Web Eliminar

16/3/20 18:55

Los Angeles Lakers

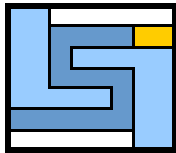
NAME	PTS	2P		3P		FT		REB			AS	ST	BS	F	TO
		M/A	%	M/A	%	M/A	%	OR	DR	TOT					
0 K. Kuzma	8	3/4	75.0	0/4	0.0	2/2	100.0	3	0	3	0	1	2	2	0
1 K. Caldwell-Pope	3	0/0	0.0	1/2	50.0	0/0	0.0	0	1	1	1	0	0	1	0
3 A. Davis	24	9/9	100.0	1/4	25.0	3/3	100.0	0	0	0	0	0	0	0	0
4 A. Caruso	0	0/0	0.0	0/0	0.0	0/0	0.0	0	0	0	0	0	0	1	0
7 J. McGee	0	0/0	0.0	0/0	0.0	0/0	0.0	0	0	0	0	0	0	0	0
9 R. Rondo	4	2/6	33.3	0/0	0.0	0/0	0.0	0	0	0	0	0	0	0	0
11 A. Bradley	0	0/0	0.0	0/0	0.0	0/0	0.0	0	0	0	0	0	0	1	0
14 D. Green	0	0/0	0.0	0/0	0.0	0/0	0.0	0	0	0	0	0	0	0	0
23 L. James	16	2/2	100.0	2/8	25.0	6/9	66.6	0	0	0	0	0	0	1	0
39 D. Howard	0	0/0	0.0	0/0	0.0	0/0	0.0	0	0	0	0	0	0	0	0
88 M. Morris	0	0/0	0.0	0/0	0.0	0/0	0.0	0	0	0	0	0	0	0	0
Total	55	16/21	76.1	4/18	22.2	11/14	78.5	3	1	4	1	1	2	6	0

Los Angeles Clippers

NAME	PTS	2P		3P		FT		REB			AS	ST	BS	F	TO
		M/A	%	M/A	%	M/A	%	OR	DR	TOT					
1 R. Jackson	0	0/0	0.0	0/0	0.0	0/0	0.0	0	0	0	1	0	0	1	0
2 K. Leonard	22	5/5	100.0	2/7	28.5	6/7	85.7	1	2	3	0	0	0	1	0
4 J. Green	0	0/0	0.0	0/0	0.0	0/0	0.0	0	0	0	0	0	0	0	0
5 M. Harrell	0	0/0	0.0	0/0	0.0	0/0	0.0	0	0	0	0	0	0	1	0
13 P. George	21	2/2	100.0	3/8	37.5	8/10	80.0	0	0	0	3	0	0	1	0
20 L. Shamet	0	0/0	0.0	0/0	0.0	0/0	0.0	0	0	0	0	0	0	0	0
21 P. Beverley	0	0/0	0.0	0/0	0.0	0/0	0.0	0	0	0	0	0	0	0	0
23 L. Williams	0	0/0	0.0	0/0	0.0	0/0	0.0	0	0	0	0	0	0	0	0
31 M. Morris	0	0/0	0.0	0/0	0.0	0/0	0.0	0	0	0	0	0	0	0	0
40 I. Zubac	0	0/0	0.0	0/0	0.0	0/0	0.0	0	0	0	0	0	0	0	0
Total	43	7/7	100.0	5/15	33.3	14/17	82.3	1	2	3	4	0	0	4	0

Figura 4.4: Pantalla de boxscore de partido en *Boxscore for basketball*.

Para finalizar resumiremos los aspectos más destacables de esta aplicación bajo nuestra opinión.



- **Pros:**

- Interfaz intuitiva.
- Cumple con sus objetivos teniendo como usuario objetivo alguien que apenas requiere profundidad estadística o no necesita poder mostrar los datos de una forma atractiva, simplemente el almacenamiento de los mismos.

- **Contras:**

- No contiene *play by play* (seguimiento de las jugadas que se han registrado en orden cronológico).
- No controla el tiempo de partido, lo cual sería una implementación muy útil con vistas a introducir el cálculo de otras estadísticas.
- Las estadísticas generadas no son acumulables a lo largo de los partidos, ya que no podemos registrar un partido en el marco de una temporada deportiva, convendría poder observar las estadísticas de un jugador a lo largo de varios partidos, mostrándonos su total o el promedio.

4.2.2 Basketball Stats Assistant

Para la siguiente aplicación el móvil no nos ha permitido realizar capturas de pantalla, por lo que las imágenes que se mostrarán son las que hay en la [web oficial](#).

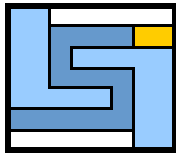
Para comenzar vamos a observar la pantalla de partido, desde la cual podemos registrar cualquier acción. En ella vemos el marcador junto al reloj de partido, el cual podemos controlar mediante los botones de *play* y *stop*. Podemos además controlar el quinteto en pista y realizar sustituciones arrastrando las camisetas desde el banquillo al jugador en pista que pretendemos sustituir.

En los botones laterales podemos consultar los tiempos muertos pedidos por cada equipo, el boxscore, las faltas o el tiempo en pista de cada jugador.

La forma de registrar las acciones es algo extraña, ya que debemos arrastrar los botones que hay en el centro de la pista (el balón que contiene un número, el símbolo de asistencia, etc) hasta el jugador que haya realizado la acción, aunque podemos entender que es una medida con el objetivo de realizar todos los registros y acciones del partido desde la misma pantalla.

Figura 4.5: Pantalla de partido en *Basketball Stats Assistant*.Figura 4.6: Estadísticas de jugador en *Basketball Stats Assistant*.

Durante el partido, al registrar un tiro, podemos elegir desde qué posición se ha realizado. Es una característica muy interesante y necesaria para poder mostrar en el futuro distintas gráficas que pueden ser de mucha utilidad para los usuarios, aunque en esta aplicación en concreto hemos encontrado varios fallos, pudiendo por ejemplo registrar un tiro de 2 puntos desde detrás de la línea de 3 puntos o viceversa, ya que el



4.2. Comparativa de aplicaciones existentes

dibujo de la pista en el que se registran los tiros es simplemente una imagen en la que añadimos un círculo en caso de anotar el tiro o una cruz en caso de fallarlo, por lo que no se está comprobando relación alguna entre el valor del tiro (2 o 3 puntos) y la zona en la que se realiza.



Figura 4.7: Registro de tiros de un jugador en *Basketball Stats Assistant*.

A continuación mostramos varias imágenes en las que podemos ver las gráficas que genera la aplicación a raíz de las acciones añadidas por el usuario, como por ejemplo las ventajas en el marcador en cada minuto del partido o los líderes estadísticos del mismo.

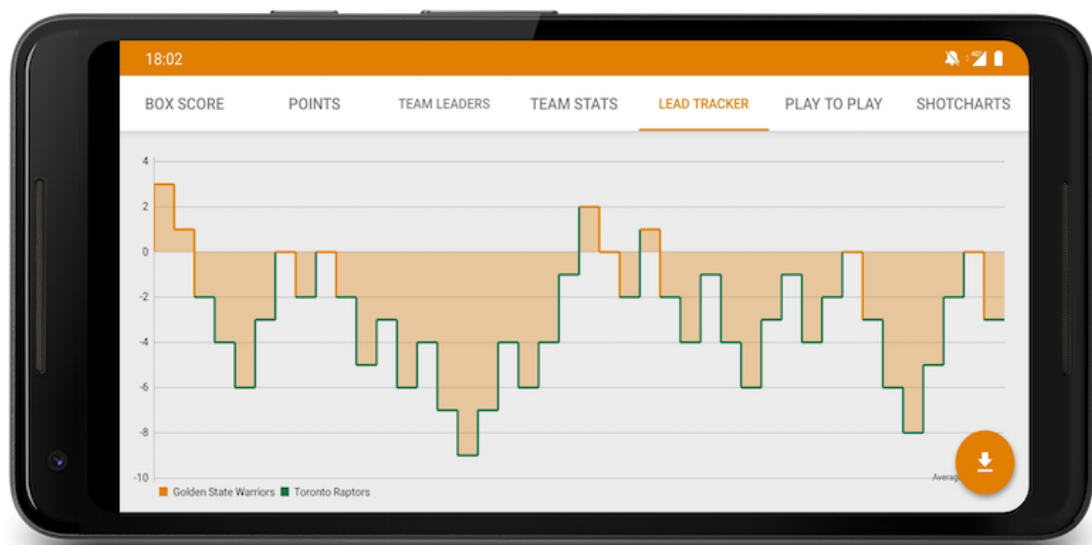
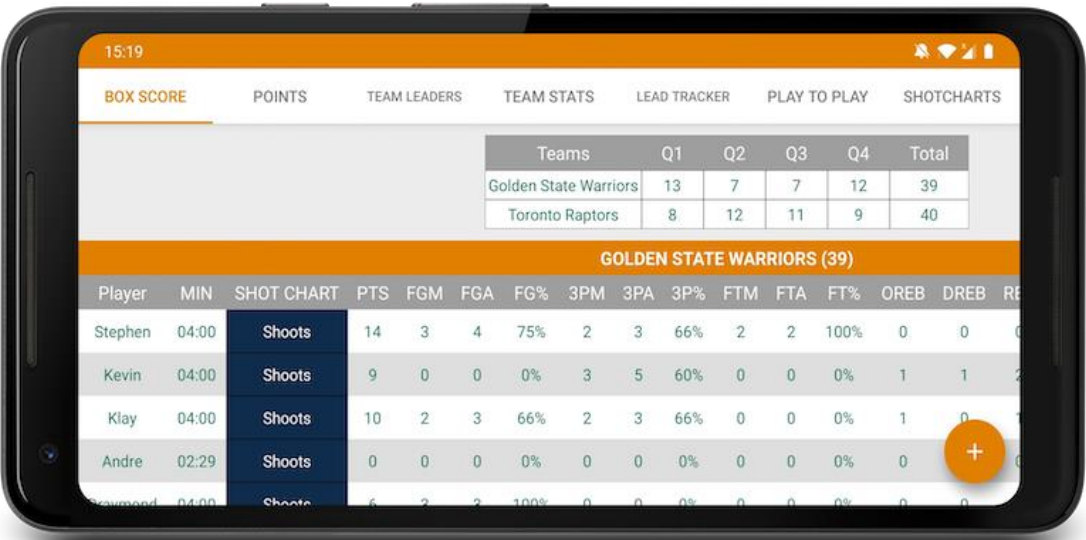


Figura 4.8: Ventajas en el marcador de un partido en *Basketball Stats Assistant*.



Figura 4.9: Líderes estadísticos del partido en *Basketball Stats Assistant*.

Para finalizar observaremos una captura del *boxscore* de un partido. Debemos añadir que, aunque en la siguiente imagen no sea del todo perceptible la optimización de esta tabla en smartphones no está del todo conseguida, ya que no nos permite cambiar el zoom predeterminado y en ocasiones, dependiendo del dispositivo, hay que hacer mucho scroll lateral para poder ver el final de la tabla.



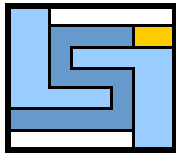


Figura 4.10: Boxscore del partido en *Basketball Stats Assistant*.

Pasamos a enumerar las características más destacables de la aplicación.

- **Pros:**
 - La funcionalidad de control del tiempo del partido resulta bastante útil.
 - Los gráficos de tiros realizados son una muy buena implementación, aunque contiene los errores anteriormente mencionados.
 - Nos permite visualizar el *play by play* del encuentro.
 - Los gráficos generados son atractivos visualmente.
- **Contras:**
 - Como ya hemos comentado, el *zoom* del *boxscore* está mal optimizado.
 - El alcance de las estadísticas presentadas es básico.
 - El registro de jugadas es algo confuso y puede inducir a errores, además de ser algo lento.

4.2.3 Breakthrough Stats App.

Ahora hablaremos sobre la que es hasta el momento la aplicación que más nos ha convencido, es una aplicación para iOS y de pago, por lo que para analizarla hemos visualizado los distintos tutoriales que se pueden encontrar en su canal oficial de YouTube, disponible bajo el nombre de BreakthroughBBall.

Para comenzar observamos un esquema de la interfaz de partido, pudiendo ver a simple vista la distribución de la pantalla y los pasos necesarios para registrar o visualizar el dato deseado.

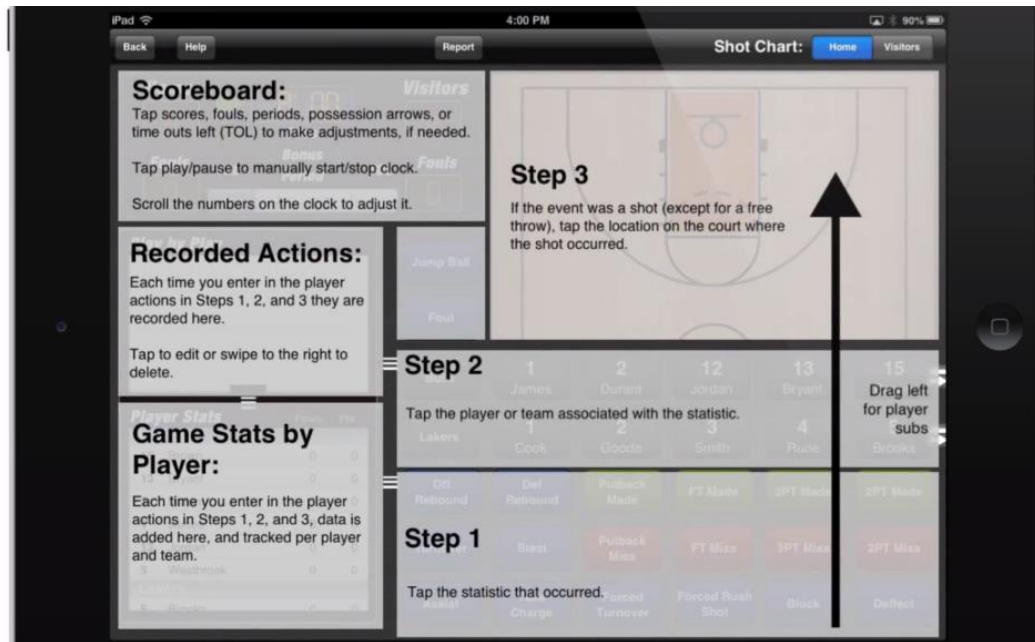
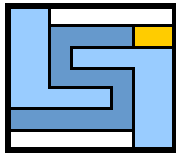


Figura 4.11: Esquema de interfaz de partido en *Breakthrough Stats App*.

En la siguiente captura podemos ver como la interfaz es muy completa y ordenada, además de atractiva. Como punto positivo podemos editar las jugadas o tiros registrados además de borrar una acción directamente desde la ventana de play by play, hecho que aunque parezca una obviedad, y como ya hemos visto, la mayoría de aplicaciones no incluyen.



Figura 4.12: Interfaz de partido en *Breakthrough Stats App*.



4.3. Necesidades y opiniones de entrenadores y jugadores

El boxscore generado no tiene ninguna característica especial en cuanto a los datos tratados aunque tampoco podemos resaltar una característica negativa, ya que cumple su cometido y además nos permite exportar la tabla a formato PDF o CSV.

Para finalizar resumimos las características principales de la aplicación.

- **Pros:**
 - Interfaz atractiva e intuitiva.
 - Boxscore muy completo y limpio visualmente.
 - Cumple su cometido en general, aunque hay que tener en cuenta que es una aplicación de pago, por lo que se debe exigir más que a las gratuitas.
- **Contras:**
 - Aunque permite el registro de la mayoría de estadísticas imaginables, como por ejemplo la zona del campo desde la que se ejecuta el tiro, luego no refleja esta cantidad de profundidad estadística en los informes generados, por lo que fuera de la limpieza y atractivo visual no se diferencia en demasía de una aplicación gratuita.

4.3 Necesidades y opiniones de entrenadores y jugadores

Una vez que tenemos una visión general de lo que proporcionan las aplicaciones en el mercado dedicadas a la recopilación de estadísticas en partidos de baloncesto es interesante conocer el punto de vista de posibles usuarios de nuestro sistema, recogiendo su opinión acerca de qué deberían aportar estas aplicaciones para que fueran útiles realmente en el día a día de un equipo.

Para ello hemos contactado con jugadores y entrenadores de baloncesto, anotando sus inquietudes, las cuales resumiremos en los siguientes puntos:

- La interfaz debe poseer claridad y sencillez, ya que no todos los posibles usuarios de nuestra plataforma están acostumbrados a este tipo de aplicaciones o a sistemas informáticos en general.
- Se debe proveer la posibilidad de consultar datos a nivel de jugador para intentar mejorar aspectos débiles de su juego o explotar los puntos fuertes, con vistas al desarrollo deportivo del mismo.

- Añadir la posibilidad de consultar datos estadísticos de los equipos, tanto contrarios como propio, para que sea una herramienta que ayude a enfocar el trabajo técnico y/o táctico de la entidad.
- Trabajar la interfaz desde un aspecto visual para hacerla atractiva, pudiendo añadir fotos de jugadores y equipos, ya que esto puede animar a más usuarios a usar nuestro proyecto.
- Los datos mostrados deben ser fáciles de comprender, trasladando cierta complejidad estadística a un lenguaje o aspecto manejable por cualquier usuario medio con unos mínimos conocimientos sobre la materia.
- Es importante poder acceder a la plataforma desde distintos dispositivos, ya que por ejemplo si el encargado de recopilar las estadísticas se encuentra visualizando en directo el partido desde el banquillo del equipo no podrá hacerlo desde un ordenador de sobremesa o portátil, por lo que sería conveniente optimizar lo máximo posible la plataforma para hacerla accesible desde ordenadores o tablets, cuyo uso está bastante extendido en estas situaciones en el deporte de alto nivel.

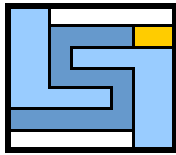
4.4 Conclusiones relativas al estado del arte

Para finalizar este apartado nos gustaría anotar nuestras conclusiones con lo respectivo al estado actual de las aplicaciones que nos ofrece el mercado, teniendo en cuenta además la opinión de las personas consultadas, dando como resultado una serie de ideas iniciales o línea a seguir para con nuestra aplicación

El sistema que realicemos debería ser una aplicación web, ya que, aunque en un principio se barajó la idea de desarrollar el proyecto exclusivamente como aplicación para tablets, la posibilidad de utilizar nuestro sistema desde varios dispositivos hace que la inclusión de usuarios sea mayor y cada uno pueda elegir en cada momento qué opción quiere utilizar.

Para ello intentaremos que la interfaz tenga un aspecto trabajado y que se visualice de forma correcta en dispositivos con distintas resoluciones de pantalla.

Como hemos visto en la comparativa de aplicaciones, aunque hay varias que cumplen un objetivo básico, ninguna de ellas provee el usuario, por ejemplo, de estadísticas acumuladas a lo largo de varios partidos o una temporada del jugador o



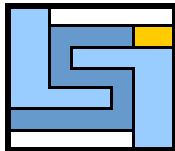
equipo, teniendo que calcular estos datos de forma manual, cuya labor puede ser ardua en el caso de ciertas estadísticas. Para solucionarlo, y visto el interés de las personas encuestadas, deberíamos incluir en nuestro sistema la posibilidad de visualizar estos datos fácilmente, ahorrando al usuario el cálculo de los mismos, por lo que deberemos invertir gran cantidad de horas en el almacenamiento y diseño del cómputo de estadísticas en nuestra base de datos.

Como último punto insistiremos en la visualización atractiva de la información a mostrar, para lo cual deberemos prestar especial atención a la representación en tablas de distintos tipos de los datos deseados.

Capítulo 5

5. Elicitación de requisitos

En este capítulo proporcionaremos una visión detallada del sistema web que realizaremos.



5.1 Introducción

El proyecto a realizar, como hemos comentado con anterioridad, consiste en desarrollar un portal web que ayude a clubes de baloncesto base a poseer un registro estadístico de sus partidos. Podemos dividir el proyecto en dos servicios:

1. **Recopilación de las estadísticas de los partidos:** el principal objetivo del sistema es facilitar la recopilación de las distintas acciones que se producen en un partido de baloncesto por los jugadores, y por ende, los equipos.
2. **Visualización y análisis de los datos:** a partir de los datos recopilados, mostraremos una serie de gráficas y herramientas que ayuden al usuario (ya sea jugador o entrenador) a analizar lo ocurrido con facilidad, pudiendo así mejorar el rendimiento deportivo individual y grupal.

Como conclusión, nuestro objetivo es ofrecer una plataforma de información estadística en lo concerniente al baloncesto que posea calidad en el diseño y profundidad de contenido útil.

5.2 Participantes en el proyecto

Participante	Jairo Fernández Carmona
Organización	Freelance
Rol	Desarrollador
Es desarrollador	Sí
Es cliente	No
Es usuario	No
Comentarios	Alumno de Ingeniería Informática – Tecnologías Informáticas por la Escuela Técnica Superior de Ingeniería Informática de la Universidad de Sevilla, el cual realizará el desarrollo completo del sistema.

Cuadro 5.1: Participante: Jairo Fernández Carmona

Participante	José Antonio Troyano Jiménez
Organización	Departamento de Lenguajes y Sistemas Informáticos
Rol	Tutor
Es desarrollador	No
Es cliente	No
Es usuario	No
Comentarios	Profesor del Departamento de Lenguajes y Sistemas Informáticos por la Universidad de Sevilla, encargado de la supervisión y control del desarrollo completo del sistema.

Cuadro 5.2: Participante: José Antonio Troyano Jiménez

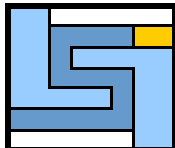
Organización	Departamento de Lenguajes y Sistemas Informáticos
Dirección	Avda. Reina Mercedes s/n. 41012 Sevilla
Teléfono	954 557 139
Fax	954 557 139
Comentarios	Email: lsi@lsi.us.es Web: www.lsi.us.es

Cuadro 5.3: Organización: Departamento de Lenguajes y Sistemas Informáticos

5.3 Descripción del Sistema Actual

En la actualidad existe una gran variedad de aplicaciones dedicadas a la recopilación de estadísticas de baloncesto, siendo las más complejas aplicaciones de pago. En la mayoría de estas aplicaciones la funcionalidad en lo referente a la recolección de estos datos es muy amplia, pero en cuanto a la visualización de estos datos se limita a mostrar el boxscore tradicional.

Para crear un añadido con respecto a estas aplicaciones pretendemos implementar varias herramientas que creen una experiencia útil en el apartado de análisis y visualización de los datos, intentando acercar a todo aquel que use nuestro sistema al



uso de la llamada estadística avanzada, que tan al día está ya en clubes profesionales o semiprofesionales.

5.4 Objetivos del sistema

Nuestro objetivo fundamental, como ya hemos mencionado, es desarrollar una aplicación que facilite la recopilación de estadísticas y contenga un apartado medianamente extenso para el análisis de estos datos.

Con tal finalidad hemos desglosado el objetivo general en los objetivos indicados a continuación.

OBJ-0001	Proporcionar interfaz de recopilación de estadísticas de un partido
Versión	1.0 (12/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Descripción	El sistema deberá ofrecer una interfaz para recopilar datos de un partido en tiempo real.
Subobjetivos	<p>[OBJ-0007] Generar play by play: El sistema deberá guardar los registros de cada una de las jugadas realizadas en los partidos, para poder visualizarlas durante el mismo y generar estadísticas a posteriori.</p> <p>[OBJ-0008] Generar boxscore: El sistema deberá crear un boxscore que se pueda visualizar durante el partido o consultar en cualquier momento posterior al mismo.</p> <p>[OBJ-0009] Calcular estadísticas avanzadas: El sistema deberá calcular una serie de estadísticas que serán generadas a partir de la recolección de datos en la interfaz y guardadas en la base de datos para su posterior consulta.</p>
Importancia	Vital.
Urgencia	Inmediatamente.
Estado	Validado.
Estabilidad	Alta.

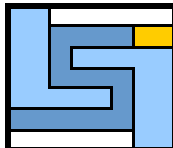
Comentarios	Esta interfaz permitirá registrar las acciones del partido de forma sencilla e intuitiva, seleccionando una serie de parámetros que nos permitirá calcular los datos pertinentes y su visualización.
--------------------	--

Cuadro 5.4: Objetivos relacionados con la interfaz de recopilación estadística de un partido.

OBJ-0002	Mostrar análisis estadístico de jugadores y equipos
Versión	1.0 (15/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Descripción	El sistema deberá implementar análisis estadísticos de jugadores y equipos, mediante gráficos que permitan su correcta visualización en los perfiles o fichas de los mismos.
Subobjetivos	[OBJ-0005] Proporcionar información sobre un Jugador o Equipo: El sistema deberá implementar fichas de jugadores y equipos en las que se mostrará la información general y un resumen de algunos parámetros considerados importantes en el ámbito estadístico relacionado con el baloncesto.
Importancia	Vital.
Urgencia	Inmediatamente.
Estado	Validado.
Estabilidad	Alta.
Comentarios	Con este objetivo pretendemos mostrar un resumen gráfico de las estadísticas avanzadas que consideremos más importantes para el usuario, distinguiendo entre equipos y jugadores.

Cuadro 5.5: Objetivos relacionados con mostrar análisis estadístico de jugadores y equipos.

OBJ-0003	Proporcionar herramientas para el análisis estadístico
Versión	1.0 (15/06/2020)
Autores	Jairo Fernández Carmona



Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Descripción	El sistema deberá implementar una serie de herramientas que permitan analizar mediante distintos parámetros estadísticos a los jugadores o equipos elegidos.
Subobjetivos	[OBJ-0004] Proporcionar eje de coordenadas para comparación estadística: El sistema deberá implementar un eje de coordenadas (X e Y), que ordene percentilmente a jugadores o equipos, mediante unos parámetros elegidos, seleccionando una estadística para el eje X y otra para el eje Y. [OBJ-0006] Proporcionar herramienta para comparación directa entre jugadores o equipos: El sistema deberá implementar una herramienta que permita comparar dos jugadores o equipos “cara a cara”, reuniendo todas las estadísticas contempladas y visualizándolas en la comparativa.
Importancia	Importante.
Urgencia	Inmediatamente.
Estado	Validado.
Estabilidad	Alta.
Comentarios	Elegiremos una serie de parámetros mediante los cuales clasificaremos a los equipos y jugadores para poder observar sus fortalezas y debilidades frente a sus rivales. Por ejemplo, en el caso de jugadores, podemos querer observar a jugadores ordenados mediante su porcentaje de tiros libres o puntos por tiro de campo.

Cuadro 5.6: Objetivos relacionados con proporcionar herramientas para el análisis estadístico.

OBJ-0010	Permitir cuentas de usuario
Versión	1.0 (15/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez

Descripción	El sistema deberá permitir el acceso únicamente a usuarios registrados y limitar la información que el usuario puede editar, consultar o registrar dependiendo de su rol.
Subobjetivos	<p>[OBJ-0011] Establecer roles de usuario: El sistema deberá ofrecer a los usuarios acciones para crear, editar o eliminar perfiles de usuario, tendremos los siguientes roles:</p> <ul style="list-style-type: none"> • Administrador: podrá gestionar la totalidad del sistema. • Empleado: no puede realizar acciones concernientes al control de usuarios, aunque tiene acceso total al resto del sistema, por lo que puede crear, editar o eliminar cualquier acción correspondiente a jugadores, equipos, partidos, etc. <p>[OBJ-0012] Control de usuarios: El sistema deberá ofrecer mecanismos a los administradores para controlar y gestionar a los usuarios que acceden a la plataforma.</p>
Importancia	Vital.
Urgencia	Inmediatamente.
Estado	Validado.
Estabilidad	Alta.
Comentarios	Implantaremos un sistema de seguridad para nuestro portal web y API para controlar el acceso, limitando las operaciones permitidas únicamente a los usuarios registrados por un administrador.

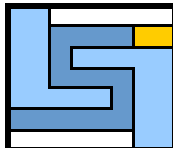
Cuadro 5.7: Objetivos relacionados con permitir cuentas de usuario.

5.5 Catálogo de requisitos del sistema

5.5.1 Requisitos de información

Para describir los requisitos de información los dividiremos en una serie de apartados para lograr una mayor organización.

Jugadores: Indicamos a continuación la información que debemos mostrar sobre los distintos jugadores.



IRQ-0001	Información sobre jugadores
Versión	1.0 (15/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	<p>[OBJ-0002] Mostrar análisis estadístico de jugadores y equipos.</p> <p>[OBJ-0005] Proporcionar información sobre un jugador o equipo.</p> <p>[OBJ-0001] Proporcionar interfaz de recopilación de estadísticas de un partido.</p> <p>[OBJ-0006] Proporcionar herramienta para comparación directa entre jugadores o equipos.</p> <p>[OBJ-0004] Proporcionar eje de coordenadas para comparación estadística.</p> <p>[OBJ-0009] Calcular estadísticas avanzadas.</p> <p>[OBJ-0003] Proporcionar herramientas para el análisis estadístico.</p>
Descripción	El sistema deberá almacenar la información correspondiente a los jugadores. En concreto:
Datos específicos	<ul style="list-style-type: none">• Identificador.• Nombre.• Apellidos.• Foto: imagen del jugador.• Edad.• Equipo.• Dorsal.• Posición.• Estadísticas: guardaremos en la base de datos una serie de estadísticas del jugador que habremos generado previamente en cada partido jugado, algunas de ellas las mostraremos en el perfil del jugador.• Teléfono.• Email.
Importancia	Vital.
Urgencia	Inmediatamente.
Estado	Validado.
Estabilidad	Alta.

Comentarios	Esta información permanecerá siempre en el sistema, ya que nos puede interesar consultarla en futuras temporadas.
--------------------	---

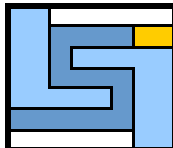
Cuadro 5.8: IRQ-0001: Requisitos de Información sobre Jugadores.

CRQ-0001	Unicidad de Identificador de jugador
Versión	1.0 (16/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	Ninguna.
Descripción	La información almacenada por el sistema deberá satisfacer la siguiente restricción: los identificadores de Jugador deben ser únicos, no pueden existir dos jugadores distintos con el mismo identificador. Un jugador se identifica mediante su nombre, apellidos y fecha de nacimiento, por lo que esta combinación no puede coincidir para dos o más jugadores.
Importancia	Vital.
Urgencia	Inmediatamente.
Estado	Validado.
Estabilidad	Alta.

Cuadro 5.9: CRQ-0001: Restricción sobre Unicidad de Identificador de Jugador.

Clubs: indicamos a continuación la información que debemos almacenar sobre los distintos clubs.

IRQ-0002	Información sobre clubs
Versión	1.0 (15/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez



Dependencias	[OBJ-0005] Proporcionar información sobre un jugador o equipo. [OBJ-0001] Proporcionar interfaz de recopilación de estadísticas de un partido.
Descripción	El sistema deberá almacenar la información correspondiente a los clubs. En concreto:
Datos específicos	<ul style="list-style-type: none">• Identificador.• Nombre.• Ciudad.• Pabellón: nombre del pabellón en el que compite como local.• Equipos: una lista en la que podremos ver a los equipos componentes del club.• Escudo del club: imagen del escudo del club.• Teléfono: teléfono de contacto del club.• Email: email de contacto.
Importancia	Vital.
Urgencia	Inmediatamente.
Estado	Validado.
Estabilidad	Alta.
Comentarios	Esta información permanecerá siempre en el sistema, ya que nos puede interesar consultarla en futuras temporadas.

Cuadro 5.10: IRQ-0002: Requisitos de Información sobre Clubs.

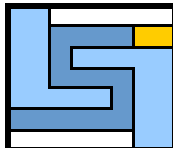
CRQ-0002	Unicidad de Identificador de club
Versión	1.0 (16/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	Ninguna.

Descripción	<p>La información almacenada por el sistema deberá satisfacer la siguiente restricción: los identificadores de Club deben ser únicos, no pueden existir dos clubs distintos con el mismo identificador.</p> <p>Cada club tiene relacionado un nombre y una ciudad/localización, mediante estos datos diferenciamos unos clubes de otros, por lo que la combinación no puede estar repetida, no puede haber dos clubes con el mismo nombre que pertenezcan a la misma localización.</p>
Importancia	Vital.
Urgencia	Inmediatamente.
Estado	Validado.
Estabilidad	Alta.

Cuadro 5.11: CRQ-0002: Restricción sobre Unicidad de Identificador de Club.

Equipos: indicamos a continuación la información que debemos almacenar sobre los equipos.

IRQ-0003	Información sobre equipos
Versión	1.0 (15/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	<p>[OBJ-0002] Mostrar análisis estadístico de jugadores y equipos.</p> <p>[OBJ-0005] Proporcionar información sobre un jugador o equipo.</p> <p>[OBJ-0001] Proporcionar interfaz de recopilación de estadísticas de un partido.</p> <p>[OBJ-0006] Proporcionar herramienta para comparación directa entre jugadores o equipos.</p> <p>[OBJ-0004] Proporcionar eje de coordenadas para comparación estadística.</p> <p>[OBJ-0009] Calcular estadísticas avanzadas.</p>
Descripción	El sistema deberá almacenar la información correspondiente a los equipos. En concreto:



Datos específicos	<ul style="list-style-type: none">• Identificador.• Temporada. (Ya que los equipos cambian de integrantes cada año debemos guardar la temporada en la que el equipo compitió).• Entrenador.• Cuerpo técnico. (Opcional).• Plantilla: lista de jugadores que componen el equipo.• Liga: liga en la que compete el equipo.• Estadísticas: guardaremos en la base de datos una serie de estadísticas de los distintos equipos que habremos generado previamente con cada partido jugado, algunas de ellas las mostraremos en el perfil del equipo.
Importancia	Vital.
Urgencia	Inmediatamente.
Estado	Validado.
Estabilidad	Alta.
Comentarios	Ya que los integrantes del equipo cambian cada año es interesante guardar un registro histórico de los equipos para poder comparar unos con otros.

Cuadro 5.12: IRQ-0003: Requisitos de Información sobre Partidos.

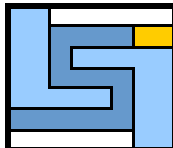
CRQ-0003	Unicidad de Identificador de equipo
Versión	1.0 (16/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	Ninguna.
Descripción	<p>La información almacenada por el sistema deberá satisfacer la siguiente restricción: los identificadores de Equipo deben ser únicos, no pueden existir dos equipos distintos con el mismo identificador.</p> <p>Cada equipo tiene relacionado una temporada, un club al que pertenece y una liga en la que compete. En el caso en el que varíe algunos de los tres datos se consideraría un equipo distinto.</p>
Importancia	Vital.

Urgencia	Inmediatamente.
Estado	Validado.
Estabilidad	Alta.

Cuadro 5.13: CRQ-0003: Restricción sobre Unicidad de Identificador de Equipo.

Partidos: indicamos la información que debemos almacenar sobre los partidos disputados.

IRQ-0005	Información sobre partidos
Versión	1.0 (15/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	<p>[OBJ-0008] Generar boxscore.</p> <p>[OBJ-0007] Generar play by play.</p> <p>[OBJ-0001] Proporcionar interfaz de recopilación de estadísticas de un partido.</p> <p>[OBJ-0003] Proporcionar herramientas para el análisis estadístico.</p>
Descripción	El sistema deberá almacenar la información correspondiente a los partidos. En concreto:
Datos específicos	<ul style="list-style-type: none"> • Identificador. • Fecha. • Liga: competición a la que pertenece el encuentro. • Equipo local: identificador del equipo local. • Equipo visitante: identificador del equipo visitante. • Anotación equipo local: puntos anotados por el equipo local. • Anotación equipo visitante: puntos anotados por el equipo visitante. • Boxscore: será generado a partir de las estadísticas recopiladas del partido. • Play by play: registro de jugadas ordenadas cronológicamente.
Importancia	Vital.



Urgencia	Inmediatamente.
Estado	Validado.
Estabilidad	Alta.
Comentarios	Ninguno.

Cuadro 5.14: IRQ-0005: Requisitos de Información sobre Equipos.

CRQ-0005	Unicidad de Identificador de partido
Versión	1.0 (16/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	Ninguna.
Descripción	La información almacenada por el sistema deberá satisfacer la siguiente restricción: los identificadores de Partido deben ser únicos, no pueden existir dos partidos distintos con el mismo identificador. Podemos distinguir dos partidos mediante los equipos que lo componen y la fecha en la que se realiza. Dos equipos específicos no pueden disputar un partido en el mismo día.
Importancia	Vital.
Urgencia	Inmediatamente.
Estado	Validado.
Estabilidad	Alta.

Cuadro 5.15: CRQ-0005: Restricción sobre Unicidad de Identificador de Partido.

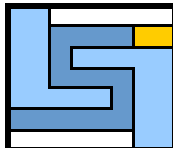
Usuarios: indicamos a continuación la información a almacenar sobre los usuarios de la plataforma.

IRQ-0006	Información sobre usuarios
Versión	1.0 (15/06/2020)

Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	[OBJ-0010] Permitir cuentas de usuarios. [OBJ-0011] Establecer roles de usuario. [OBJ-0012] Control de usuarios.
Descripción	El sistema deberá almacenar la información correspondiente a los usuarios. En concreto:
Datos específicos	<ul style="list-style-type: none"> • Identificador. • Nombre. • Apellidos. • Contraseña. • Email. • Teléfono. • Rol: debe pertenecer a uno de los roles especificados anteriormente. • Club: entidad a la que pertenece.
Importancia	Vital.
Urgencia	Inmediatamente.
Estado	Validado.
Estabilidad	Alta.
Comentarios	Ninguno.

Cuadro 5.16: IRQ-0006: Requisitos de Información sobre Usuarios.

CRQ-0006	Unicidad de Identificador de usuario
Versión	1.0 (16/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	Ninguna.



Descripción	<p>La información almacenada por el sistema deberá satisfacer la siguiente restricción: los identificadores de Usuario deben ser únicos, no pueden existir dos usuarios distintos con el mismo identificador.</p> <p>Distinguimos los usuarios mediante su email, no puede haber más de una cuenta de usuario asociada a un mismo email.</p>
Importancia	Vital.
Urgencia	Inmediatamente.
Estado	Validado.
Estabilidad	Alta.

Cuadro 5.17: CRQ-0006: Restricción sobre Unicidad de Identificador de Usuario.

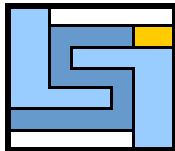
Ligas: indicamos a continuación la información a almacenar sobre las ligas o competiciones en las que se disputan los partidos y a la que pertenecen los equipos.

IRQ-0006	Información sobre ligas
Versión	1.0 (16/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	<p>[OBJ-0003] Proporcionar herramientas para el análisis estadístico.</p> <p>[OBJ-0004] Proporcionar eje de coordenadas para comparación estadística.</p> <p>[OBJ-0009] Calcular estadísticas avanzadas.</p> <p>[OBJ-0002] Mostrar análisis estadístico de jugadores y equipos.</p>
Descripción	El sistema deberá almacenar la información correspondiente a las ligas. En concreto:

Datos específicos	<ul style="list-style-type: none"> • Identificador. • Nombre. • País/Localización. • Número de partes (denominadas comúnmente cuartos). • Duración de cada parte. • Tiempo de posesión. • Logo.
Importancia	Vital.
Urgencia	Inmediatamente.
Estado	Validado.
Estabilidad	Alta.
Comentarios	Ninguno.

Cuadro 5.18: IRQ-0007: Requisitos de Información sobre Ligas.

CRQ-0006	Unicidad de Identificador de liga
Versión	1.0 (16/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	Ninguna.
Descripción	<p>La información almacenada por el sistema deberá satisfacer la siguiente restricción: los identificadores de Liga deben ser únicos, no pueden existir dos usuarios distintos con el mismo identificador.</p> <p>Una liga se considera distinta de otra cuando los nombres no coinciden.</p>
Importancia	Vital.
Urgencia	Inmediatamente.
Estado	Validado.
Estabilidad	Alta.



Cuadro 5.19: CRQ-0007: Restricción sobre Unicidad de Identificador de Liga.

5.5.2 Requisitos funcionales

Esta sección contiene la lista de requisitos funcionales, los cuales especificaremos a partir de los casos de uso identificados.

Dividiremos esta sección en subsecciones para organizar mejor la información y facilitar su comprensión.

Comenzaremos definiendo los actores.

5.5.2.1 Actores

ACT-0001	Administrador
Versión	1.0 (16/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Descripción	Este actor representa a los usuarios del sistema que abarcan desde el registro de estadísticas hasta la moderación de los usuarios, en resumen, tienen un control total sobre el portal web. Una de sus principales funciones será registrar usuarios.
Comentarios	Como ya hemos especificado, estos usuarios tienen control total sobre el sistema, por lo que debe ser un tipo de usuario poco numeroso y con conocimiento sobre el portal.

Cuadro 5.20: ACT-0001: Actor Administrador.

ACT-0002	Empleado
Versión	1.0 (16/06/2020)
Autores	Jairo Fernández Carmona

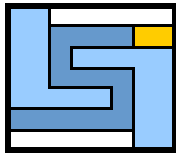
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Descripción	Este actor representa a los usuarios cuyo objetivo es registrar y generar datos, sin tener acceso al control de usuarios.
Comentarios	Son registrados por los administradores.

Cuadro 5.21: ACT-0002: Actor Empleado.

ACT-0003	Usuario
Versión	1.0 (17/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Descripción	Este actor representa al grupo de usuarios (tanto administradores como empleados) que interactúan con el sistema.
Comentarios	Utilizamos este tipo de actor como simplificación del modelo, utilizando una jerarquía de actores, cuyo valor dependerá de los usuarios que participen en cada caso de uso.

Cuadro 5.22: ACT-0003: Actor Usuario.

5.5.2.2 Casos de uso



A continuación mostramos los diferentes diagramas de casos de uso para nuestro sistema. Comenzamos definiendo los distintos subsistemas.



Figura 5.1: Diagrama de subsistemas.

Ahora mostraremos los diferentes diagramas de casos de uso asociados a cada subsistema.

- **Gestión de Usuarios**

En este subsistema realizamos las consultas, altas, bajas o ediciones de usuarios, así como la identificación del mismo, para interactuar con nuestro portal web.

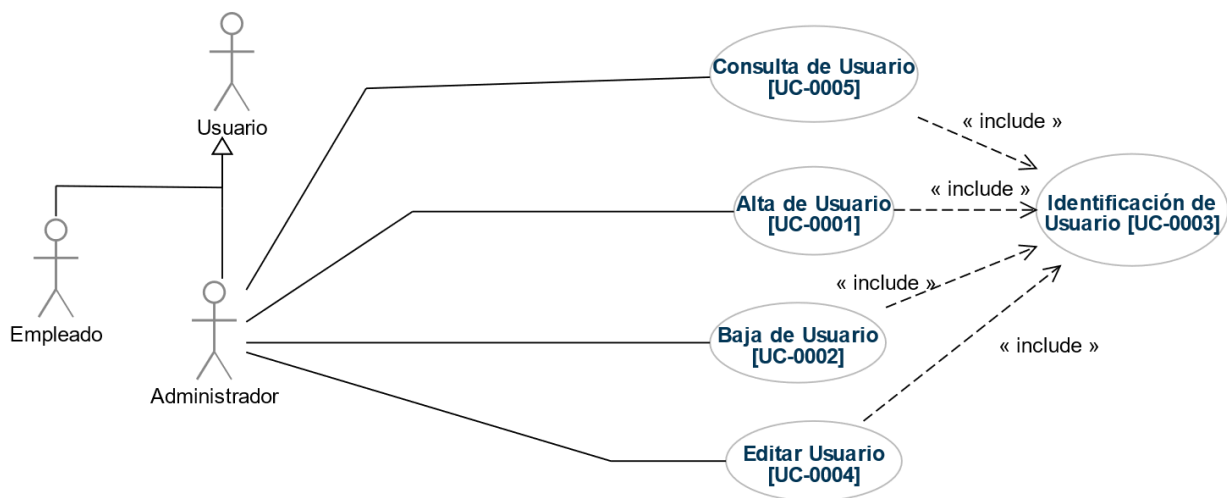
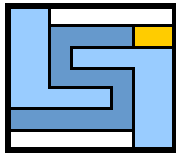


Figura 5.2: Diagrama de Casos del Subsistema Gestión de Usuarios.

Hemos representado los diferentes tipos de usuarios mediante una jerarquía, en la cual el tipo *Usuario* representa las funciones comunes a todos los roles.

Los usuarios con el rol de *Administrador* deberían poseer conocimientos informáticos, ya que además de ser los encargados del control de usuarios deberían trabajar en el mantenimiento y actualización de la API y la base de datos.

Solo los usuarios registrados podrán realizar peticiones a la base de datos, ya sea mediante la interfaz web o la petición HTTP a nuestra API. Para controlar estas peticiones manejamos la creación de tokens de forma automática cuando un usuario se logea en la web.



• Gestión de Partidos

En este subsistema podremos crear, editar, eliminar o consultar partidos, al crearlos y añadir las correspondientes jugadas se generarán las estadísticas de partido y temporada correspondientes para los jugadores y equipos implicados.

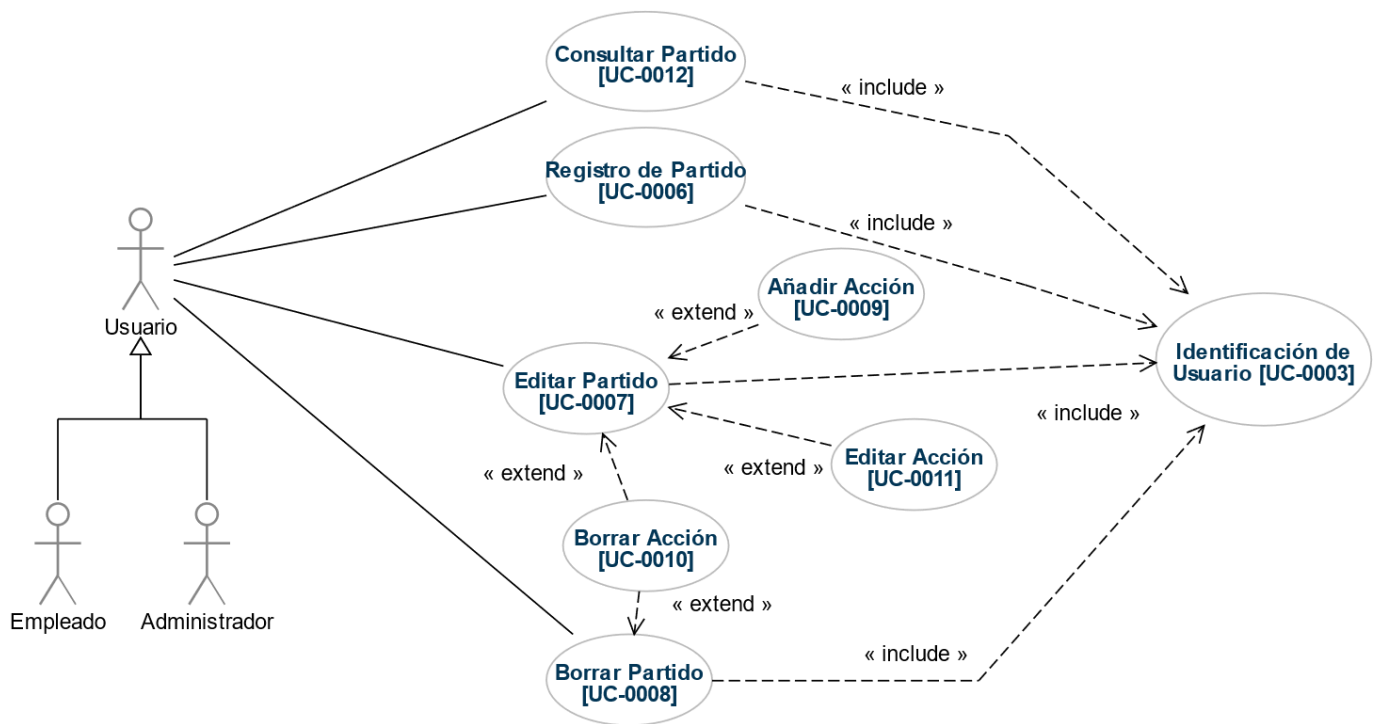


Figura 5.3: Diagrama de Casos del Subsistema Gestión de Partidos.

Como podemos observar cualquier usuario que acceda a la plataforma puede consultar, registrar, editar o borrar partidos, así como las acciones deportivas correspondientes, ya que la idea del portal es trabajar de forma conjunta en la recopilación de datos.

- **Gestión de Clubs**

Este subsistema es el encargado de lo concerniente a la creación, consulta, edición y borrado de los clubs.

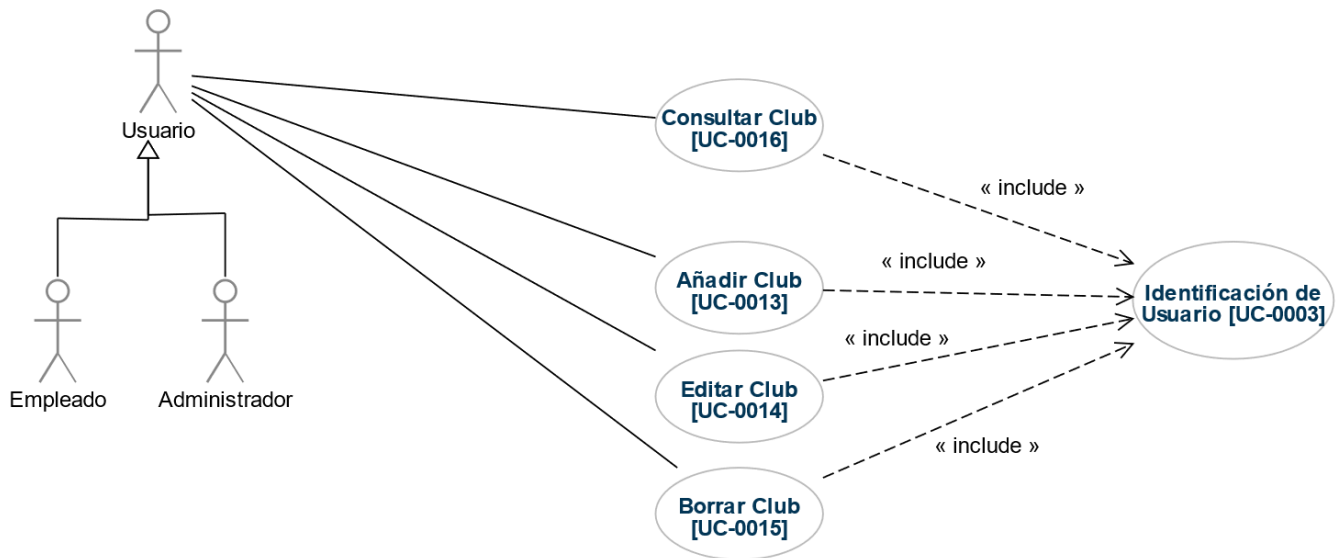
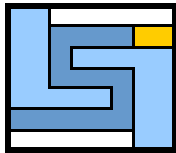


Figura 5.4: Diagrama de Casos del Subsistema Gestión de Clubs.

Observamos que este subsistema tiene unos permisos definidos para los roles de usuarios equivalente al área de Gestión de Partidos, ya que cualquier usuario posee todos los permisos relacionados con el subsistema.

La creación de clubs debe ser uno de los primeros pasos a realizar en nuestra plataforma, ya que todos los equipos que queramos registrar deben pertenecer a un club, por lo tanto hasta que no creemos un club no podremos crear un equipo.



• Gestión de Equipos

En el siguiente subsistema podemos crear, editar, borrar o consultar la información relacionada a los equipos.

La creación de equipos es un apartado muy importante, ya que las estadísticas que se mostrarán serán a nivel de equipo y no de club. Ya explicamos que los clubes están compuestos por equipos, que serán distintos cada temporada, aunque se mantengan en la misma categoría.

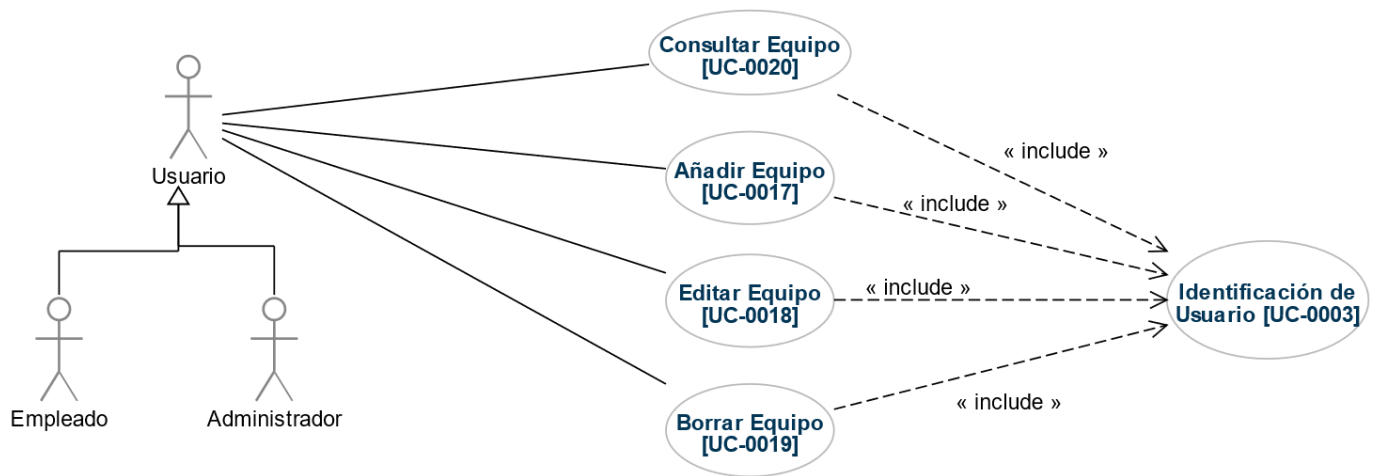


Figura 5.5: Diagrama de Casos del Subsistema Gestión de Equipos.

Este subsistema, al igual que los dos anteriores, permite a cualquier usuario realizar cualquier acción relacionada con los equipos.

Al igual que antes especificamos que la creación de clubs debería ser un paso prioritario a la hora de registrar equipos en nuestro portal, ya que es el paso predecesor, la creación de equipos es igual de importante con vistas a añadir jugadores a nuestra base de datos.

- **Gestión de Ligas**

Mediante la creación de ligas en el sistema podremos registrar equipos que participen en esas ligas.

Además tomaremos información de las ligas a la hora de comenzar un partido, como puede ser la duración de los cuartos.

A continuación observamos las acciones que se pueden realizar en el subsistema.

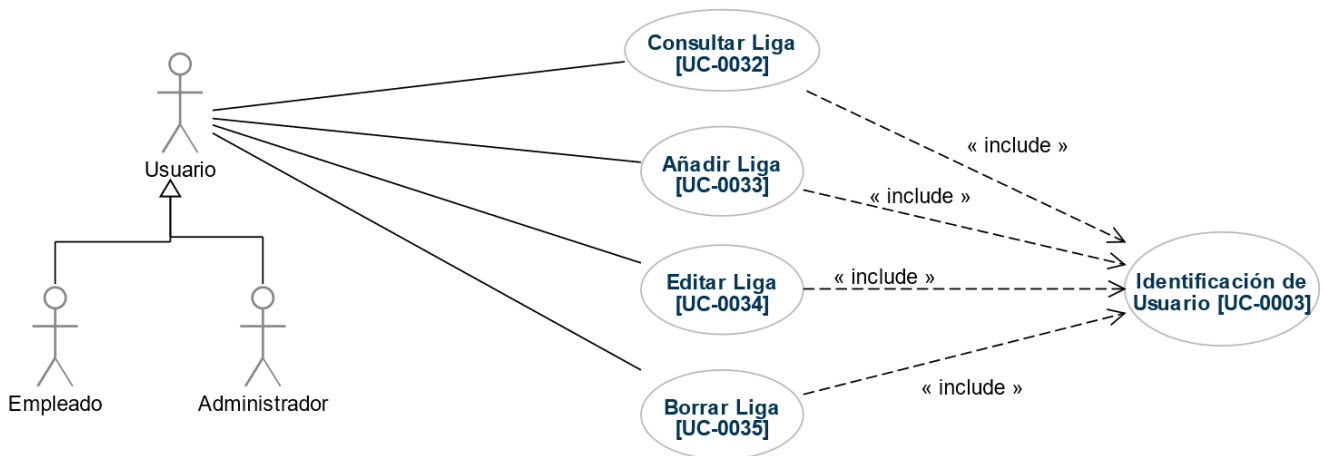
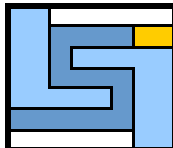


Figura 5.6: Diagrama de Casos del Subsistema Gestión de Ligas.

Cualquier usuario del sistema, como vemos en el diagrama, puede realizar las acciones relacionadas con el tratamiento de ligas.

En todos los subsistemas anteriores hemos visto la importancia del orden de creación de datos, ya que para por ejemplo para registrar un equipo debe estar registrado con anterioridad el club al que pertenece, en este caso la liga es otro apartado a tener en cuenta, ya que también será necesario a la hora de registrar un equipo.



En adelante presentaremos los casos de uso identificados y nombrados anteriormente, en plantilla con tal función.

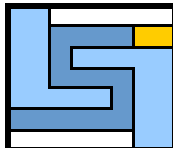
Comenzaremos con el subsistema de Gestión de Usuarios.

UC-0001	Alta de Usuario	
Versión	1.0 (17/06/2020)	
Autores	Jairo Fernández Carmona	
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez	
Dependencias	[IRQ-0006] Información sobre Usuarios. [OBJ-0010] Información sobre Usuarios. [OBJ-0011] Establecer Roles de Usuarios. [OBJ-0012] Control de Usuarios.	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando un usuario (con rol Administrador) desee registrar un nuevo usuario, otorgándole acceso al sistema y ciertos permisos, definidos por su rol.	
Precondición	El usuario que desea crear una nueva cuenta de usuario es un usuario registrado con los permisos necesarios (rol de Administrador). Recordamos que los permisos relacionados al alta de usuarios son: <ul style="list-style-type: none">• Administrador: puede dar de altas a otros administradores o empleados.• Empleado: no puede dar de alta cuentas de usuario. Para comprobar que el usuario requiere de los permisos necesarios se habrá ejecutado con anterioridad al inicio de la acción el UC-0003.	
Secuencia normal	Paso	Acción
	1	El actor Administrador (ACT-0001) solicita al sistema comenzar el proceso de registro de un nuevo usuario.
	2	El sistema solicita al usuario administrador los datos necesarios para registrar al nuevo usuario.
	3	El actor Administrador (ACT-0001) proporciona los datos introducidos.

	4	El sistema comprueba la validez de los datos introducidos.
	5	El sistema almacena la información del nuevo usuario en la base de datos.
	6	El sistema indica que el registro se ha realizado correctamente.
Postcondición	El usuario se ha registrado con éxito en el sistema y puede acceder al mismo.	
Excepciones	Paso	Acción
	3	El sistema pedirá que se introduzca el rol del nuevo usuario, a continuación este caso de uso continua.
	6	Si se produce algún error durante el proceso el sistema informa al usuario del error pertinente, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

Cuadro 5.23: UC-0001: Alta de Usuario.

UC-0002	Baja de Usuario
Versión	1.0 (17/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	[IRQ-0006] Información sobre Usuarios. [OBJ-0010] Información sobre Usuarios. [OBJ-0011] Establecer Roles de Usuarios. [OBJ-0012] Control de Usuarios.
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando un usuario (con rol Administrador) desee dar de baja a un usuario.



Precondición	<p>El usuario que desea borrar una cuenta de usuario es un usuario registrado con los permisos necesarios (rol de Administrador).</p> <p>Recordamos que los permisos relacionados al borrado de usuarios son:</p> <ul style="list-style-type: none">• Administrador: puede dar de baja a otros administradores o empleados.• Empleado: no puede dar de baja cuentas de usuario. <p>Para comprobar que el usuario requiere de los permisos necesarios se habrá ejecutado con anterioridad al inicio de la acción el UC-0003.</p>	
Secuencia normal	Paso	Acción
	1	El actor Administrador (ACT-0001) solicita al sistema comenzar el proceso de borrado de un usuario.
	2	El sistema elimina al usuario correspondiente.
	3	El sistema indica que el borrado se ha realizado correctamente.
Postcondición	El usuario se ha eliminado del sistema con éxito.	
Excepciones	Paso	Acción
	1	El actor iniciará la petición seleccionando al usuario elegido desde un listado.
	3	Si se produce algún error durante el proceso el sistema informa al usuario del error pertinente, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

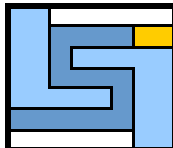
Cuadro 5.24: UC-0002: Baja de Usuario.

UC-0003	Identificación de Usuario
Versión	1.0 (17/06/2020)

Autores	Jairo Fernández Carmona	
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez	
Dependencias	[IRQ-0006] Información sobre Usuarios. [OBJ-0010] Información sobre Usuarios. [OBJ-0011] Establecer Roles de Usuarios. [OBJ-0012] Control de Usuarios.	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando quiera acceder a la plataforma web, por lo tanto se realizará un login para controlar este acceso.	
Precondición	Ninguna.	
Secuencia normal	Paso	Acción
	1	El sistema solicita al usuario que se identifique, mediante su correo electrónico y su contraseña.
	2	El usuario proporciona los datos de identificación.
	3	El sistema confirma la identidad del usuario y le dirige a la página principal de la web, garantizándole acceso a la misma.
Postcondición	El sistema conoce al usuario y le permite el acceso a la plataforma.	
Excepciones	Paso	Acción
	3	Si la información introducida no es correcta el sistema muestra el correspondiente mensaje de error y solicita de nuevo al usuario los datos de identificación, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

Cuadro 5.25: UC-0003: Identificación de Usuario.

UC-0004	Editar Usuario
Versión	1.0 (18/06/2020)

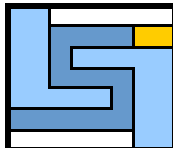


Autores	Jairo Fernández Carmona	
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez	
Dependencias	[IRQ-0006] Información sobre Usuarios. [OBJ-0010] Información sobre Usuarios. [OBJ-0011] Establecer Roles de Usuarios. [OBJ-0012] Control de Usuarios.	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando se desee modificar la información correspondiente a un usuario.	
Precondición	<p>El usuario que desea editar una cuenta de usuario es un usuario registrado con los permisos necesarios (rol de Administrador).</p> <p>Recordamos que los permisos relacionados a la edición de usuarios son:</p> <ul style="list-style-type: none">• Administrador: puede editar la información correspondiente a los usuarios.• Empleado: no puede editar cuentas de usuario. <p>Para comprobar que el usuario requiere de los permisos necesarios se habrá ejecutado con anterioridad al inicio de la acción el UC-0003.</p>	
Secuencia normal	Paso	Acción
	1	El actor Administrador (ACT-0001) solicita al sistema comenzar el proceso de modificación de datos, seleccionando el usuario a editar desde un listado.
	2	El sistema provee los datos del usuario especificado.
	3	El actor Administrador (ACT-0001) modifica los datos deseados.
	4	El sistema comprueba la validez de los datos introducidos.
	5	El sistema almacena los datos introducidos con la información actualizada.
	6	El sistema indica al usuario que la operación se ha realizado con éxito.

Postcondición	El sistema ha modificado los datos del usuario.	
Excepciones	Paso	Acción
	4	Si la información introducida no es correcta el sistema muestra el correspondiente mensaje de error y provee de nuevo al usuario de los datos requeridos, permitiendo de nuevo su edición y mostrando el error.
	6	Si se produce algún error durante el proceso, el sistema informa al usuario de lo ocurrido, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

Cuadro 5.26: UC-0004: Edición de Usuario.

UC-0005	Consulta Usuario
Versión	1.0 (18/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	[IRQ-0006] Información sobre Usuarios. [OBJ-0010] Información sobre Usuarios. [OBJ-0011] Establecer Roles de Usuarios. [OBJ-0012] Control de Usuarios.
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando se desee consultar los datos de cuenta de un usuario.



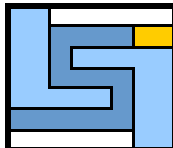
Precondición	<p>El usuario que desea consultar una cuenta de usuario es un usuario registrado con los permisos necesarios (rol de Administrador).</p> <p>Recordamos que los permisos relacionados a la consulta de usuarios son:</p> <ul style="list-style-type: none">• Administrador: puede consultar la información correspondiente a los usuarios.• Empleado: no puede consultar cuentas de usuario. <p>Para comprobar que el usuario requiere de los permisos necesarios se habrá ejecutado con anterioridad al inicio de la acción el UC-0003.</p>	
Secuencia normal	Paso	Acción
	1	El actor Administrador (ACT-0001) solicita al sistema la información de un usuario específico, eligiendo este usuario de un listado.
	2	El sistema provee los datos del usuario especificado.
Postcondición	El sistema muestra los datos del usuario seleccionado.	
Excepciones	Paso	Acción
	2	Si se produce algún error durante el proceso, el sistema informa al usuario de lo ocurrido, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

Cuadro 5.27: UC-0005: Consulta de Usuario.

Continuamos con el subsistema de Gestión de Partidos:

UC-0006	Registro de Partido
----------------	----------------------------

Versión	1.0 (18/06/2020)	
Autores	Jairo Fernández Carmona	
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez	
Dependencias	[IRQ-0005] Información sobre Partidos. [OBJ-0001] Proporcionar interfaz de recopilación de estadísticas de un partido. [OBJ-0007] Generar play by play. [OBJ-0008] Generar boxscore.	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando un usuario desee crear un partido para registrar acciones del mismo.	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) solicita crear un nuevo partido.
	2	El sistema solicita al usuario que solicite una serie de parámetros para la creación del partido, como pueden ser la liga, equipo local, equipo visitante, jugadores convocados, jugadores titulares, etc.
	3	El actor Usuario (ACT-0004) introduce los datos correspondientes.
	4	El sistema valida los datos introducidos.
	5	El sistema registra el partido.
	6	El sistema conduce al usuario a la interfaz del partido.
Postcondición	Se crea el partido en la base de datos.	
Excepciones	Paso	Acción
	4	Si los datos introducidos no son válidos, el sistema pedirá al usuario que repita la elección de parámetros, a continuación este caso de uso queda sin efecto.



	5	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

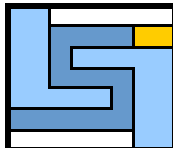
Cuadro 5.28: UC-0006: Registro de Partido.

UC-0007	Editar Partido	
Versión	1.0 (18/06/2020)	
Autores	Jairo Fernández Carmona	
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez	
Dependencias	[IRQ-0005] Información sobre Partidos. [OBJ-0001] Proporcionar interfaz de recopilación de estadísticas de un partido. [OBJ-0007] Generar play by play. [OBJ-0008] Generar boxscore. [OBJ-0009] Calcular estadísticas avanzadas.	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario desee añadir, editar o borrar las acciones correspondientes a un partido.	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) solicita editar un partido, el cual selecciona mediante una lista.
	2	El sistema dirige al usuario a la interfaz de partido.
	3	El actor Usuario (ACT-0004) edita los datos deseados.

	4	El sistema actualiza la base de datos.
	5	El sistema actualiza las estadísticas de temporada de los jugadores y equipos implicados en el partido editado al finalizar el mismo.
Postcondición	Los datos del partido son actualizados.	
Excepciones	Paso	Acción
	4	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
	5	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

Cuadro 5.29: UC-0007: Editar Partido.

UC-0008	Borrar Partido
Versión	1.0 (18/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	[IRQ-0005] Información sobre Partidos. [OBJ-0007] Generar play by play. [OBJ-0008] Generar boxscore. [OBJ-0009] Calcular estadísticas avanzadas.
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario desee borrar un partido y las acciones que almacena.



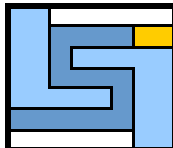
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003 y el partido que se desea borrar existe en la base de datos.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) solicita borrar un partido, eligiéndolo de una lista.
	2	El sistema confirma la eliminación del partido de la base de datos, además de actualizar las estadísticas de la temporada de los equipos y jugadores implicados en el partido.
Postcondición	El partido es borrado del sistema y las estadísticas actualizadas.	
Excepciones	Paso	Acción
	2	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

Cuadro 5.30: UC-0008: Borrar Partido.

UC-0009	Añadir acción
Versión	1.0 (18/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	[IRQ-0005] Información sobre Partidos. [OBJ-0001] Proporcionar interfaz de recopilación de estadísticas de un partido. [OBJ-0007] Generar play by play. [OBJ-0008] Generar boxscore. [OBJ-0009] Calcular estadísticas avanzadas.

Descripción	<p>El sistema deberá comportarse tal como se describe en el siguiente caso de uso abstracto durante la realización de los siguientes casos de uso:</p> <p>[UC-0006] Añadir Partido.</p> <p>[UC-0007] Editar Partido.</p>	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003 y se encuentra en la interfaz de partido.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) selecciona uno de los jugadores que está actualmente en pista.
	2	El actor Usuario (ACT-0003) selecciona la acción a realizar por el jugador, por ejemplo un tiro fallado, la captura de un rebote, etc.
	3	El sistema añade la acción al partido.
	4	El sistema actualiza las estadísticas del partido.
Postcondición	Se añade la acción al partido deseado.	
Excepciones	Paso	Acción
	3	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
	4	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

Cuadro 5.31: UC-0009: Añadir acción.

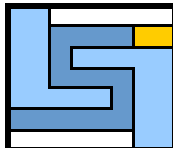


UC-0010	Borrar acción	
Versión	1.0 (18/06/2020)	
Autores	Jairo Fernández Carmona	
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez	
Dependencias	[IRQ-0005] Información sobre Partidos. [OBJ-0001] Proporcionar interfaz de recopilación de estadísticas de un partido. [OBJ-0007] Generar play by play. [OBJ-0008] Generar boxscore. [OBJ-0009] Calcular estadísticas avanzadas.	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso abstracto durante la realización de los siguientes casos de uso: [UC-0006] Añadir Partido. [UC-0007] Editar Partido. [UC-0008] Borrar Partido.	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) selecciona la acción del partido deseada para borrar.
	2	El sistema borra la acción del partido.
	3	El sistema actualiza las estadísticas del partido.
Postcondición	Se borra la acción al partido deseado.	
Excepciones	Paso	Acción
	2	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.

	3	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

Cuadro 5.32: UC-0010: Borrar acción.

UC-0011	Consultar Partido	
Versión	1.0 (18/06/2020)	
Autores	Jairo Fernández Carmona	
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez	
Dependencias	[IRQ-0005] Información sobre Partidos. [OBJ-0001] Proporcionar interfaz de recopilación de estadísticas de un partido. [OBJ-0007] Generar play by play. [OBJ-0008] Generar boxscore. [OBJ-0009] Calcular estadísticas avanzadas.	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario quiera consultar la información relacionada con un partido.	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003 y el partido deseado está registrado en el sistema.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) introduce los datos del partido que quiere consultar en una serie de filtros.
	2	El sistema muestra el/los partido/s que coinciden con los filtros introducidos.



Postcondición	Se muestra el partido deseado.	
Excepciones	Paso	Acción
	2	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

Cuadro 5.33: UC-0011: Consultar Partido.

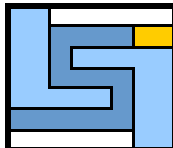
Mostramos los casos de uso identificados para el subsistema de Gestión de Clubs:

UC-0012	Añadir Club	
Versión	1.0 (19/06/2020)	
Autores	Jairo Fernández Carmona	
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez	
Dependencias	[IRQ-0002] Información sobre Clubs. [OBJ-0003] Proporcionar herramientas para el análisis estadístico. [OBJ-0005] Proporcionar información sobre un Jugador o Equipo.	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario desee registrar un nuevo club.	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) realiza la petición correspondiente al sistema para registrar un nuevo club.

	2	El sistema muestra al usuario un formulario que debe rellenar con la información correspondiente al club.
	3	El actor Usuario (ACT-0003) rellena el formulario con la información del club.
	4	El sistema valida la información introducida.
	5	El sistema registra en la base de datos el nuevo club.
	6	El sistema muestra un mensaje indicando que la operación se ha realizado con éxito.
Postcondición	El club queda registrado en el sistema.	
Excepciones	Paso	Acción
	4	Si la información introducida no es correcta el sistema muestra el correspondiente mensaje de error y solicita de nuevo al usuario los datos necesarios, a continuación este caso de uso queda sin efecto.
	5	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

Cuadro 5.34: UC-0012: Añadir Club.

UC-0013	Editar Club
Versión	1.0 (19/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	[IRQ-0002] Información sobre Clubs. [OBJ-0003] Proporcionar herramientas para el análisis estadístico.

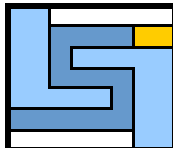


	[OBJ-0005] Proporcionar información sobre un Jugador o Equipo.	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario desee editar la información correspondiente a un club.	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003 y que el club deseado esté registrado en el sistema.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) realiza la petición de editar la información de un club, seleccionándolo de una lista proporcionada por el sistema, la cual puede actualizar mediante filtros para facilitar la búsqueda del club.
	2	El sistema muestra la información correspondiente al club.
	3	El actor Usuario (ACT-0003) actualiza la información correspondiente al club.
	4	El sistema valida la información introducida.
	5	El sistema registra la información del club actualizada.
	6	El sistema muestra un mensaje indicando que la operación se ha realizado con éxito.
Postcondición	La información del club se actualiza en la base de datos.	
Excepciones	Paso	Acción
	4	Si la información introducida no es correcta el sistema muestra el correspondiente mensaje de error y solicita de nuevo al usuario los datos necesarios, a continuación este caso de uso queda sin efecto.
	5	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	

Urgencia	Inmediatamente.
-----------------	-----------------

Cuadro 5.35: UC-0013: Editar Club.

UC-0014	Borrar Club	
Versión	1.0 (19/06/2020)	
Autores	Jairo Fernández Carmona	
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez	
Dependencias	[IRQ-0002] Información sobre Clubs. [OBJ-0003] Proporcionar herramientas para el análisis estadístico. [OBJ-0005] Proporcionar información sobre un Jugador o Equipo.	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario desee borrar un club del sistema.	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003 y el club que se desea borrar existe en el sistema.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) selecciona el club que desea borrar mediante una lista proporcionada por el sistema, la cual puede actualizar mediante filtros para facilitar la búsqueda del club.
	2	El sistema borra el club de la base de datos.
	3	El sistema muestra un mensaje confirmando el borrado del club y actualiza la vista de la lista de clubes.
Postcondición	El club es eliminado del sistema.	
Excepciones	Paso	Acción



	2	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

Cuadro 5.36: UC-0014: Borrar Club.

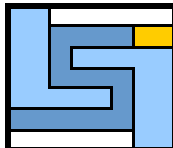
UC-0015	Consultar Club	
Versión	1.0 (19/06/2020)	
Autores	Jairo Fernández Carmona	
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez	
Dependencias	[IRQ-0002] Información sobre Clubs. [OBJ-0003] Proporcionar herramientas para el análisis estadístico. [OBJ-0005] Proporcionar información sobre un Jugador o Equipo.	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario quiera consultar la información relativa a un club.	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003 y el club que se desea consultar existe en el sistema.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) selecciona el club del cual desea consultar la información mediante una lista proporcionada por el sistema, la cual puede actualizar mediante filtros para facilitar la búsqueda del club.
	2	El sistema muestra la información guardada del club.

Postcondición	El sistema muestra en pantalla la información del club deseado.	
Excepciones	Paso	Acción
	2	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

Cuadro 5.37: UC-0015: Consultar Club.

Continuamos con los casos de uso propios del subsistema de Gestión de Equipos:

UC-0016	Añadir Equipo	
Versión	1.0 (19/06/2020)	
Autores	Jairo Fernández Carmona	
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez	
Dependencias	[IRQ-0003] Información sobre Equipos. [OBJ-0003] Proporcionar herramientas para el análisis estadístico. [OBJ-0005] Proporcionar información sobre un Jugador o Equipo.	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario desee añadir un nuevo equipo en el sistema.	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) realiza la petición correspondiente para registrar un nuevo equipo.

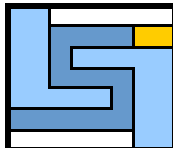


	2	El sistema muestra un formulario con los datos necesarios para registrar un nuevo equipo.
	3	El actor Usuario (ACT-0003) rellena los datos necesarios relativos al equipo.
	4	El sistema valida los datos introducidos.
	5	El sistema registra el equipo en la base de datos.
	6	El sistema muestra un mensaje indicando que la operación se ha realizado con éxito.
Postcondición	El equipo es registrado en el sistema.	
Excepciones	Paso	Acción
	4	Si la información introducida no es correcta el sistema muestra el correspondiente mensaje de error y solicita de nuevo al usuario los datos necesarios, a continuación este caso de uso queda sin efecto.
	5	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

Cuadro 5.38: UC-0016: Añadir Equipo.

UC-0017	Editar Equipo
Versión	1.0 (19/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	[IRQ-0003] Información sobre Equipos.

	[OBJ-0003] Proporcionar herramientas para el análisis estadístico. [OBJ-0005] Proporcionar información sobre un Jugador o Equipo.	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario desee editar un equipo.	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003 y el equipo está registrado en el sistema.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) realiza la petición correspondiente para editar un equipo, seleccionándolo de una lista, la cual puede actualizarse mediante filtros para facilitar su búsqueda.
	2	El sistema muestra la información del equipo deseado.
	3	El actor Usuario (ACT-0003) edita la información.
	4	El sistema valida los datos introducidos.
	5	El sistema registra la información actualizada en la base de datos.
	6	El sistema muestra un mensaje indicando que la operación se ha realizado con éxito.
Postcondición	La información relativa al equipo es actualizada en el sistema.	
Excepciones	Paso	Acción
	4	Si la información introducida no es correcta el sistema muestra el correspondiente mensaje de error y solicita de nuevo al usuario los datos necesarios, a continuación este caso de uso queda sin efecto.
	5	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	



Urgencia	Inmediatamente.
-----------------	-----------------

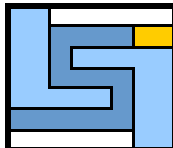
Cuadro 5.39: UC-0017: Editar Equipo.

UC-0018	Borrar Equipo	
Versión	1.0 (19/06/2020)	
Autores	Jairo Fernández Carmona	
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez	
Dependencias	[IRQ-0003] Información sobre Equipos. [OBJ-0003] Proporcionar herramientas para el análisis estadístico. [OBJ-0005] Proporcionar información sobre un Jugador o Equipo.	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario desee borrar un equipo del sistema.	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003 y el equipo está registrado en el sistema.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) realiza la petición correspondiente para borrar un equipo, seleccionándolo de una lista, la cual puede actualizarse mediante filtros para facilitar su búsqueda.
	2	El sistema borra el equipo deseado del sistema.
	3	El sistema muestra un mensaje de confirmación y actualiza la vista de la lista de equipos.
Postcondición	La información relativa al equipo es actualizada en el sistema.	
Excepciones	Paso	Acción

	2	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

Cuadro 5.40: UC-0018: Borrar Equipo.

UC-0019	Consultar Equipo	
Versión	1.0 (19/06/2020)	
Autores	Jairo Fernández Carmona	
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez	
Dependencias	[IRQ-0003] Información sobre Equipos. [OBJ-0003] Proporcionar herramientas para el análisis estadístico. [OBJ-0005] Proporcionar información sobre un Jugador o Equipo.	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario desee consultar el perfil y estadísticas de un equipo.	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003 y el equipo está registrado en el sistema.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) realiza la petición correspondiente para ver el perfil de un equipo, seleccionándolo de una lista, la cual puede actualizarse mediante filtros para facilitar su búsqueda.
	2	El sistema muestra el perfil del equipo, que contiene toda la información relativa a sus estadísticas e información del mismo.



Postcondición	La información relativa al equipo es actualizada en el sistema.	
Excepciones	Paso	Acción
	2	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

Cuadro 5.41: UC-0019: Consultar Equipo.

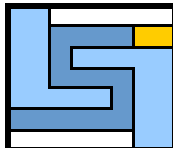
A continuación mostramos los casos de uso identificados para el subsistema de Visualización de Estadísticas:

UC-0020	Comparar Jugadores
Versión	1.0 (19/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	<p>[IRQ-0001] Información sobre Jugadores.</p> <p>[OBJ-0002] Mostrar análisis estadístico de jugadores y equipos.</p> <p>[OBJ-0003] Proporcionar herramientas para el análisis estadístico.</p> <p>[OBJ-0006] Proporcionar herramienta para comparación directa entre jugadores o equipos.</p> <p>[OBJ-0009] Calcular estadísticas avanzadas.</p>
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario desee comparar las estadísticas de dos jugadores “cara a cara”.

Precondición	El usuario ha accedido a la plataforma mediante el UC-0003, los jugadores deseados están registrados en el sistema y han disputado algún partido.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) accede al perfil de uno de los jugadores.
	2	El actor Usuario (ACT-0003) elige entre comparación de estadísticas utilizando una tabla o comparación de hexágonos estadísticos.
	3	El actor Usuario (ACT-0003) elige el segundo jugador a comparar, utilizando los filtros a su disposición.
	4	El sistema muestra las estadísticas de ambos jugadores.
Postcondición	Se puede observar una comparación estadística entre dos jugadores.	
Excepciones	Paso	Acción
	4	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

Cuadro 5.42: UC-0020: Comparar Jugadores.

UC-0021	Comparar Equipos
Versión	1.0 (19/06/2020)
Autores	Jairo Fernández Carmona

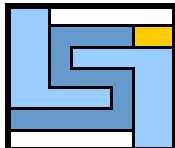


Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez	
Dependencias	<p>[IRQ-0003] Información sobre Equipos.</p> <p>[OBJ-0002] Mostrar análisis estadístico de jugadores y equipos.</p> <p>[OBJ-0003] Proporcionar herramientas para el análisis estadístico.</p> <p>[OBJ-0006] Proporcionar herramienta para comparación directa entre jugadores o equipos.</p> <p>[OBJ-0009] Calcular estadísticas avanzadas.</p>	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario desee comparar las estadísticas de dos equipos “cara a cara”.	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003, los equipos deseados están registrados en el sistema y han disputado algún partido.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) accede al perfil de uno de los equipos.
	2	El actor Usuario (ACT-0003) elige entre comparación de estadísticas utilizando una tabla o comparación de hexágonos estadísticos.
	3	El actor Usuario (ACT-0003) elige el segundo equipo a comparar, utilizando los filtros a su disposición.
	4	El sistema muestra las estadísticas de ambos equipos.
Postcondición	Se puede observar una comparación estadística entre dos equipos.	
Excepciones	Paso	Acción
	4	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.

Importancia	Vital.
Urgencia	Inmediatamente.

Cuadro 5.43: UC-0021: Comparar Equipos.

UC-0022	Comparar Jugadores en Eje de Coordenadas	
Versión	1.0 (19/06/2020)	
Autores	Jairo Fernández Carmona	
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez	
Dependencias	<p>[IRQ-0001] Información sobre Jugadores.</p> <p>[OBJ-0002] Mostrar análisis estadístico de jugadores y equipos.</p> <p>[OBJ-0003] Proporcionar herramientas para el análisis estadístico.</p> <p>[OBJ-0004] Proporcionar eje de coordenadas para comparación estadística.</p> <p>[OBJ-0006] Proporcionar herramienta para comparación directa entre jugadores o equipos.</p> <p>[OBJ-0009] Calcular estadísticas avanzadas.</p>	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario desee observar una comparativa de jugadores, escogiendo dos parámetros estadísticos que servirán para ordenar percentilmente (en los ejes X e Y) a los jugadores en un eje de coordenadas.	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003, hay jugadores registrados en el sistema que han disputado algún partido.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) accede al apartado de visualización de estadísticas.

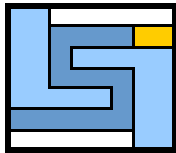


	2	El actor Usuario (ACT-0003) accede al apartado de estadísticas de jugadores.
	3	El actor Usuario (ACT-0003) elige la liga de la cual comparará a los jugadores.
	4	El actor Usuario (ACT-0003) elige una estadística de jugador para el eje X, la cual ordenará percentilmente a los jugadores en ese eje.
	5	El actor Usuario (ACT-0003) elige una estadística de jugador para el eje Y, la cual ordenará percentilmente a los jugadores en ese eje.
	6	El sistema mostrará todos los jugadores de la liga ordenados percentilmente mediante los parámetros elegidos.
Postcondición	Se puede observar un eje de coordenadas (con ejes X e Y) en el que se ordenarán percentilmente todos los jugadores de la liga.	
Excepciones	Paso	Acción
	6	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

Cuadro 5.44: UC-0022: Comparar Jugadores en Eje de Coordenadas.

UC-0023	Comparar Equipos en Eje de Coordenadas
Versión	1.0 (19/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	[IRQ-0003] Información sobre Equipos. [OBJ-0002] Mostrar análisis estadístico de jugadores y equipos.

	<p>[OBJ-0003] Proporcionar herramientas para el análisis estadístico.</p> <p>[OBJ-0004] Proporcionar eje de coordenadas para comparación estadística.</p> <p>[OBJ-0006] Proporcionar herramienta para comparación directa entre jugadores o equipos.</p> <p>[OBJ-0009] Calcular estadísticas avanzadas.</p>	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario desee observar una comparativa de equipos, escogiendo dos parámetros estadísticos que servirán para ordenar percentilmente (en los ejes X e Y) a los equipos en un eje de coordenadas.	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003, hay equipos registrados en el sistema que han disputado algún partido.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) accede al apartado de visualización de estadísticas.
	2	El actor Usuario (ACT-0003) accede al apartado de estadísticas de equipos.
	3	El actor Usuario (ACT-0003) elige la liga de la cual comparará a los equipos.
	4	El actor Usuario (ACT-0003) elige una estadística de equipo para el eje X, la cual ordenará percentilmente a los equipos en ese eje.
	5	El actor Usuario (ACT-0003) elige una estadística de equipo para el eje Y, la cual ordenará percentilmente a los equipos en ese eje.
	6	El sistema mostrará todos los equipos de la liga ordenados percentilmente mediante los parámetros elegidos.
Postcondición	Se puede observar un eje de coordenadas (con ejes X e Y) en el que se ordenarán percentilmente todos los equipos de la liga.	
Excepciones	Paso	Acción



	6	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

Cuadro 5.45: UC-0023: Comparar Equipos en Eje de Coordenadas.

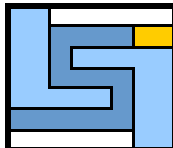
Mostramos ahora los casos de uso del subsistema de Gestión de Jugadores:

UC-0024	Añadir Jugador	
Versión	1.0 (19/06/2020)	
Autores	Jairo Fernández Carmona	
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez	
Dependencias	[IRQ-0001] Información sobre Jugadores. [OBJ-0002] Mostrar análisis estadístico de jugadores y equipos. [OBJ-0005] Proporcionar información sobre un Jugador o Equipo.	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario desee registrar un nuevo jugador en el sistema.	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003, el equipo al que se quiere asignar el jugador ya está registrado en la plataforma.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) realiza la petición para registrar el jugador en el sistema.
	2	El sistema muestra un formulario con los datos a introducir del jugador.

	3	El actor Usuario (ACT-0003) rellena los datos correspondientes.
	4	El sistema valida la información proporcionada.
	5	El sistema registra al jugador en la base de datos.
	6	El sistema muestra un mensaje de confirmación, indicando que el registro se ha realizado con éxito.
Postcondición	Se registra al jugador en el sistema.	
Excepciones	Paso	Acción
	4	Si la información introducida no es correcta el sistema muestra el correspondiente mensaje de error y solicita de nuevo al usuario los datos necesarios, a continuación este caso de uso queda sin efecto.
	5	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

Cuadro 5.46: UC-0024: Añadir Jugador.

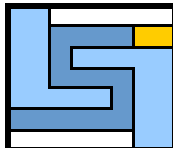
UC-0025	Editar Jugador
Versión	1.0 (19/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	[IRQ-0001] Información sobre Jugadores. [OBJ-0002] Mostrar análisis estadístico de jugadores y equipos. [OBJ-0005] Proporcionar información sobre un Jugador o Equipo.



Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario desee editar la información de un jugador que ya esté registrado en el sistema.	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003, el jugador deseado existe en la base de datos.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) elige al jugador que quiere editar mediante una lista proveída por el sistema, la cual puede filtrar con una serie de parámetros.
	2	El sistema muestra un formulario con los actuales del jugador.
	3	El actor Usuario (ACT-0003) modifica los datos deseados.
	4	El sistema valida la información proporcionada.
	5	El sistema actualiza la información en la base de datos.
	6	El sistema muestra un mensaje de confirmación, indicando que el registro se ha realizado con éxito.
Postcondición	Se actualiza la información del jugador en el sistema.	
Excepciones	Paso	Acción
	4	Si la información introducida no es correcta el sistema muestra el correspondiente mensaje de error y solicita de nuevo al usuario los datos necesarios, a continuación este caso de uso queda sin efecto.
Excepciones	5	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

Cuadro 5.47: UC-0025: Editar Jugador.

UC-0026	Borrar Jugador	
Versión	1.0 (19/06/2020)	
Autores	Jairo Fernández Carmona	
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez	
Dependencias	[IRQ-0001] Información sobre Jugadores. [OBJ-0002] Mostrar análisis estadístico de jugadores y equipos. [OBJ-0005] Proporcionar información sobre un Jugador o Equipo.	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario desee borrar uno de los jugadores registrados en la base de datos.	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003, el jugador deseado existe en la base de datos.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) elige al jugador que quiere eliminar mediante una lista proveída por el sistema, la cual puede filtrar con una serie de parámetros.
	2	El sistema elimina al jugador de la base de datos.
	3	El sistema muestra un mensaje de confirmación del borrado.
Postcondición	Se borra el jugador de la base de datos.	
Excepciones	Paso	Acción
	2	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	



Urgencia	Inmediatamente.
-----------------	-----------------

Cuadro 5.48: UC-0026: Borrar Jugador.

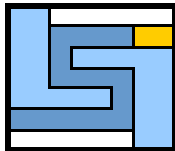
UC-0027	Consultar Jugador	
Versión	1.0 (19/06/2020)	
Autores	Jairo Fernández Carmona	
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez	
Dependencias	[IRQ-0001] Información sobre Jugadores. [OBJ-0002] Mostrar análisis estadístico de jugadores y equipos. [OBJ-0005] Proporcionar información sobre un Jugador o Equipo.	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario desee consultar la información relacionada a un jugador.	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003, el jugador deseado existe en la base de datos.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) elige al jugador del cual quiere observar la información o estadísticas relacionadas.
	2	El sistema muestra la información del jugador.
Postcondición	Se muestra la información del jugador especificado.	
Excepciones	Paso	Acción
	2	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.

Importancia	Vital.
Urgencia	Inmediatamente.

Cuadro 5.49: UC-0027: Consultar Jugador.

Para finalizar con los casos de uso, mostramos los relacionados al subsistema de Gestión de Ligas:

UC-0028	Añadir Liga	
Versión	1.0 (19/06/2020)	
Autores	Jairo Fernández Carmona	
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez	
Dependencias	[IRQ-0007] Información sobre Ligas. [OBJ-0002] Mostrar análisis estadístico de jugadores y equipos. [OBJ-0003] Proporcionar herramientas para el análisis estadístico. [OBJ-0005] Proporcionar información sobre un Jugador o Equipo.	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario desee registrar una nueva liga en el sistema.	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) realiza la petición específica para comenzar el registro de una nueva liga en el sistema.
	2	El sistema muestra un formulario con los datos que se desean guardar de la liga.

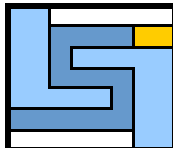


	3	El actor Usuario (ACT-0003) rellena los campos del formulario con la información relativa a la liga.
	4	El sistema valida la información proporcionada.
	5	El sistema registra la liga en la base de datos.
	6	El sistema muestra un mensaje de confirmación.
Postcondición	Se registra la liga en el sistema.	
Excepciones	Paso	Acción
	4	Si la información introducida no es correcta el sistema muestra el correspondiente mensaje de error y solicita de nuevo al usuario los datos necesarios, a continuación este caso de uso queda sin efecto.
	5	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

Cuadro 5.50: UC-0028: Añadir Liga.

UC-0029	Editar Liga
Versión	1.0 (19/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	[IRQ-0007] Información sobre Ligas. [OBJ-0002] Mostrar análisis estadístico de jugadores y equipos. [OBJ-0003] Proporcionar herramientas para el análisis estadístico.

	[OBJ-0005] Proporcionar información sobre un Jugador o Equipo.	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario desee editar la información relacionada a una liga.	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003 y la liga deseada existe en el sistema.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) elige la liga a editar mediante una lista proporcionada por el sistema, la cual puede actualizar mediante una serie de filtros para facilitar su búsqueda.
	2	El sistema muestra la información de la liga almacenada en la base de datos.
	3	El actor Usuario (ACT-0003) actualiza los campos deseados.
	4	El sistema valida la información proporcionada.
	5	El sistema actualiza la información en la base de datos.
	6	El sistema muestra un mensaje de confirmación.
Postcondición	Se actualiza la información de la liga en el sistema.	
Excepciones	Paso	Acción
	4	Si la información introducida no es correcta el sistema muestra el correspondiente mensaje de error y solicita de nuevo al usuario los datos necesarios, a continuación este caso de uso queda sin efecto.
	5	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	



Urgencia	Inmediatamente.
-----------------	-----------------

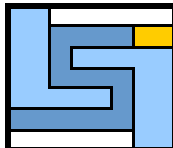
Cuadro 5.51: UC-0029: Editar Liga.

UC-0030	Borrar Liga	
Versión	1.0 (19/06/2020)	
Autores	Jairo Fernández Carmona	
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez	
Dependencias	[IRQ-0007] Información sobre Ligas. [OBJ-0002] Mostrar análisis estadístico de jugadores y equipos. [OBJ-0003] Proporcionar herramientas para el análisis estadístico. [OBJ-0005] Proporcionar información sobre un Jugador o Equipo.	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario desee borrar una liga de la base de datos.	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003 y la liga deseada existe en el sistema.	
Secuencia normal	Paso	Acción
	1	El actor Usuario (ACT-0003) elige la liga a eliminar mediante una lista proporcionada por el sistema, la cual puede actualizar mediante una serie de filtros para facilitar su búsqueda.
	2	El sistema elimina la liga de la base de datos.
	3	El sistema muestra un mensaje de confirmación.
	4	El sistema actualiza la vista de la lista de las ligas.

Postcondición	Se actualiza la información de la liga en el sistema.	
Excepciones	Paso	Acción
	2	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

Cuadro 5.52: UC-0030: Borrar Liga.

UC-0031	Consultar Liga	
Versión	1.0 (19/06/2020)	
Autores	Jairo Fernández Carmona	
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez	
Dependencias	[IRQ-0007] Información sobre Ligas. [OBJ-0002] Mostrar análisis estadístico de jugadores y equipos. [OBJ-0003] Proporcionar herramientas para el análisis estadístico. [OBJ-0005] Proporcionar información sobre un Jugador o Equipo.	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario desee consultar la información registrada de una liga.	
Precondición	El usuario ha accedido a la plataforma mediante el UC-0003 y la liga deseada existe en el sistema.	
Secuencia normal	Paso	Acción



	1	El actor Usuario (ACT-0003) elige la liga que desea consultar mediante una lista proporcionada por el sistema, la cual puede actualizar mediante una serie de filtros para facilitar su búsqueda.
	2	El sistema muestra la información de la liga.
Postcondición	Se muestra la información relacionada a la liga.	
Excepciones	Paso	Acción
	2	Si se detecta algún error durante el proceso, el sistema mostrará un mensaje de error al usuario, a continuación este caso de uso queda sin efecto.
Importancia	Vital.	
Urgencia	Inmediatamente.	

Cuadro 5.53: UC-0031: Consultar Liga.

5.5.3 Requisitos No Funcionales

En esta sección mostraremos la lista de requisitos no funcionales del sistema, especificados mediante las siguientes tablas:

NFR-0001	Usabilidad
Versión	1.0 (19/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	Ninguna.

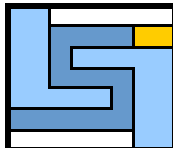
Descripción	El sistema deberá presentar información útil para cumplir con los objetivos descritos anteriormente, utilizando herramientas de análisis de datos, representados algunos de ellos por gráficos de fácil entendimiento. De esa forma animaremos a todas las personas a utilizar el sistema y no habrá grupos que sientan incomodidad en el uso del portal web.
Importancia	Vital.
Urgencia	Inmediatamente.

Cuadro 5.54: NFR-0001: Accesibilidad.

NFR-0002	Calidad
Versión	1.0 (19/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	Ninguna.
Descripción	El sistema deberá presentar un buen nivel de calidad, tanto técnica como visual, consiguiendo así la atención de los usuarios del sistema. Para asegurar esta calidad utilizaremos guías y documentación de estilo para el diseño de interfaces.
Importancia	Vital.
Urgencia	Inmediatamente.

Cuadro 5.55: NFR-0002: Calidad.

NFR-0003	Seguridad
-----------------	------------------



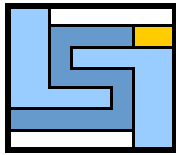
Versión	1.0 (19/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	Ninguna.
Descripción	El sistema deberá proporcionar mecanismos que proporcionen al sistema un nivel adecuado de seguridad, protegiendo el acceso a la plataforma y la edición de datos conforma a lo establecido en puntos anteriores. Al contener datos personales este aspecto es de suma importancia para evitar el filtrado de los mismos.
Importancia	Vital.
Urgencia	Inmediatamente.

Cuadro 5.56: NFR-0003: Seguridad.

NFR-0004	Uso de Software de Actualidad
Versión	1.0 (19/06/2020)
Autores	Jairo Fernández Carmona
Fuentes	Jairo Fernández Carmona José Antonio Troyano Jiménez
Dependencias	Ninguna.
Descripción	<p>El sistema deberá ser desarrollado con tecnologías y softwares actuales, ya que nos facilitará la consulta de documentación, entre otros aspectos, además de tener una finalidad didáctica para el autor del proyecto, como ya mencionamos.</p> <p>Para cumplir con este requisito utilizaremos el stack MEAN, compuesto por:</p> <ul style="list-style-type: none">• NodeJS.• AngularJS.• MongoDB.• Express.

Importancia	Vital.
Urgencia	Inmediatamente.

Cuadro 5.57: NFR-0004: Uso de Software de Actualidad.



Capítulo 6

6. Diseño e implementación

En este capítulo hablaremos, entrando más en profundidad, del diseño del proyecto y cómo hemos implementado distintas soluciones a los problemas encontrados en el desarrollo.

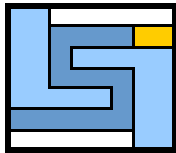
6.1 Stack tecnológico

Vamos a mencionar a continuación las tecnologías utilizadas en este trabajo de fin de grado.

- **IDE:** Visual Studio Code.
- **Control de versiones:** GitHub.
- **Stack MEAN:**
 - MongoDB.
 - Express.
 - AngularJS.
 - NodeJS.
- **Esquemas tipados en MongoDB:** Mongoose.
- **Lenguajes:**
 - JavaScript.
 - HTML.
 - CSS.
 - TypeScript (siendo este un superconjunto de JavaScript).
- **Hosting web y API:** Heroku.
- **Hosting de base de datos:** MongoDB Atlas.
- **Hosting de imágenes:** Cloudinary.
- **Aplicación de seguimiento de tiempo:** Toggl.

Para finalizar comentaremos la finalidad de algunos paquetes de Node instalados que nos han facilitado ciertas operaciones:

- [Jsonwebtoken](#), para la creación de tokens.



- [Bcrypt](#), para la encriptación de las contraseñas en la base de datos.
- [Api-query-params](#), convierte los parámetros enviados en las peticiones en queries de MongoDB.
- [Cloudinary](#), integración del servicio de almacenamiento de imágenes para Node.
- [Dotenv](#), creación de variables de entorno para el backend.
- [Multer](#), middleware para manejar la subida de imágenes desde el portal web.

6.2 Desarrollo

En este apartado vamos a comentar una serie de implementaciones que hemos realizado en el proyecto que creemos que merecen la pena ser comentadas, ya que han sido decisiones que hemos tomado entre distintas opciones.

6.2.1 MongoDB Atlas

Para comenzar hablaremos sobre el uso de MongoDB Atlas. Mencionamos anteriormente que queríamos abaratar el precio del trabajo lo máximo posible haciendo que el proyecto fuese escalable en el futuro.

Para ello hemos utilizado la versión cloud de MongoDB, bajo el nombre de MongoDB Atlas, que en su tier gratuito nos permite almacenar hasta 512MB. Utilizar las versiones cloud de los softwares con los que hemos desarrollado el trabajo nos ha parecido la mejor idea, ya que hace que el tamaño de almacenamiento de la aplicación completa esté repartida, mejorando su escalabilidad para el futuro.

En este caso tenemos la base de datos en MongoDB Atlas, el hosting web y de la API en Heroku y el host de imágenes en Cloudinary (del cual hablaremos más adelante).

El desarrollo en MongoDB a través del paquete Mongoose ha sido muy satisfactorio y la conexión a la versión cloud muy sencilla.

Aunque la idea inicial del proyecto no era utilizar esquemas en nuestra base de datos, nos pareció una implementación importante una vez que comenzamos el trabajo, aunque durante el desarrollo del mismo nos hizo invertir mucho tiempo en actualizar esta característica, ya que añadimos o editamos campos de los modelos en varias ocasiones, se decidió que era algo necesario en el proyecto, ya que los datos tenían

cierta complejidad (como por ejemplo muchos campos anidados), por lo que el tipado nos pareció la opción correcta.

Como conclusión, creemos que la decisión de utilizar MongoDB Atlas y Mongoose ha sido la correcta, haciendo que la escalabilidad esté asegurada y que al tipar los datos los futuros desarrolladores del proyecto no se encuentren con una curva de aprendizaje elevada en cuanto al diseño de la base de datos.

6.2.2 Seguridad mediante tokens

Otro punto importante del trabajo era la seguridad, ya que almacenaremos datos personales que no deberían ser accedidos por cualquier persona.

Para controlar el acceso a estos datos hemos implementado tokens, los cuales creamos y comprobamos mediante el paquete de Node [jsonwebtoken](#).

Cada vez que recibimos un login en nuestra API comprobamos que el usuario exista en nuestra base de datos, en caso de ser así creamos un token que firmamos utilizando el paquete mencionado en el párrafo anterior con una seed privada almacenada en nuestro sistema.

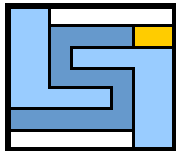
Al generar este token lo almacenamos en el *Local Storage* del usuario y comprobaremos su existencia y validez cuando se quiera acceder a la plataforma web o realizar alguna petición a nuestra API.

6.2.3 Hosting de imágenes en Cloudinary

La decisión en cuanto al almacenamiento de imágenes fue un tema recurrente cuando realizamos el diseño inicial del proyecto.

La primera idea fue guardar las imágenes en el mismo documento que la contenía, por ejemplo para un jugador, la idea era guardar su foto en un campo del esquema Jugador en MongoDB Atlas.

Al pensar en la escalabilidad descartamos esta idea, optamos por hacer que el tamaño de memoria utilizada en el host de MongoDB Atlas no se viese afectado por la cantidad que pudiese derivar del almacenamiento de imágenes, por lo que buscamos



alternativas gratuitas cuya calidad técnica estuviera al nivel del proyecto, hasta que encontramos Cloudinary.

Cloudinary nos permite, en su tier gratuito, almacenar hasta 25GB de imágenes, a través de su API o de su aplicación web.

En nuestro caso, cada vez que el usuario desea subir una imagen a nuestra aplicación realizamos una conexión a la API de Cloudinary, almacenamos en nuestro sistema la imagen seleccionada, subimos a nuestra cuenta de Cloudinary la imagen, recibimos la URL de la misma, guardamos esta URL como propiedad del jugador (o del objeto con el que estemos trabajando) y borramos la imagen de nuestro sistema.

Creemos que la opción elegida al desarrollar con Cloudinary cumple con los objetivos que nos planteamos, por lo que el resultado ha sido totalmente satisfactorio.

6.2.4 Creación de gráficos con Echarts

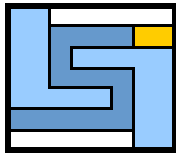
Otra de las decisiones que tuvimos que tomar fue la elección de la herramienta de creación de gráficos para visualizar los datos estadísticos.

Aunque barajamos muchas opciones nos decantamos por Echarts, ya que comprende una amplia variedad de gráficos que son visualmente muy atractivos. Creemos que es una librería muy amplia, que permite mucha personalización y que además automatiza muchos de los valores que no introduzcamos a la hora de crear la gráfica, por lo que es muy útil para usuarios que no deseen editar mucho sus gráficas (aunque no es nuestro caso). Aunque pueda parecer una nimiedad otras de las librerías que probamos a utilizar necesitaban de una personalización excesiva desde nuestro punto de vista, como tener que especificar todos los colores de las series de datos introducidas.

Capítulo 7

7. Manual de usuario

A continuación mostraremos un manual para nuevos usuarios de la plataforma, que nos ayuda además a mostrar el portal desde un punto de vista estético.



7.1 Login

Vamos a comenzar mostrando el login de la aplicación, que hemos nombrado **My Team App**.

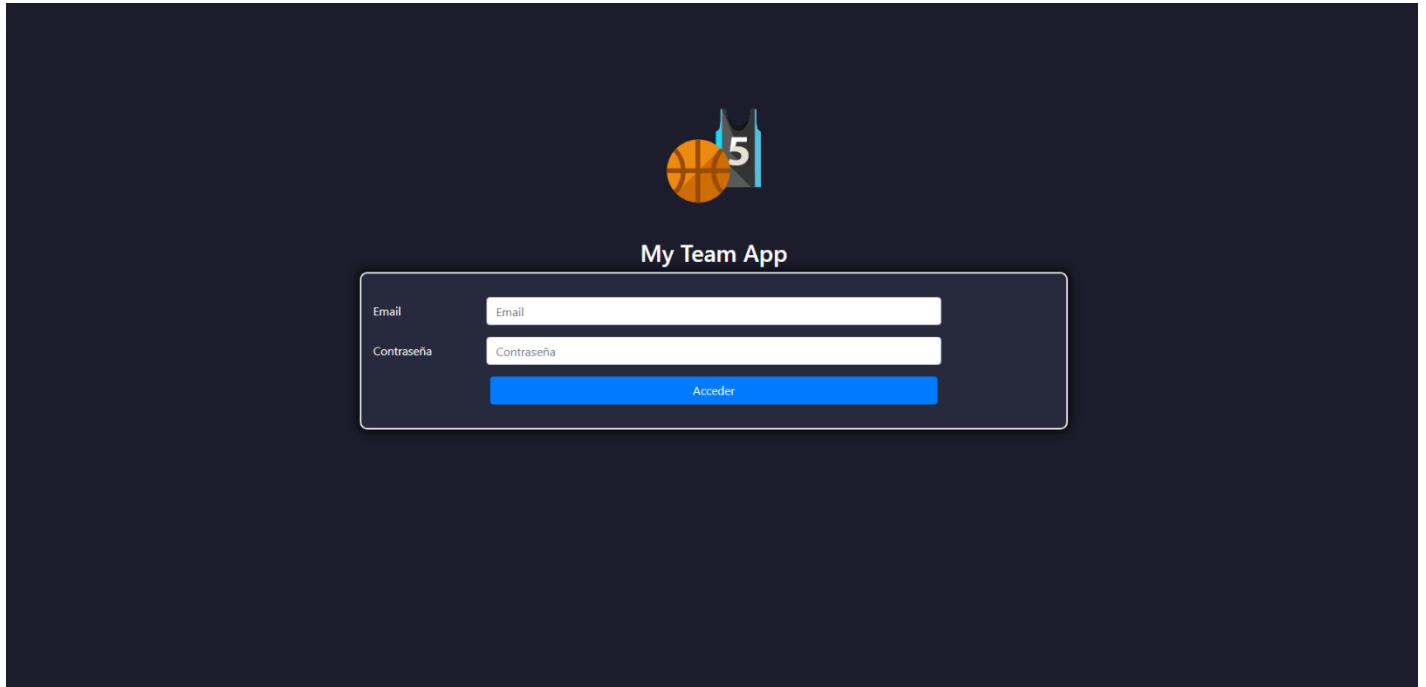


Figura 7.1: Login.

Podemos observar que hemos optado por utilizar una gama de colores básicos y oscuros.

7.2 Partidos

En la siguiente imagen podemos ver la página a la que se dirige el usuario una vez realiza el acceso a la web. En esta pantalla se encuentra la lista de partidos registrados en el sistema, que podemos filtrar por liga o club que disputa el partido. Mediante los botones laterales de la tarjeta del partido podemos dirigirnos a la pantalla de edición del partido o eliminarlo.

En la parte izquierda está el menú de la web, desde donde podemos movernos entre secciones de la misma, que son:

- Partidos.
- Jugadores.
- Equipos.

- Clubs.
- Ligas.
- Estadísticas.
- Usuarios.

Debajo de estas secciones tenemos el botón para cerrar sesión. Por último, aclarar que la sección de *Usuarios* solo es accesible por usuarios con rol de Administrador, además de desaparecer del menú si el usuario que accede es un Empleado.

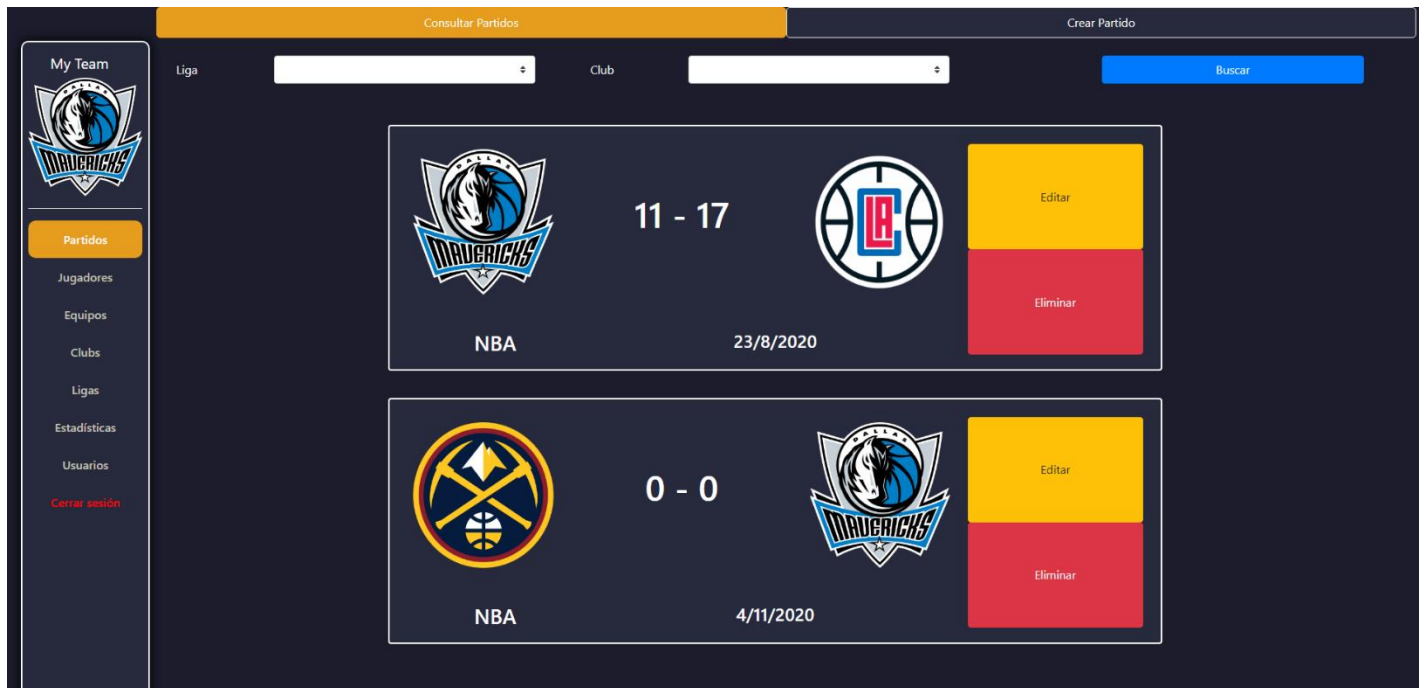


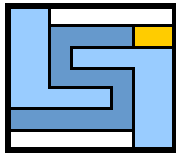
Figura 7.2: Pantalla de Partidos.

En la pantalla de partido tenemos tres partes, la pantalla principal, llamada *En Vivo*, el *Boxscore* y el *Play by Play*. Podemos movernos entre estas secciones utilizando el menú superior.

En los siguientes puntos mostraremos los aspectos más interesantes del portal web.

7.2.1 En Vivo

En la sección de *En Vivo* podemos registrar todas las acciones que ocurren en el partido: anotar canastas, rebotes, controlar el tiempo del partido, realizar sustituciones de jugadores, etc.



Comenzaremos mostrando una imagen general de esta pantalla y luego añadiremos los distintos menús por los que se puede mover el usuario para añadir cualquier acción del partido.

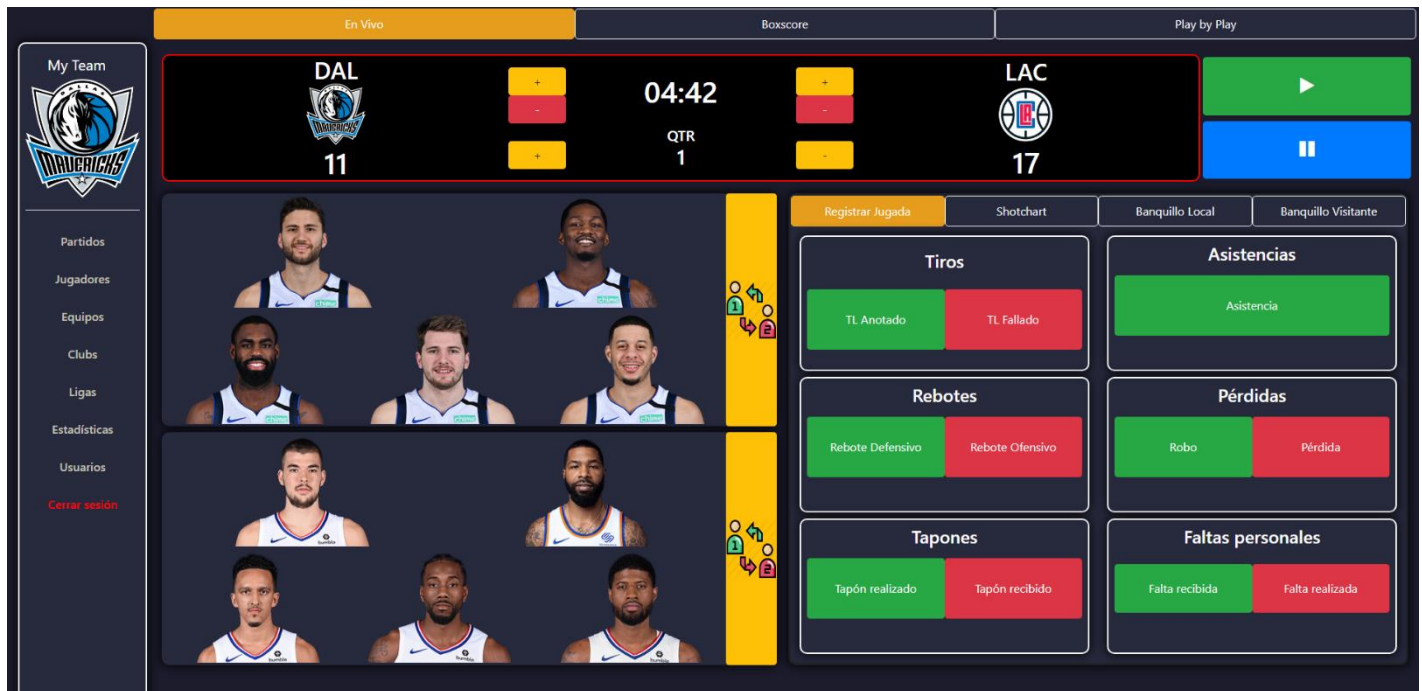


Figura 7.3: Pantalla de Partido: En Vivo.

A continuación veremos el panel que por defecto se ve al entrar a la página, el de registro de jugadas, su funcionamiento es muy simple, seleccionamos un jugador pulsando en él y acto seguido pulsamos en el botón que se corresponde con la acción elegida.



Figura 7.4: Pantalla de Partido: Registrar Jugada.

Una de las características que teníamos como objetivo desde el principio del proyecto era crear un shotchart, pudiendo anotar no solo los tiros de los jugadores, sino también en qué posición de la pista se producen. Dependiendo de la zona seleccionada el shotchart calcula automáticamente si el tiro ha sido de dos o tres puntos. La división de las zonas de la pista es una adaptación de la realizada por Adrià Arbues, el 18 de abril de 2020, en su artículo *Clustering Euroleague shooters* [2]. Para anotar un tiro simplemente se selecciona un jugador, se pulsa en la zona deseada de la pista y se elige entre “Acierto” o “Fallo”.

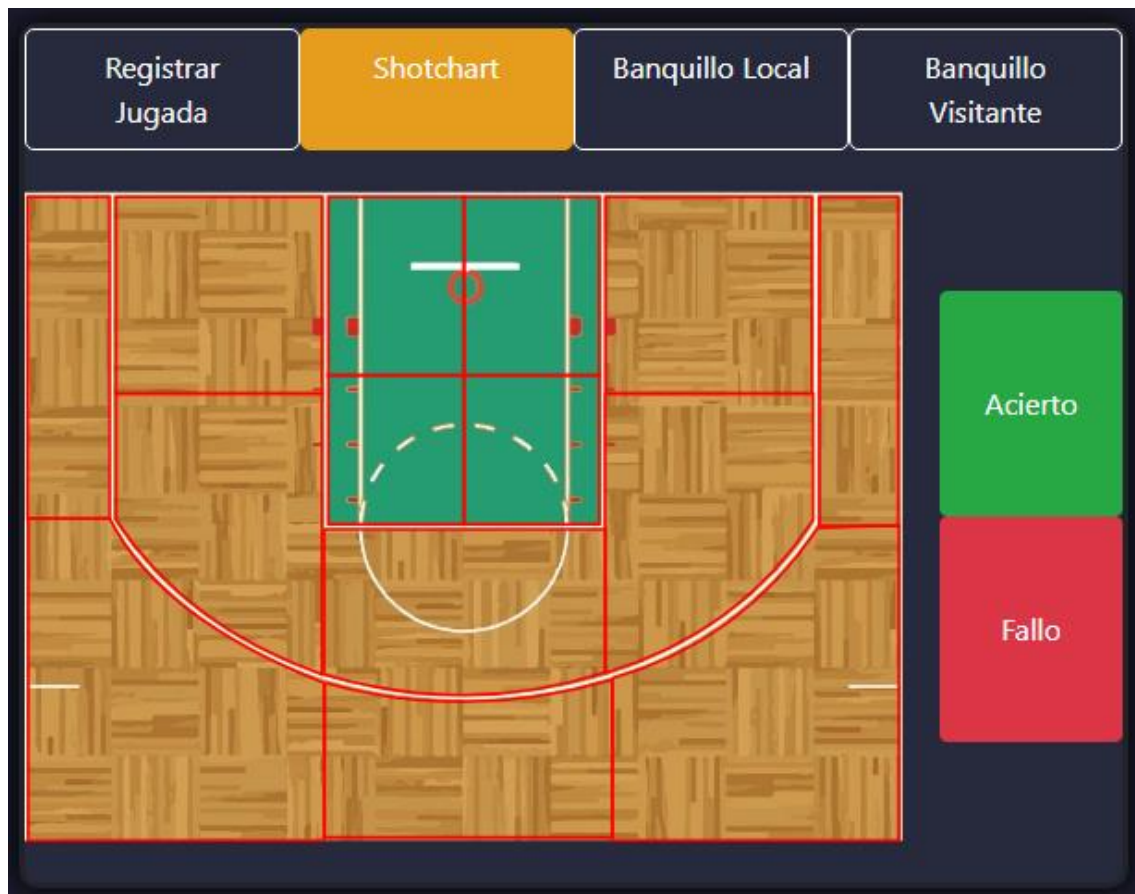
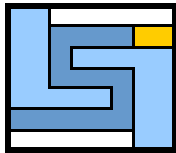


Figura 7.5: Pantalla de Partido: Shotchart.

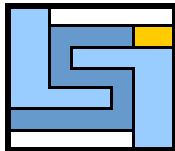
El funcionamiento de las sustituciones de jugadores es parecida, simplemente seleccionamos uno de los jugadores en pista, hacemos lo mismo con uno de los jugadores del banquillo y pulsamos en el botón amarillo que se encuentra entre ellos. En la siguiente imagen hemos seleccionado a Jose Juan Barea como jugador que entrará a pista.



Figura 7.6: Pantalla de Partido: Banquillo.

7.2.2 Boxscore

En esta pantalla podemos ver las estadísticas del partido en tiempo real, cada vez que accedemos a esta página las estadísticas son actualizadas, por lo que se puede obtener un punto de vista estadístico con rapidez en cualquier momento.



7.2. Partidos

En Vivo

Boxscore

Play by Play

Dallas Mavericks

D	Nombre	Min	P	TL A	TL I	%TL	T2 A	T2 I	%T2	T3 A	T3 I	%T3	RD	RO	RT	A	BR	BP	TF	TC	FP R	FP C	%USO
1	Courtney Lee	00:00	0	0	0	0%	0	0	0%	0	0	0%	0	0	0	0	0	0	0	0	0	0	0%
5	Jose Juan Barea	00:00	0	0	0	0%	0	0	0%	0	0	0%	0	0	0	0	0	0	0	0	0	0	0%
9	Michael Kidd-Gilchrist	00:00	0	0	0	0%	0	0	0%	0	0	0%	0	0	0	0	0	0	0	0	0	0	0%
10	Dorian Finney-Smith	07:18	0	0	0	0%	0	1	0%	0	0	0%	1	0	1	0	1	0	1	0	0	0	6.25%
11	Tim Hardaway	07:18	2	0	0	0%	1	2	50%	0	3	0%	1	0	1	0	0	0	0	0	0	1	31.25%
23	Trey Burke	06:19	5	0	0	0%	1	1	100%	1	1	100%	0	0	0	0	0	0	0	0	0	0	14.29%
30	Seth Curry	00:59	2	0	0	0%	1	1	100%	0	0	0%	0	0	0	0	0	0	0	0	0	0	44.49%
42	Maxi Kleber	07:18	0	0	0	0%	0	1	0%	0	1	0%	1	1	2	0	0	0	0	0	0	1	12.5%
44	Justin Jackson	00:00	0	0	0	0%	0	0	0%	0	0	0%	0	0	0	0	0	0	0	0	0	0	0%
51	Boban Marjanovic	00:00	0	0	0	0%	0	0	0%	0	0	0%	0	0	0	0	0	0	0	0	0	0	0%
77	Luka Doncic	07:18	2	0	0	0%	1	2	50%	0	1	0%	3	0	3	3	0	2	0	0	1	0	31.25%
	TOTAL		11	0	0	0%	4	8	50%	1	6	16.67%	6	1	7	3	1	2	1	0	1	2	0

Los Angeles Clippers

D	Nombre	Min	P	TL A	TL I	%TL	T2 A	T2 I	%T2	T3 A	T3 I	%T3	RD	RO	RT	A	BR	BP	TF	TC	FP R	FP C	%USO
1	Reggie Jackson	00:00	0	0	0	0%	0	0	0%	0	0	0%	0	0	0	0	0	0	0	0	0	0	0%

Figura 7.7: Pantalla de Partido: Boxscore.

7.2.3 Play by Play

Por último, en esta sección podemos ver un resumen de las jugadas del partido, ordenadas de más nuevas a más antiguas.

Desde aquí podemos borrar cualquiera de las jugadas, actualizándose todas las jugadas posteriores, además de las estadísticas del partido.











En Vivo			Boxscore			Play by Play		
1C	04:46		Luka Doncic					
11	17	Asistencia						
1C	04:47		Seth Curry					
11	17	+2 Tiro anotado						
1C	04:54		Luka Doncic					
9	17	Rebote defensivo						
1C	04:58		Kawhi Leonard					
9	17	X Tiro fallado						

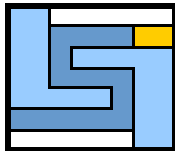
Figura 7.8: Pantalla de Partido: Play by Play.

7.2.4 Crear Partido

A continuación vamos a mostrar el formulario desde el cual el usuario puede introducir los parámetros necesarios para crear un partido.

El primer paso es elegir la liga a la que pertenece el partido, la fecha y la temporada. A continuación se seleccionan los dos equipos y el sistema carga todos los jugadores, pudiendo elegir los jugadores convocados y los titulares.

Para este último paso el sistema comprueba que se cumplen una serie de restricciones, como escoger un mínimo de 5 jugadores por equipo, escoger un quinteto inicial, no poder seleccionar un jugador para el quinteto inicial que no esté convocado, etc.



7.3. Jugadores

Consultar Partidos

Crear Partido

Crear partido

Liga

NBA

Dejar en blanco o seleccionar "Amistoso" en caso de no ser partido oficial.

Fecha del partido

13/11/2020

Temporada

2020

Equipo Local

Equipo

Dallas Mavericks

Dorsal	Jugador	Convocado	Si
1	Courtney Lee	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	Jose Juan Barea	<input checked="" type="checkbox"/>	<input type="checkbox"/>
6	Kristaps Porzingis	<input type="checkbox"/>	<input type="checkbox"/>
7	Dwight Powell	<input type="checkbox"/>	<input type="checkbox"/>
9	Michael Kidd-Gilchrist	<input checked="" type="checkbox"/>	<input type="checkbox"/>
10	Dorian Finney-Smith	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
11	Tim Hardaway	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Equipo Visitante

Equipo

Los Angeles Clippers

Dorsal	Jugador	Convocado	Si
1	Reggie Jackson	<input checked="" type="checkbox"/>	<input type="checkbox"/>
2	Kawhi Leonard	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
4	JaMychal Green	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	Montrezl Harrell	<input checked="" type="checkbox"/>	<input type="checkbox"/>
13	Paul George	<input checked="" type="checkbox"/>	<input type="checkbox"/>
14	Terance Mann	<input type="checkbox"/>	<input type="checkbox"/>
20	Landry Shamet	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Figura 7.9: Crear Partido.

7.3 Jugadores

7.3.1 Crear Jugador

Consultar Jugadores

Crear Jugador

My Team

Partidos

Jugadores

Equipos

Clubs

Ligas

Estadísticas

Usuarios

Cerrar sesión

Crear jugador

Nombre

Nombre

Apellido/s

Apellidos

Imagen

Seleccionar archivo

Ningún archivo seleccionado

Subir imagen

Fecha de nacimiento

dd/mm/aaaa

Correo

Correo electrónico

Posición principal

Posición secundaria

Peso

Peso (kg)

Altura

Altura (cm)

Club

Dorsal

Dorsal

Guardar

Figura 7.10: Crear Jugador.

7.3.2 Consultar Jugadores

En esta pantalla tenemos una lista de todos los jugadores registrados en el sistema, pudiendo filtrarlos y elegir entre observar su perfil, editar su información o eliminarlo.

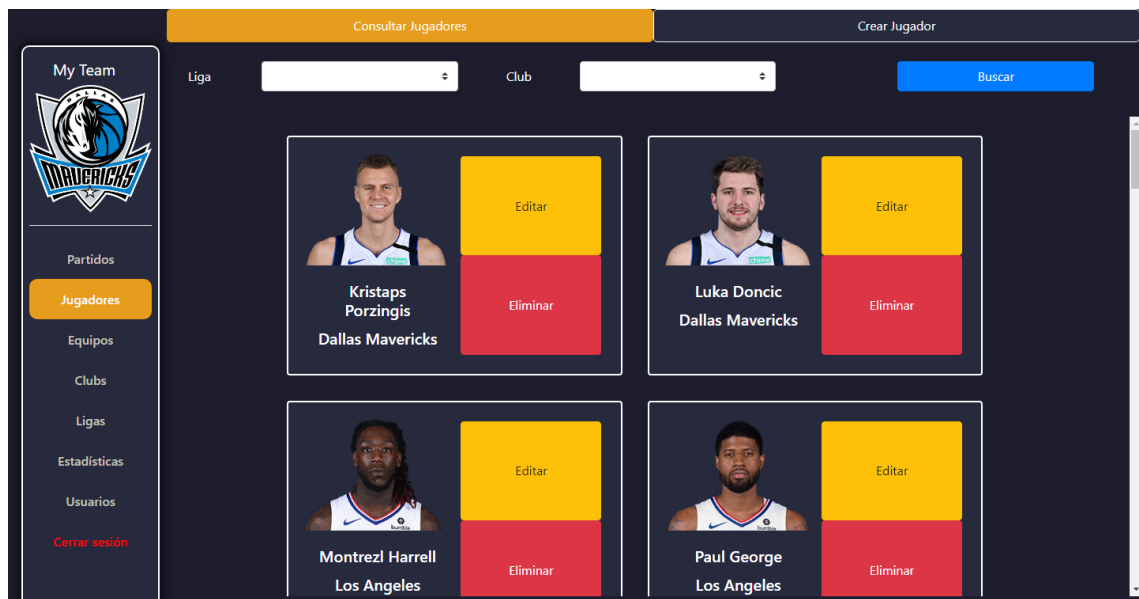


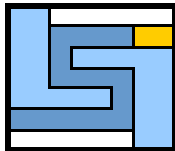
Figura 7.11: Consultar Jugadores.

7.3.3 Perfil Jugador

A continuación observamos el perfil de un jugador.

En su parte superior tenemos información personal y un heptágono estadístico. El heptágono actúa como resumen estadístico de la temporada del jugador, este pentágono realiza una comparación entre todos los jugadores de la liga a la que pertenecen y los ordena percentilmente en cada una de las estadísticas.

En la parte inferior tenemos tres secciones, en la primera podemos ver las estadísticas acumuladas y por partido del jugador en la temporada, sus estadísticas de tiro y un par de gráficos que nos muestran un desglose de los tiros realizados por el jugador.



7.3. Jugadores

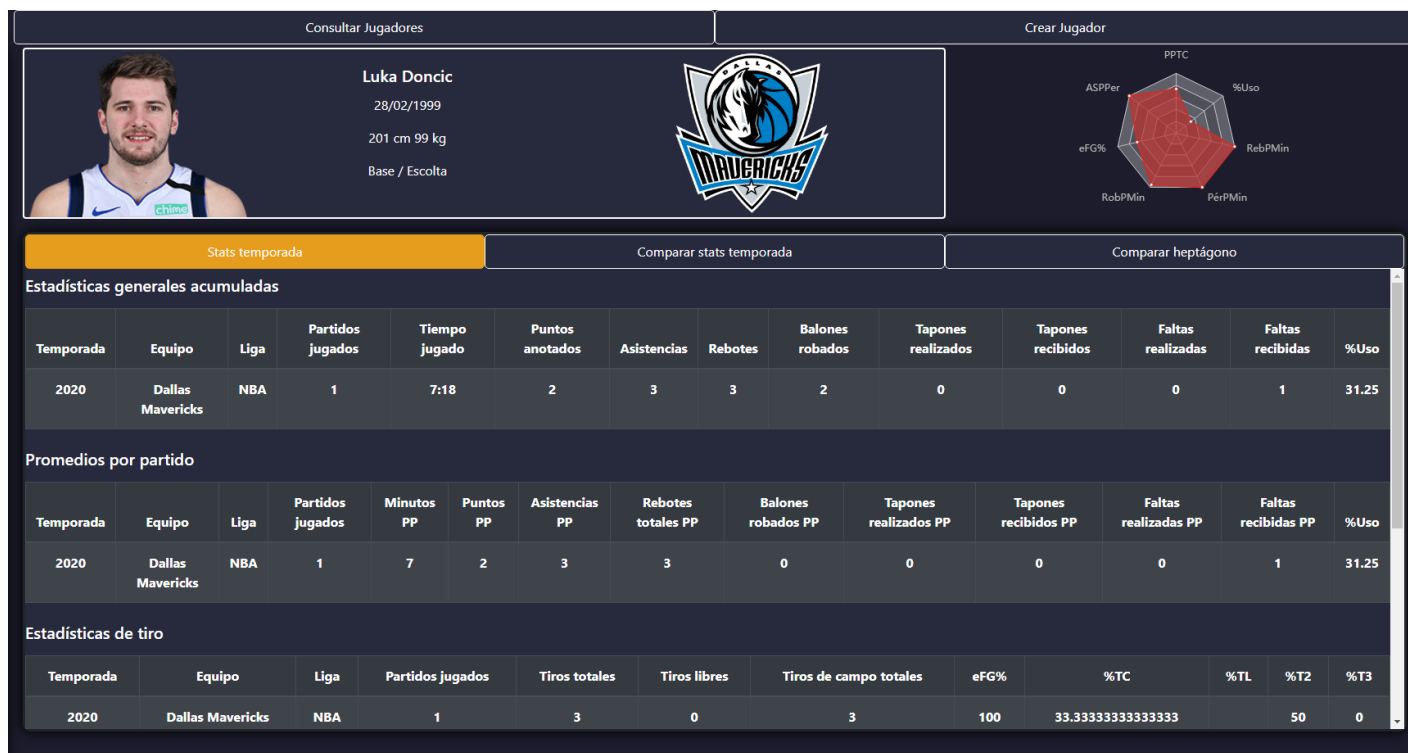


Figura 7.12: Perfil de Jugador.



Figura 7.13: Perfil de Jugador: Estadísticas de tiro.

En las siguientes secciones podemos elegir un jugador con el que comparar las estadísticas de jugador de la temporada contra el jugador el cual estamos observando.


Stats temporada	Comparar stats temporada		Comparar heptágono
	1	Puntos / T2	0
	0	Puntos / T3	0
	3	Asistencias PP	0
	1.5	Asistencias PPer	
	3	Rebotes PP	2
	0.43	Rebotes PMin	0.29
	0	Rebotes Ofensivos PMin	0.14
	0.43	Rebotes Defensivos PMin	0.14
	0	Tapones Realizados PP	0
	0	Tapones Realizados PMin	0
	0	Tapones Recibidos PP	0

Figura 7.14: Perfil de Jugador: Comparador de estadísticas.

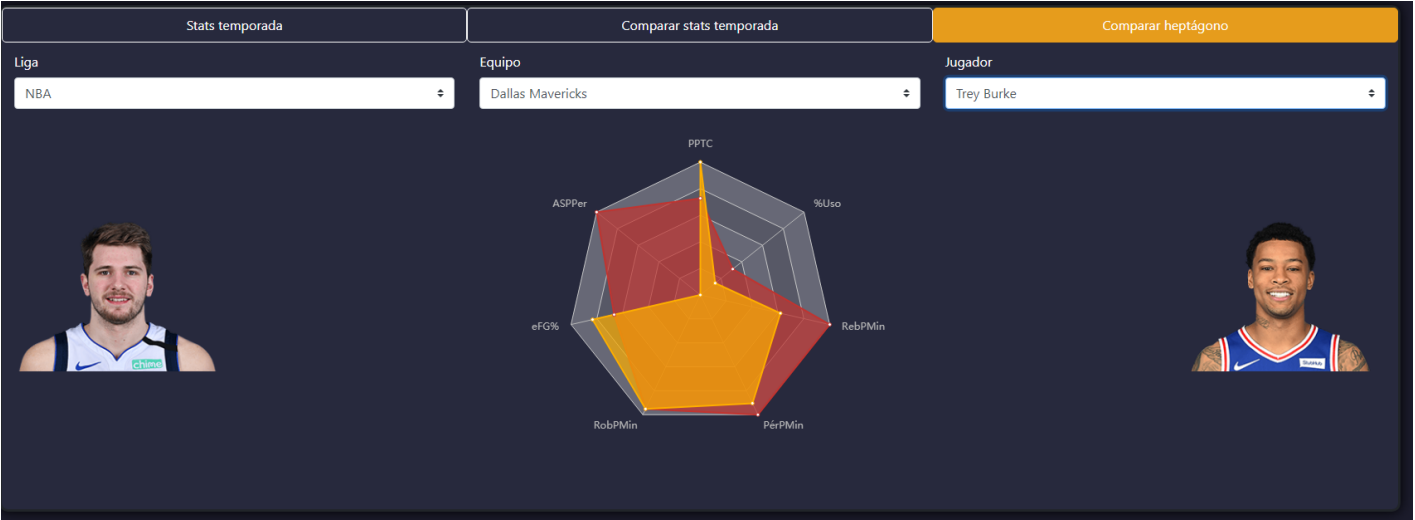
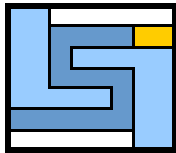


Figura 7.15: Perfil de Jugador: Comparador de heptágonos.

7.4 Equipos

7.4.1 Perfil de Equipo



7.5. Estadísticas

En la siguiente imagen mostramos el perfil del equipo, el cual es muy parecido al de los jugadores, la principal diferencia con este es que en el apartado de *Stats temporada* podemos observar más gráficas que detallan el funcionamiento deportivo del equipo.

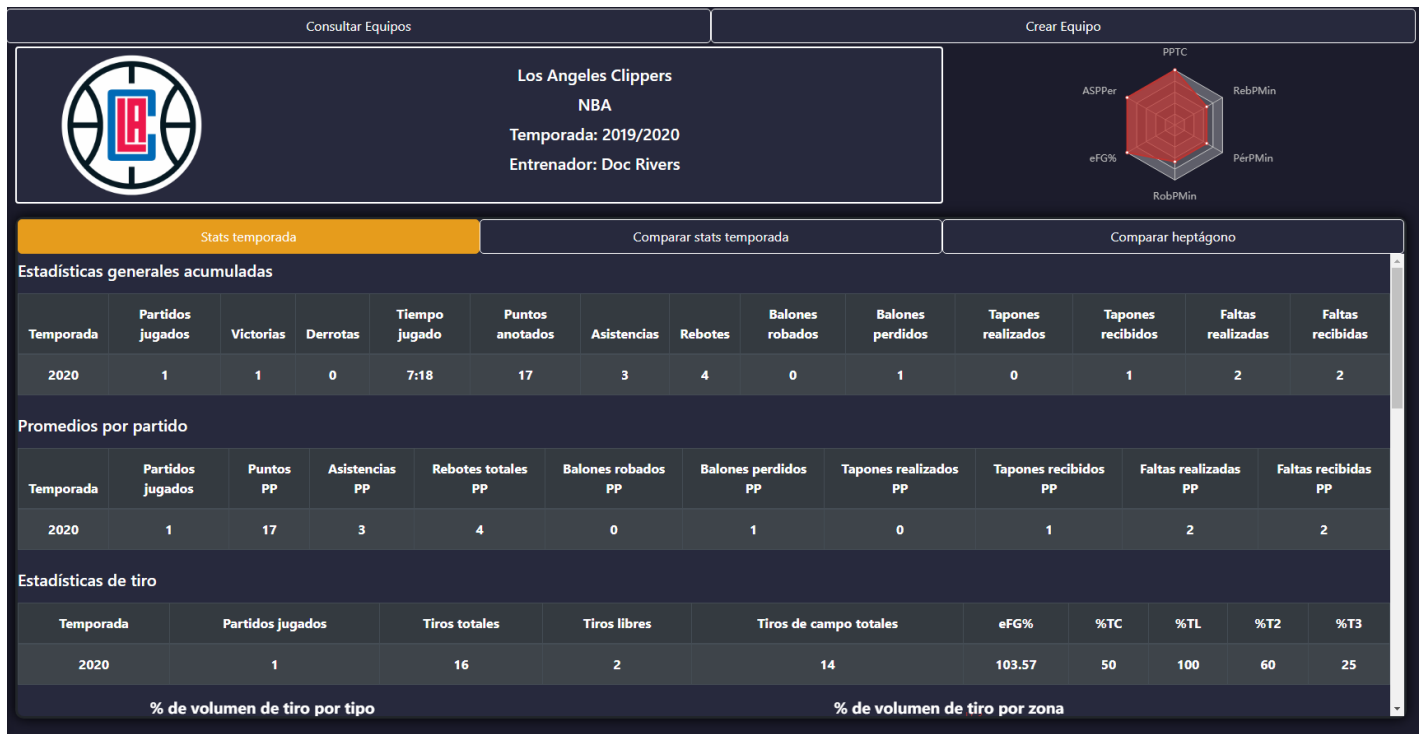


Figura 7.16: Perfil de Equipo.

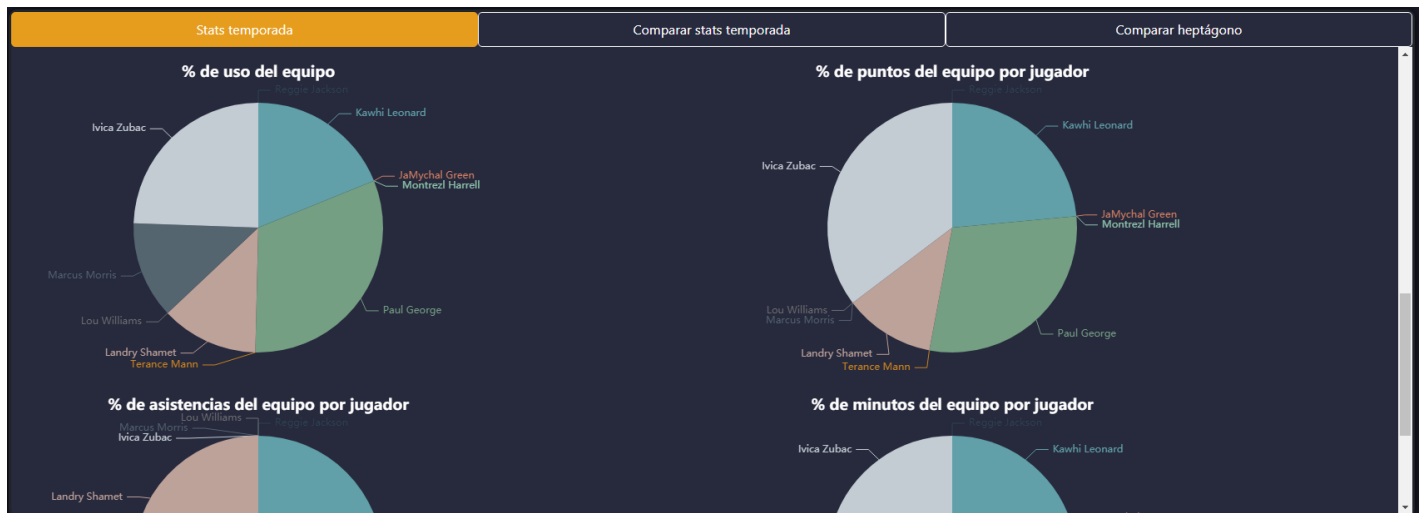


Figura 7.17: Perfil de Equipo: Estadísticas de temporada.

7.5 Estadísticas

Para finalizar este apartado mostraremos la tabla comparativa. En ella podemos comparar jugadores o equipos, eligiendo los ejes por los cuales ordenaremos percentilmente a todos los jugadores de la liga seleccionada.

Disponemos en esta tabla de **41 estadísticas distintas** por las que ordenar a los jugadores y 42 en el caso de los equipos.

Como muestra adjuntamos una imagen en la que comparamos todos los jugadores de la liga NBA, ordenando en el eje Y por el %Uso y en el eje X por los Puntos totales anotados en la temporada y en el que resaltamos uno de los puntos para observar el nombre del jugador y el percentil exacto de ambas estadísticas.

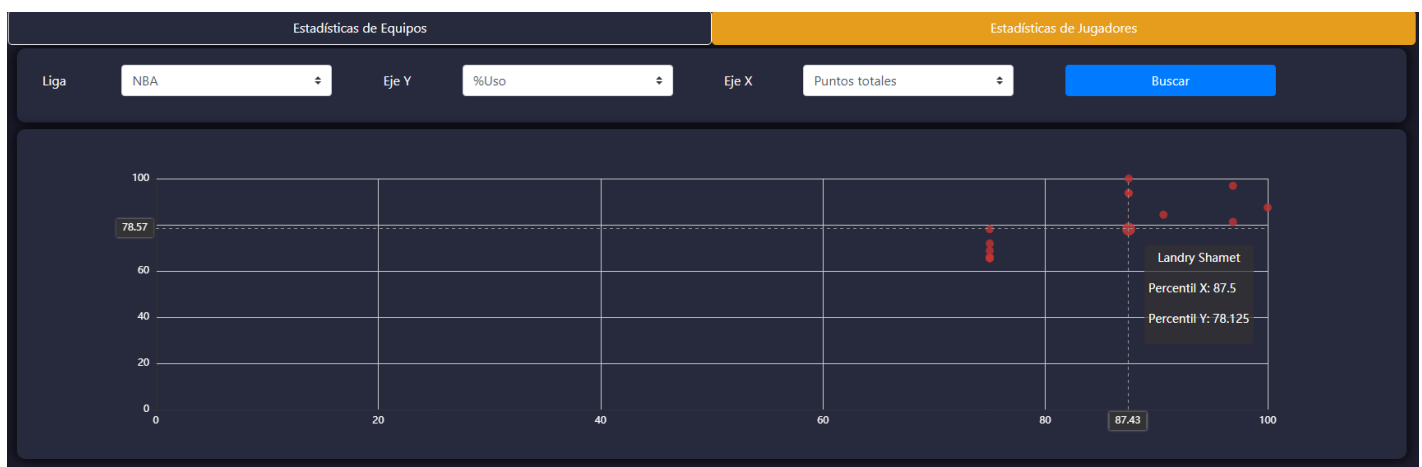
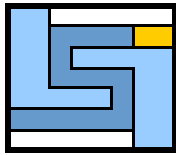


Figura 7.18: Tabla comparativa de Jugadores.



Capítulo 8

8. Conclusiones

En el último capítulo de este trabajo de fin de grado, analizaremos los objetivos del proyecto, comprobando si se han cumplido y comentando las dificultades principales encontradas.

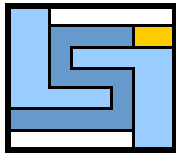
8.1 Evaluación del proyecto

El querer desarrollar una plataforma de registro de estadísticas de baloncesto ha hecho que tuviésemos que trabajar en muchos aspectos del desarrollo web, desde creación de base de datos hasta el despliegue de la aplicación.

Al realizar el estudio del arte, comparando aplicaciones de uso actual, nos dimos cuenta de que la diversidad en contenido no comprendía más allá del registro de estadísticas, sin poder realizar consultas posteriores más allá del boxscore. En algunos casos, además, la interfaz gráfica era pobre y confusa. Por lo que nos marcamos el objetivo de realizar un portal web agradable visualmente y con contenido de utilidad, yendo más allá del simple registro de las acciones de los partidos, para ello, debíamos desarrollar un trabajo de calidad en la base de datos y servidor, dividiendo los datos almacenados en diferentes hosts para asegurar la escalabilidad, además de abaratar el precio.

Con el trabajo final de grado finalizado, cuyo contenido hemos expuesto a lo largo de este documento, podemos asegurar que los objetivos técnicos han sido satisfechos, habiendo conseguido profundidad estadística, una interfaz atractiva y sencilla y un sistema robusto.

Desde el punto de vista de los objetivos académicos, que comprendían una complejidad añadida al no tener conocimientos previos del stack MEAN al completo, la valoración es muy positiva, ya que se han adquirido conocimientos sobre el stack con el desarrollo de una aplicación completa de cierta complejidad.



Bibliografía

[1] BOE, resolución de 22 de febrero de 2018, XVII Convenio colectivo estatal de empresas de consultoría y estudios de mercado y de la opinión pública. 2008. (Página 11). [Link](#).

[2] Adrià Arbués, Clustering Euroleague shooters, Medium. 18 abril 2020. (Página 105). [Link](#).