



**Universidad de Jaén**

*Escuela Politécnica Superior de Jaén*

# APLICACIÓN MÓVIL PARA GESTIÓN DE ENTIDADES DEPORTIVAS BASADA EN SERVICIOS WEB EN LA NUBE

Autor/a: Javier Torres Gálvez

Grado:

Tutor/a: Dr. Francisco J. Martínez Mimblera  
Dr. Pedro J. Sánchez Sánchez

Fecha: 05/06/2025

Licencia CC



CREA





Universidad de Jaén  
Escuela Politécnica Superior de Jaén  
Departamento de Informática

Don Francisco J. Martínez Mimblera y Don Pedro J. Sánchez Sánchez, tutores del Proyecto Fin de Carrera titulado: APLICACIÓN MÓVIL PARA GESTIÓN DE ENTIDADES DEPORTIVAS BASADA EN SERVICIOS WEB EN LA NUBE, que presenta Javier Torres Gálvez, autoriza su presentación para defensa y evaluación en la Escuela Politécnica Superior de Jaén.

Jaén, Junio de 2025

El alumno:

Javier Torres Gálvez

Los tutores:

MARTINEZ  
MIMBRERA  
FRANCISCO  
JESUS -  
77334644L

Firmado  
digitalmente por  
MARTINEZ  
MIMBRERA  
FRANCISCO JESUS  
- 77334644L  
Fecha: 2025.06.05  
11:44:12 +02'00'

SANCHEZ  
SANCHEZ  
PEDRO JOSE  
- 26007525E

Firmado  
digitalmente por  
SANCHEZ  
SANCHEZ PEDRO  
JOSE - 26007525E  
Fecha: 2025.06.05  
11:47:15 +02'00'

Francisco J. Martínez Mimblera

Pedro J. Sánchez Sánchez





| FICHA DEL TRABAJO FIN DE TÍTULO |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|---------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Titulación</b>               | Grado en Ingeniería Informática                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| <b>Modalidad</b>                | Proyecto de Ingeniería                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <b>Especialidad (solo TFG)</b>  | Ingeniería del software                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| <b>Mención (solo TFG)</b>       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| <b>Idioma</b>                   | Español                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| <b>Tipo</b>                     | Específico                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| <b>TFT en equipo</b>            | No                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| <b>Autor/a</b>                  | Javier Torres Gálvez                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>Fecha de asignación</b>      |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| <b>Descripción corta</b>        | <p>Actualmente el desarrollo de aplicaciones móviles está teniendo una gran demanda gracias al gran número de dispositivos que existen en el mercado y a las nuevas necesidades que han surgido por su uso cotidiano. Este tipo de tecnología ofrece un valor añadido a las organizaciones y empresas que lo utilizan. Dado que es un sector en crecimiento el alumno tendrá que desarrollar una aplicación móvil que permita a los usuarios/empleados de la entidad deportiva, poder acceder a los servicios que tiene y ofrece a sus socios y aficionados.</p> <p>Es por ello que en este TFG se ha desarrollado una aplicación móvil la cual permite la gestión de entidades deportivas y ofrece a los usuarios todo un abanico de posibilidades de noticias, información, abonos, entradas etc para facilitar su relación con la entidad. Además, la aplicación ha sido desarrollada multiplataforma (Android e IOS) y los microservicios de gestión se encuentran en la nube para facilitar la gestión y seguridad de los datos.</p> |

| NORMAS APLICADAS EN ESTE DOCUMENTO |                                                                                                                                                                  |
|------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| LOCALES                            |                                                                                                                                                                  |
| TFT-UJA:2017                       | Normativa de Trabajos Fin de Grado, Fin de Máster y otros Trabajos Fin de Título de la Universidad de Jaén (Normativa marco UJA aprobada en Consejo de Gobierno) |
| TFT-EPSJ:2017                      | Normativa sobre Trabajos Fin de Grado y Fin de Máster en la Escuela Politécnica Superior de Jaén (Normativa EPSJ aprobada en Junta de Escuela)                   |
| TFT-EPSJ                           | Criterios de evaluación y normas de estilo para TFG y TFM de la Escuela Politécnica Superior de Jaén                                                             |
| NACIONALES E INTERNACIONALES       |                                                                                                                                                                  |
| ISO 2145:1978                      | Documentación - Numeración de divisiones y subdivisiones en documentos escritos                                                                                  |
| UNE 50132:1994                     | Traducción de la ISO 2145                                                                                                                                        |
| IEEE                               | Estilo de referencias y citas de IEEE (Institute of Electrical and Electronics Engineers)                                                                        |

| NORMAS UTILIZADAS COMO BASE O REFERENCIA                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |                                                                                                      |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------|
| NACIONALES                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |                                                                                                      |
| UNE 157001:2014                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | Criterios generales para la elaboración formal de los documentos que constituyen un proyecto técnico |
| UNE 157801:2007                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | Criterios generales para la elaboración de proyectos de sistemas de información                      |
| <p><i>Estas normas se han utilizado <b>como base o referencia</b> para la inclusión de algunos contenidos y definiciones sobre elaboración de proyectos, entendiendo como <b>proyecto</b> la documentación consensuada entre una empresa y un cliente, que da lugar al perfeccionamiento de un contrato para la elaboración de una obra o la prestación de un servicio. Por consiguiente, no debe esperarse la aplicación de estas normas en cuanto a la completitud de los contenidos ni a la organización de los mismos.</i></p> |                                                                                                      |

## Índice

### Tabla de contenido

|                                                                                 |            |
|---------------------------------------------------------------------------------|------------|
| <b>1. Introducción.....</b>                                                     | <b>10</b>  |
| 1.1 Contexto.....                                                               | 10         |
| 1.2 Motivación.....                                                             | 10         |
| 1.2 Objetivos .....                                                             | 11         |
| 1.3 Estructura del trabajo .....                                                | 11         |
| 1.4 Planificación temporal .....                                                | 12         |
| 1.4.1 Tareas.....                                                               | 12         |
| 1.4.2 Diagrama de Gantt.....                                                    | 14         |
| <b>2. Análisis del sistema .....</b>                                            | <b>15</b>  |
| 2.1 Análisis del sistema de gestión deportiva actual implantado en el club..... | 15         |
| 2.2 Especificación de los requerimientos del sistema .....                      | 17         |
| 2.2.1 Requerimientos funcionales.....                                           | 17         |
| 2.2.2 Requerimientos no funcionales.....                                        | 19         |
| 2.3 Casos de uso .....                                                          | 20         |
| 2.4 Análisis posible arquitectura de la solución .....                          | 30         |
| <b>3. Diseño del sistema .....</b>                                              | <b>33</b>  |
| 3.1 Modelo de Datos.....                                                        | 34         |
| 3.2 Tablas del sistema .....                                                    | 37         |
| 3.3 Diseño UX/UI aplicación móvil.....                                          | 38         |
| 3.3.1 StoryBoards .....                                                         | 39         |
| 3.3.2 Wireframes.....                                                           | 55         |
| 3.3.3 Guía de estilo .....                                                      | 72         |
| <b>4. Implementación del sistema.....</b>                                       | <b>77</b>  |
| 4.1 Herramientas utilizadas para la implementación.....                         | 77         |
| 4.2 Arquitectura implementada.....                                              | 81         |
| 4.3 Servicios implementados .....                                               | 82         |
| <b>5. Pruebas del software .....</b>                                            | <b>87</b>  |
| 5.1 Pruebas de análisis de código estático: Seguridad y calidad .....           | 87         |
| 5.2 Pruebas manuales del software.....                                          | 92         |
| <b>6. Conclusiones y trabajos futuros.....</b>                                  | <b>101</b> |
| <b>Anexos.....</b>                                                              | <b>103</b> |

|                                                                                        |                   |
|----------------------------------------------------------------------------------------|-------------------|
| <b>A. Manual instalación y configuración servicios web en servidor.....</b>            | <b>103</b>        |
| <b>B. Manual instalación y funcionamiento Aplicación móvil .....</b>                   | <b>106</b>        |
| <b>C. Documentación del proceso de subida de la Aplicación móvil a la tienda .....</b> | <b>120</b>        |
| <b><i>Bibliografía.....</i></b>                                                        | <b><i>129</i></b> |

## Tabla de imágenes

|                                        |    |
|----------------------------------------|----|
| Diagrama de Gantt del proyecto 1 ..... | 14 |
| Figura web Oleoinnova Mengíbar 2.....  | 15 |
| Diagrama de “CU-01” 2.3.....           | 20 |
| Diagrama de “CU-02” 2.4.....           | 21 |
| Diagrama de “CU-03” 5.....             | 21 |
| Diagrama de “CU-04” 6.....             | 22 |
| Diagrama de “CU-05” 7.....             | 23 |
| Diagrama de “CU-06” 8.....             | 24 |
| Diagrama de “CU-07” 9.....             | 25 |
| Diagrama de “CU-08” 10.....            | 25 |
| Diagrama de “CU-09” 11.....            | 26 |
| Diagrama de “CU-10” 12.....            | 27 |
| Diagrama de “CU-11” 13.....            | 27 |
| Diagrama de “CU-12” 14.....            | 28 |
| Diagrama de “CU-13” 15.....            | 28 |
| Diagrama de “CU-14” 16.....            | 29 |
| Diagrama de “CU-15” 17.....            | 29 |
| Diagrama de Frontera 18.....           | 30 |
| Diagrama posible arquitectura 19.....  | 31 |
| Modelo E-R 20.....                     | 34 |
| Tablas finales del sistema 21.....     | 37 |
| Storyboard inicio sesión 22.....       | 39 |
| Storyboard ver noticias 23.....        | 40 |
| Storyboard compartir noticias 24.....  | 41 |
| Storyboard clasificación 25.....       | 42 |
| Storyboard calendario 26.....          | 43 |
| Storyboard plantilla 27.....           | 44 |
| Storyboard patrocinadores 28.....      | 45 |
| Storyboard contacto 29.....            | 46 |
| Storyboard carné abonado 30.....       | 47 |
| Storyboard perfil 31.....              | 48 |
| Storyboard editar perfil 32.....       | 49 |
| Storyboard cerrar sesión 33.....       | 50 |
| Storyboard escaneo QR 34.....          | 51 |
| Storyboard QR manual 35.....           | 52 |

|                                                |    |
|------------------------------------------------|----|
| Storyboard asistencia 36 .....                 | 53 |
| Storyboard taquilla 37 .....                   | 54 |
| Wireframe de "Login" 38 .....                  | 55 |
| Wireframe de "Registro" 39.....                | 56 |
| Wireframe de "Home" 40 .....                   | 57 |
| Wireframe de "Side Menu" 41.....               | 58 |
| Wireframe de "Noticias" 42 .....               | 59 |
| Wireframe de "Detalle Post" 43.....            | 60 |
| Wireframe de "Carné Abonado" 44.....           | 61 |
| Wireframe de "Plantilla" 45 .....              | 62 |
| Wireframe de "Mi Perfil" 46 .....              | 63 |
| Wireframe de "Actualizar Perfil" 47 .....      | 64 |
| Wireframe de "Clasificación" 48.....           | 65 |
| Wireframe de "Patrocinadores" 49.....          | 66 |
| Wireframe de "Calendario y Resultados" 50..... | 67 |
| Wireframe de "Contacto" 51.....                | 68 |
| Wireframe de "Escaneo QR" 52.....              | 69 |
| Wireframe de "Taquilla" 53 .....               | 70 |
| Wireframe de "Asistencia" 54.....              | 71 |
| Pizarra Figma "paleta de colores" 55.....      | 72 |
| Pizarra Figma "Fuente" 56 .....                | 73 |
| Pizarra Figma "Iconografía" 57 .....           | 74 |
| Pizarra Figma "Botones" 58 .....               | 75 |
| Pizarra Figma "Logotipos" 59.....              | 75 |
| Logo Symfony 2.60 .....                        | 78 |
| Logo IONIC 2.61 .....                          | 78 |
| Logo PhpStorm 2.62 .....                       | 79 |
| IDE Android Studio 2.63 .....                  | 79 |
| Logo XCode IDE 2.64 .....                      | 80 |
| Logo NGINX 65.....                             | 80 |
| Logo Bitbucket 2.66 .....                      | 80 |
| Diagrama arquitectura implementada 67.....     | 82 |
| Figura código de "Escaneo QR" 68 .....         | 83 |
| Figura código de "Escaneo QR" 69 .....         | 83 |
| Figura código de "Escaneo QR" 70 .....         | 84 |
| Figura código de "Mostrar Jornadas" 71.....    | 84 |
| Figura código de "Mostrar Asistencia" 72 ..... | 85 |
| Figura código de "Mostrar Taquilla" 73.....    | 85 |
| Figura código de "Mostrar Noticias" 74.....    | 86 |
| Logo SonarQube 75.....                         | 87 |
| Figura Inicio SonarQube 5.76.....              | 87 |
| Figura SonarQube en funcionamiento 5.77 .....  | 88 |
| Figura SonarQube en funcionamiento 78.....     | 89 |
| Figura SonarQube resultados 79.....            | 90 |
| Figura SonarQube resultados 80.....            | 90 |
| Figura SonarQube resultados 81.....            | 91 |

|                                                 |     |
|-------------------------------------------------|-----|
| Figura SonarQube resultados reparados 82.....   | 91  |
| Pruebas manuales de Inicio de sesión 83 .....   | 92  |
| Pruebas manuales Escaneo de entradas 84 .....   | 93  |
| Pruebas manuales Asistencia 85.....             | 94  |
| Pruebas manuales Taquilla 86 .....              | 94  |
| Pruebas manuales Editar perfil 87 .....         | 95  |
| Pruebas manuales noticias 88 .....              | 95  |
| Pruebas manuales Clasificación 89.....          | 96  |
| Pruebas manuales Calendario y partidos 90 ..... | 96  |
| Pruebas manuales plantilla 91 .....             | 97  |
| Pruebas manuales Patrocinadores 92 .....        | 97  |
| Pruebas manuales Contacto 93.....               | 98  |
| Pruebas manuales Salir 94.....                  | 98  |
| Pruebas manuales Cerrar sesión 95 .....         | 99  |
| Captura docker-compose.yml 96 .....             | 104 |
| Captura docker-compose.yml 97 .....             | 105 |
| Figura Google Play Console 98 .....             | 120 |
| Figura de Android Studio 99 .....               | 121 |
| Figura de Android Studio 100 .....              | 122 |
| Figura de Google Play Console 101 .....         | 122 |
| Figura de Google Play Console 102.....          | 123 |
| Figura de Google Play Console 103.....          | 123 |
| Figura de Google Play Console 104.....          | 124 |
| Figura de Google Play Console 105.....          | 125 |
| Figura de Google Play Console 106.....          | 126 |
| Figura de Google Play Console 107 .....         | 126 |
| Figura de Google Play Console 108.....          | 127 |
| Figura de Google Play Console 109.....          | 128 |

## **1. Introducción**

En este primer capítulo se introduce al lector en el contexto y motivación en el cual se encuadra el ámbito del proyecto. Además de lo anterior, se especifican los objetivos que se pretenden alcanzar con el desarrollo de éste, así como la estructura de este trabajo fin de grado para situar al lector y permitirle una mejor comprensión de este escrito. Por último, se incluye una planificación temporal de las tareas previstas para lograr el objetivo del proyecto.

### **1.1 Contexto**

El Oleoinnova Mengíbar FS es un club de fútbol sala con una gran base de aficionados y abonados a nivel provincial. Éste dispone de una herramienta web para gestionar sus operaciones internas a nivel back y una aplicación móvil para ofrecer información a los socios y simpatizantes. La aplicación móvil se encuentra ya obsoleta en las tiendas móviles (Apple Store y Play Store). El club pretende incorporar nuevos servicios a la parte back de gestión, así como desarrollar una nueva aplicación móvil que incorpore dichos servicios para la administración del club y los aficionados. Ante esta necesidad, el club se enfrenta al desafío de modernizar y optimizar su sistema de gestión mediante una nueva solución. Entre las mejoras que proponen se encuentra crear nuevos servicios que permita ampliar funcionalidades de gestión tanto a los abonados/aficionados, como a la administración general del club, tales como gestión de entradas, abonos, información personalizada, compras etc.

### **1.2 Motivación**

Esta propuesta surge de la necesidad que se ha detectado en el club de contar con una solución innovadora que mejore la experiencia de los seguidores y faciliten las operaciones internas del club. Actualmente, como hemos comentado, el club cuenta ya con una aplicación móvil obsoleta y que no cubre todos los servicios que el club desea ofrecer al aficionado. Para ello se pretende modernizar las herramientas existentes construyendo nuevos servicios sobre los existentes y una nueva aplicación móvil. La aplicación permitirá a dichos aficionados acceder a todo tipo de información en tiempo real sobre fichajes, partidos clave, comunicados oficiales y mucho más, además de nuevas funcionalidades como utilizar un código QR para agilizar el acceso al estadio, eliminando los tickets físicos y abonos. Para el personal de club, sería una herramienta eficiente para gestionar el conteo de personas al estadio, dinero recaudado y estadísticas de acceso de socios para personalizar ofertas a dichos socios.

## 1.2 Objetivos

El objetivo principal de este Trabajo de Fin de Grado es crear una aplicación móvil y sus servicios back asociados, la cual cumpla con las demandas del Oleoinnova Mengíbar FS, ofreciendo nuevas herramientas digitales que mejoren las relaciones entre el club y sus seguidores y la gestión del día a día de club. Algunos de estos objetivos incluyen:

- Revisión bibliográfica estado del arte del desarrollo actual de aplicaciones móviles y servicios.
- Análisis y diseño de la nueva aplicación móvil y servicios de gestión para el club siguiendo los estándares de la industria de la ingeniería del software.
- Desarrollo de los nuevos servicios y adecuación de estos a los ya existentes.
- Desarrollo de la aplicación móvil utilizando un estándar de la industria que permita reutilizar el código y construir la aplicación para las 2 tiendas más importantes.
- Interconexión entre los servicios existentes en back y la nueva aplicación móvil.
- Pruebas de calidad.

## 1.3 Estructura del trabajo

A continuación, se realizará una breve descripción de los capítulos en los que está estructurado este proyecto, así como los contenidos expuestos en los mismos.

### Capítulo 2. Análisis del sistema

El capítulo 2 nos presenta las diferentes etapas del análisis de un sistema software llevadas a cabo en este proyecto. En él, incluimos un análisis de la gestión deportiva actual que lleva el club, así como una especificación de requerimientos, los casos de uso planteados y por último la posible arquitectura del sistema.

### Capítulo 3. Diseño del sistema

En el capítulo 3, se expone los modelos de datos utilizados para la solución y sus tablas, así como el diseño UX/UI elegidos para la misma (Guía de estilos, StoryBoards, Wireframes etc.).

### Capítulo 4. Implementación del sistema

En este capítulo vamos a presentar cómo ha sido llevada a cabo la implementación de la solución. Nos centraremos en mostrar las herramientas elegidas y cómo ha sido el proceso de construcción de ésta.



## **Capítulo 5. Pruebas y Ejemplos**

El capítulo 5, nos presenta las pruebas realizadas al código fuente de la aplicación y cómo estas pruebas han supuesto una mejora para el mismo.

## **Capítulo 6. Conclusiones y trabajos futuros**

El capítulo 6 presenta las conclusiones obtenidas del trabajo realizado y los posibles trabajos futuros resultantes de este trabajo.

## **Bibliografía y Anexos**

Por último, vamos a presentar la bibliografía consultada para el desarrollo del proyecto y los anexos con la información de los manuales de usuario e instalación.

### **1.4 Planificación temporal**

Seguidamente vamos a presentar la planificación temporal estimada para la construcción de la solución planteada en este Trabajo Fin de Grado.

Se han considerado las fechas de inicio del proyecto cuando se obtuvo la resolución favorable por parte del EPS y la fecha de fin proyecto para poder presentar el proyecto en fecha para la defensa ordinaria.

- Fecha de inicio del proyecto: 18 de Noviembre de 2024
- Fecha del fin del proyecto: 5 de Mayo de 2025

Las tareas definidas corresponden con las diferentes etapas en los procesos de ingeniería del software que van a ser llevados a cabo en este trabajo fin de grado. La mayoría de las tareas se ejecutan de manera secuencial, aunque hay alguna que hemos podido solapar.

A continuación, presentamos las tareas definidas y el diagrama de Gantt asociado a las mismas:

#### **1.4.1 Tareas**

1. Definición de requisitos / Alcance del proyecto: 7 días
2. Revisión del estado del arte: 7 días
3. Análisis de la aplicación y servicios actuales: 7 días
4. Diseño de la arquitectura (back y front): 12 días
5. Diseño interfaz gráfica (Mockup / Wireframes): 11 días

6. Adecuación y migración de los servicios ya existentes: 10 días
7. Desarrollo de los nuevos servicios: 25 días
8. Interconexión de los servicios nuevos con los ya existentes: 5 días
9. Configuración del entorno de desarrollo móvil: 5 días
10. Desarrollo de las funcionalidades de información para los aficionados: 10 días
11. Implementación de las funcionalidades para los administradores: 20 días
12. Integración con los servicios del backend: 10 días
13. Optimización del código: 4 días
14. Pruebas unitarias y funcionales: 6 días
15. Pruebas de la app en dispositivos reales: 10 días
16. Corrección de errores y ajustes finales: 4 días
17. Redacción de la memoria y manual del usuario: 12 días
18. Preparación de la presentación y la defensa: 14 días

Una vez revisadas las tareas definidas, pasamos a presentar el diagrama de Gantt que las contiene.

### 1.4.2 Diagrama de Gantt

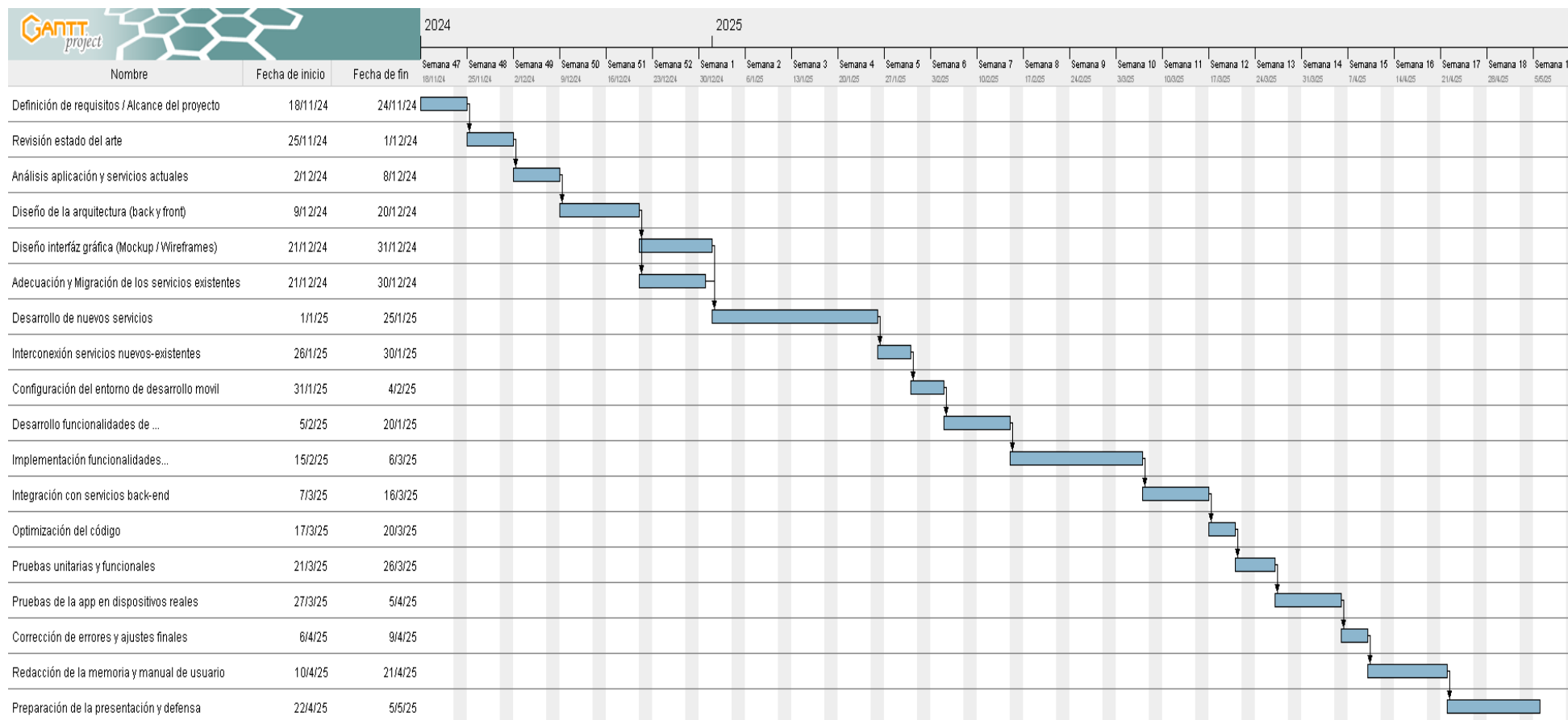


Diagrama de Gantt del proyecto 1

## 2. Análisis del sistema

En el siguiente apartado vamos a presentar el análisis del sistema llevado a cabo para la construcción de la aplicación siguiendo los conceptos principales de la Ingeniería del Software.

En él vamos a presentar un análisis del actual sistema de gestión deportiva implantado en el club y la especificación de requerimientos captados con el cliente(club) habiendo intentado captar toda la máxima información posible y trasladarla a la aplicación. Una vez revisados los requerimientos, se van a presentar los casos de uso que han salido de la toma de requerimientos llevada a cabo, los cuales nos van a servir para terminar de definir los flujos de trabajo de la aplicación en su relación con el usuario, los gestores y el sistema. Por último, vamos a realizar una revisión bibliográfica del estado del arte de tecnologías más destacadas utilizadas para la construcción de aplicaciones móviles con servicios en la nube y describiremos las más destacadas que vamos a utilizar en el desarrollo del sistema.

### 2.1 Análisis del sistema de gestión deportiva actual implantado en el club.

El Oleoinnova Mengíbar FS es un club deportivo de la provincia de Jaén, más concretamente de Mengíbar. Dicho club cuenta con distintas categorías y secciones deportivas. Todo ello se puede ver reflejado en su actual página web del club. <https://atleticomengibar.com/>

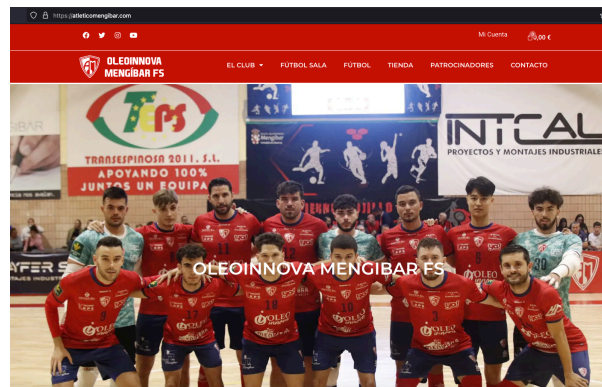


Figura web Oleoinnova Mengíbar 2

Dentro de la organización, la sección profesional con la que cuentan es la de fútbol sala. Actualmente el equipo milita en la categoría de 2ªB nacional, pero en los últimos años ha militado en 2ª División y ha estado a punto de ascender a 1ª División.

Existen otras secciones y categorías como el fútbol, pero estas se gestionan a nivel amateur al igual que las categorías inferiores y no es necesaria una gestión profesional al nivel que la del primer equipo de fútbol sala.

Hablando ya más concretamente de la *gestión deportiva actual del club de fútbol sala*, éste cuenta con una junta directiva, un director deportivo y la plantilla con su entrenador, preparador físico y jugadores. Además de todo ello, el club cuenta con socios que anualmente renuevan su abono para acceder a los partidos y con particulares que compran sus entradas y el patrocinio de distintas instituciones públicas y empresas privadas.

La entidad actualmente tiene un sistema de gestión web obsoleto de conteo de entradas y abonos y gestión de información de jugadores, entrenadores y equipos etc. La página web que utilizan y la app que usan para la gestión no cumplen ya con los estándares actuales de seguridad y además las funcionalidades para las que inicialmente fueron diseñadas han quedado pequeñas tras el gran crecimiento sufrido por la entidad en los últimos años.

El sistema que se utiliza guarda la información sobre jugadores, entrenadores y equipos tanto del primer equipo como de los equipos competidores. Asimismo, permite realizar la gestión económica del conteo de entradas y abonos, aunque el método es tedioso y costoso en tiempo, lo que provoca a veces grandes colas en los accesos. Las distintas categorías, no llevan gestión económica a nivel entradas y por tanto no se lleva registro de ello.

En la gestión de un club de fútbol, es muy importante el apartado de patrocinadores de este, pues son estos los que alimentan económicamente a la entidad para poder seguir creciendo. En el caso del Atlético Mengíbar, en su sistema cuentan con un apartado para ello, pero también está obsoleto y requiere ser modernizado e incorporado a la app para permitir una gestión más eficiente de los mismos y más automatizada.

Para la generación de entradas y envío a imprenta se utiliza el sistema web actual, pero se ha visto que no funciona correctamente y que a la hora de conectar con la app tiene ya dificultades y no es seguro.

El club no requiera una gran gestión informática, pero es muy necesario el control de la información que aparece en la web, así como la gestión de entradas y abonados del club. Es por ello, que se decidió apostar por la renovación del sistema y como socio y abonado le ofrecí la posibilidad de realizar dichas mejoras como trabajo fin de grado de manera totalmente gratuita con apoyo de la entidad MENGISOFT la cual hizo el primer software.

Tras analizar cómo el club lleva actualmente la gestión, pasamos a presentar los requerimientos obtenidos de las sesiones de entrevistas llevadas a cabo con el personal directivo del club.

## 2.2 Especificación de los requerimientos del sistema

En la revisión bibliográfica llevada a cabo para este trabajo sobre ingeniería del software [1][2][3][5], hemos encontrado una definición sobre cómo llevar a cabo correctamente un proceso de especificación de requerimientos. Dicha definición viene dada por R.S.Pressman en [1], en la cual se indica “que el análisis de los requisitos de un problema es una tarea propia de la ingeniería del software, la cual cubre el hueco entre la definición del software a nivel sistema y el diseño del software”. Pressman afirma que “ésta puede dividirse en cinco áreas de esfuerzo” las cuales enumeramos a continuación [1]:

- Reconocimiento del problema
- Evaluación y síntesis
- Modelado
- Especificación
- Revisión

Como definición general, podemos afirmar que un “proceso de toma de requisitos software consiste en lograr obtener la información sobre lo que se desea que el sistema haga y cómo poder hacerlo” [1]. Pressman indica en [1] que “los requisitos funcionales definen qué debe hacer el sistema, mientras que los requisitos no funcionales definen cómo debe de ser el sistema”.

Tras la revisión llevada a cabo en la literatura [1][2][3][5], podemos afirmar que realizar una correcta toma de requerimientos siempre produce muchos menos errores e incertidumbre a la hora de realizar el diseño del software. Este proceso facilita la labor de los desarrolladores que vayan a realizar el trabajo, ya que éstos pueden desarrollar con menor dificultad lo que necesita el cliente gracias a dichos requerimientos.

Por último, indicar que hemos constatado que la forma más habitual de obtener dichos requerimientos suele ser una reunión o entrevista preliminar según los revisado en la bibliografía.

### 2.2.1 Requerimientos funcionales

En este apartado describiremos las diferentes funcionalidades que va a proporcionar nuestra aplicación para satisfacer las necesidades tanto del usuario como del administrador. Dichos requerimientos han sido obtenidos de diversas entrevistas llevadas a cabo con el presidente de la entidad (Juan La Virgen Gómez) y el director deportivo (José García López). A continuación, mostraremos los principales requerimientos funcionales (FR por sus siglas en inglés de *functional requirement*) [1] que hemos identificado tras el análisis de necesidades y requisitos realizado:

- FR-01: Los usuarios deben poder iniciar sesión con sus credenciales de socio ya registradas al abonarse.
- FR-02: La aplicación debe proporcionar una opción para recuperar la contraseña mediante su correo electrónico en caso de que la olviden o la pierdan.
- FR-03: Los usuarios registrados deben poder ver y actualizar su perfil de usuario con su nombre, foto y demás datos.
- FR-04: La aplicación debe mostrar un listado de las últimas noticias que se componga de imagen, título y descripción de la noticia.
- FR-05: Los usuarios deben poder compartir las noticias a través de redes sociales mediante un botón al lado de la noticia.
- FR-06: La aplicación debe permitir la búsqueda de noticias filtrando por palabras clave.
- FR-07: Los administradores deben poder validar las entradas mediante un lector de QR que debe tener tanto la opción de escaneo como la opción de introducir el código de la entrada manualmente.
- FR-08: Los administradores deben poder seleccionar el evento en el que se está realizando el escaneo de entradas, de manera que cada entrada validada quede correctamente asociada a dicho evento.
- FR-09: Los administradores deben poder acceder al conteo de asistencia de un evento concreto, permitiéndoles filtrar por el tipo de entrada.
- FR-10: Los administradores deben poder acceder a una vista detallada de la recaudación de la taquilla en un evento concreto, especificando de donde viene cada parte del dinero recaudado.
- FR-11: Los usuarios deben poder acceder a una ficha con todos sus carnés de abonado con información del usuario y del propio abono (tipo de abono, temporada, QR, código manual, etc.).
- FR-12: La aplicación debe mostrar una vista actualizada de la clasificación de la liga, además, con un código de colores que asocie la posición de cada equipo a los puestos de ascenso y descenso que sea acorde a la temporada y división que se está jugando.
- FR-13: La aplicación debe mostrar un listado de todos los partidos (ya jugados, y sin jugar) de la temporada activa, mostrando los equipos y el resultado o fecha de dichos partidos.
- FR-14: La aplicación debe mostrar un listado de la actual plantilla activa del equipo, mostrando el rol, dorsal, nombres y foto de los jugadores.
- FR-15: La aplicación debe mostrar un listado de todos los patrocinadores del equipo agrupados según el tipo que sean. Además, debe permitir navegar hasta la página oficial de cada patrocinador haciendo clic en el nombre, en caso de que tengan una.

- FR-16: La aplicación debe mostrar una lista de datos de contacto, así como las redes sociales del equipo, además, de un mapa interactivo en el que se pueda navegar y se muestre la ubicación de la sede del club.
- 

### 2.2.2 Requerimientos no funcionales

En este apartado describiremos los requerimientos no funcionales, por definición [1], un requerimiento no funcional (NFR por sus siglas en inglés de *nonfunctional requirement*) puede describirse como un atributo de calidad, un atributo de rendimiento, un atributo de seguridad o una restricción general del sistema. Estos requisitos no funcionales responden a la pregunta de cómo debe comportarse el sistema, en lugar de qué debe hacer el sistema. En nuestro caso hemos definido los siguientes:

- NFR-01: La aplicación no debe tardar más de 3 segundos en cargar en dispositivos con conexión y condiciones normales (conexión WiFi o 4G).
- NFR-02: La aplicación debe ser escalable para soportar futuros cambios en funcionalidad y número de usuarios.
- NFR-03: La comunicación entre aplicación y servicio web debe realizarse mediante HTTPS por seguridad.
- NFR-04: La aplicación debe ser compatibles con dispositivos Android a partir de la versión 5.1, y estar dirigida a dispositivos de hasta la versión 14 de Android.
- NFR-05: La interfaz debe cumplir con los mínimos de accesibilidad especificados en la ley EAA.
- NFR-06: La aplicación debe estar optimizada y adaptada para móviles y tabletas con diferentes sistemas operativos y resoluciones de pantallas.
- NFR-07: La aplicación solo debe estar disponible para su uso en modo vertical y restringir su uso en horizontal.

Una vez revisados todos los requerimientos obtenidos de las entrevistas con personal del club, vamos a pasar a presentar los distintos casos de uso que se desprenden de dichos requerimientos.



## 2.3 Casos de uso

Siguiendo con lo estudiado en la bibliografía [1][5], vamos a presentar los casos de uso más destacados que hemos establecido para el desarrollo de la aplicación son los siguientes:

|                   |                                                                                                                                                                                                                                           |
|-------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Caso de Uso       | <b>Iniciar Sesión – “CU-01”</b>                                                                                                                                                                                                           |
| Actores           | Usuario                                                                                                                                                                                                                                   |
| Precondiciones    | - El usuario debe estar ya registrado en el sistema.                                                                                                                                                                                      |
| Flujo Principal   | 1. El usuario accede a la aplicación.<br>2. El usuario pulsa en “Iniciar sesión”.<br>3. El usuario introduce sus credenciales en el formulario.<br>4. El usuario pulsa “Acceder”.<br>5. El sistema verifica las credenciales del usuario. |
| Flujo Alternativo | - Si el usuario introduce credenciales incorrectas se muestra un mensaje de error.                                                                                                                                                        |
| Postcondiciones   | - El usuario accede a su cuenta si las credenciales son correctas.                                                                                                                                                                        |
| Resumen           | Permite a los usuarios iniciar sesión con sus credenciales en la aplicación.                                                                                                                                                              |

Visual Paradigm Standard (jg0023/Universidad de Jaén)

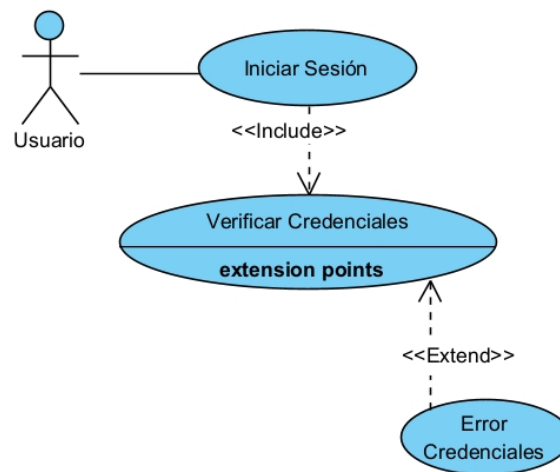


Diagrama de “CU-01” 2.3

|                   |                                                                                                                                                                                                                                                        |
|-------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Caso de Uso       | <b>Recuperar Contraseña – “CU-02”</b>                                                                                                                                                                                                                  |
| Actores           | Usuario                                                                                                                                                                                                                                                |
| Precondiciones    | - El usuario debe estar ya registrado en el sistema.<br>- El usuario debe haber olvidado la contraseña.                                                                                                                                                |
| Flujo Principal   | 1. El usuario accede a la aplicación.<br>2. El usuario pulsa en “Iniciar sesión”.<br>3. El usuario pulsa en “Recordar mi contraseña”.<br>4. El sistema envía un correo al email asociado del usuario con un enlace para poder recuperar la contraseña. |
| Flujo Alternativo | Ninguno.                                                                                                                                                                                                                                               |

|                 |                                                                                                                                       |
|-----------------|---------------------------------------------------------------------------------------------------------------------------------------|
| Postcondiciones | - El sistema envía un correo electrónico con un enlace para restablecer su contraseña.                                                |
| Resumen         | Permite a los usuarios que haya olvidado su contraseña, recuperarla a través de un enlace enviado por el correo electrónico asociado. |

Visual Paradigm Standard (jg00023(Universidad de Jaén))

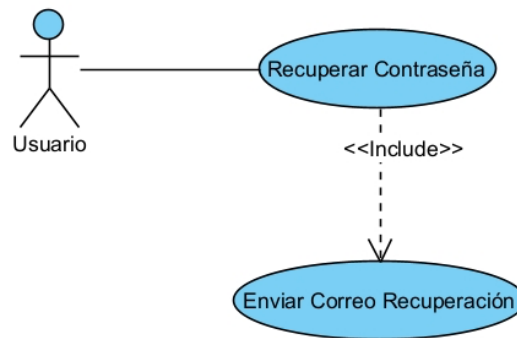


Diagrama de “CU-02” 2.4

|                   |                                                                                                                                                                                                                                                                                                                                                            |
|-------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Caso de Uso       | <b>Actualizar perfil de usuario – “CU-03”</b>                                                                                                                                                                                                                                                                                                              |
| Actores           | Usuario                                                                                                                                                                                                                                                                                                                                                    |
| Precondiciones    | - El usuario debe haber iniciado sesión previamente.                                                                                                                                                                                                                                                                                                       |
| Flujo Principal   | <ol style="list-style-type: none"> <li>1. El usuario pulsa en “Mi perfil”.</li> <li>2. El usuario pulsa en “Actualizar perfil”.</li> <li>3. El usuario rellena el formulario con los datos que desea modificar.</li> <li>4. El usuario pulsa en “Confirmar cambios”.</li> <li>5. El sistema actualiza el perfil del usuario con dichos cambios.</li> </ol> |
| Flujo Alternativo | Ninguno.                                                                                                                                                                                                                                                                                                                                                   |
| Postcondiciones   | - Se actualizan los datos del perfil de usuario.                                                                                                                                                                                                                                                                                                           |
| Resumen           | Permite al usuario modificar su información personal mediante un formulario en el que aparecen todos los campos que se pueden modificar.                                                                                                                                                                                                                   |

Visual Paradigm Standard (jg00023(Universidad de Jaén))

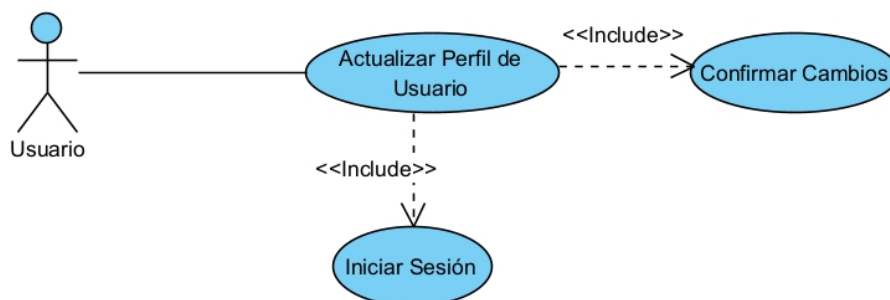


Diagrama de “CU-03” 5

|                   |                                                                                                                                                                                                                                                         |
|-------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Caso de Uso       | <b>Ver y compartir noticias – “CU-04”</b>                                                                                                                                                                                                               |
| Actores           | Usuario                                                                                                                                                                                                                                                 |
| Precondiciones    | Ninguna.                                                                                                                                                                                                                                                |
| Flujo Principal   | 1. El usuario pulsa en “Noticias”<br>2. El usuario pulsa en “Ver” o en “Compartir” en la noticia que desee.                                                                                                                                             |
| Flujo Alternativo | Ninguno.                                                                                                                                                                                                                                                |
| Postcondiciones   | - Si se pulsa en “Ver” se muestra una vista más detallada de la noticia.<br>- Si se pulsa en compartir, se abre una ventana que permite compartir en enlace, o compartirlo en las diferentes aplicaciones de redes sociales (Whatsapp, Instagram, ...). |
| Resumen           | Permite a los usuarios visualizar y compartir las noticias del equipo que aparecen dentro de la vista “Noticias”.                                                                                                                                       |

Visual Paradigm Standard (jg00023(Universidad de Jaén))

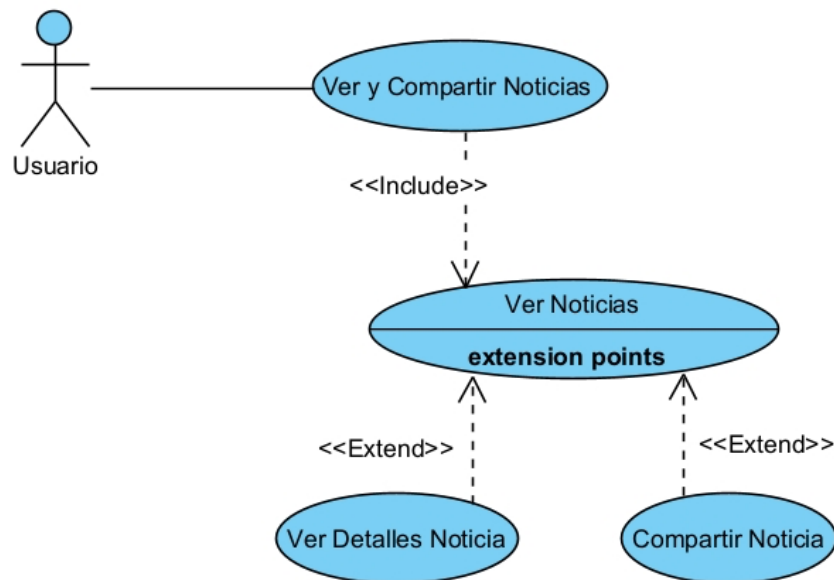


Diagrama de “CU-04” 6

|                   |                                                                                                                                                                                                                                               |
|-------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Caso de Uso       | <b>Buscar noticias por palabras clave – “CU-05”</b>                                                                                                                                                                                           |
| Actores           | Usuario                                                                                                                                                                                                                                       |
| Precondiciones    | Ninguna.                                                                                                                                                                                                                                      |
| Flujo Principal   | 1. El usuario pulsa en “Noticias”.<br>2. El usuario pulsa en la barra de búsqueda.<br>3. El usuario escribe la palabra o palabras por las que quiera filtrar.<br>4. El sistema muestra solo las noticias que contengan las palabras escritas. |
| Flujo Alternativo | - Si ninguna noticia contiene las palabras clave buscadas, se muestra un mensaje de que no hay noticias.                                                                                                                                      |

|                 |                                                                                                                          |
|-----------------|--------------------------------------------------------------------------------------------------------------------------|
| Postcondiciones | - Se muestra un listado con las noticias filtradas por las palabras ingresadas en el buscador de noticias                |
| Resumen         | Permite a los usuarios buscar noticias específicas mediante palabras clave en el buscador dentro de la vista “Noticias”. |

Visual Paradigm Standard (jg00023(Universidad de Jaén))

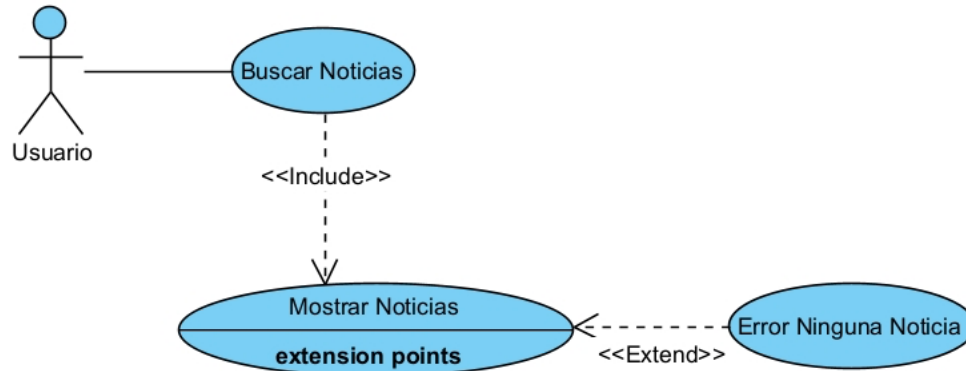
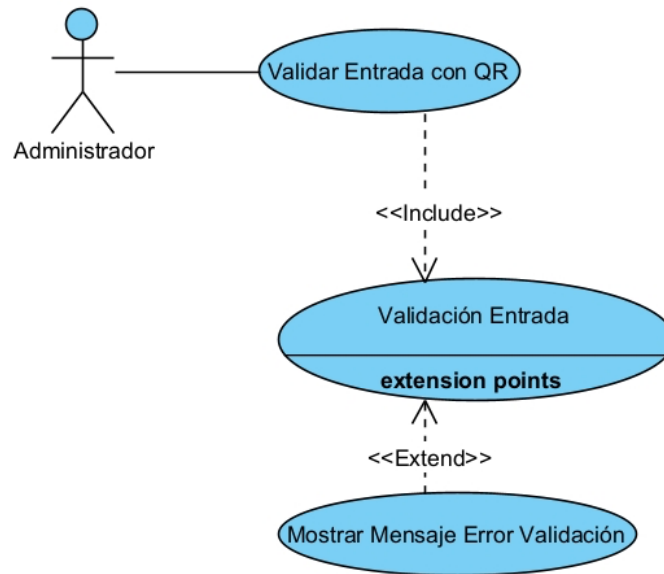


Diagrama de “CU-05” 7

|                   |                                                                                                                                                                                                                                                                                                                                                                                                           |
|-------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Caso de Uso       | <b>Validar entradas con código QR – “CU-06”</b>                                                                                                                                                                                                                                                                                                                                                           |
| Actores           | Administrador                                                                                                                                                                                                                                                                                                                                                                                             |
| Precondiciones    | - Debe existir un evento creado con la fecha y horas establecidas para el momento en que se va a escanear.                                                                                                                                                                                                                                                                                                |
| Flujo Principal   | <ol style="list-style-type: none"> <li>1. El administrador pulsa en “Escanear Entradas”.</li> <li>2. El administrador selecciona el evento del que quiere escanear las entradas.</li> <li>3. El administrador pulsa en “Escanear”.</li> <li>4. El administrador enfoca el código QR con la cámara del móvil.</li> <li>5. El sistema devuelve el mensaje de validación del código por pantalla.</li> </ol> |
| Flujo Alternativo | - Si el código QR es incorrecto o inválido, se muestra un mensaje de error con el motivo por el cual no se permite la entrada.                                                                                                                                                                                                                                                                            |
| Postcondiciones   | - Se valida el código QR de la entrada o abono mediante el escaneo, y se muestra un mensaje de validación.                                                                                                                                                                                                                                                                                                |
| Resumen           | Permite a los administradores validar las entradas de un evento concreto mediante el escaneo de códigos QR.                                                                                                                                                                                                                                                                                               |



### Diagrama de “CU-06” 8

|                   |                                                                                                                                                                                                                                                                                                                                                                                                                  |
|-------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Caso de Uso       | <b>Ingresar código de entrada manualmente – “CU-07”</b>                                                                                                                                                                                                                                                                                                                                                          |
| Actores           | Administrador                                                                                                                                                                                                                                                                                                                                                                                                    |
| Precondiciones    | - Debe existir un evento creado con la fecha y horas establecidas para el momento en que se vaya a introducir el código.                                                                                                                                                                                                                                                                                         |
| Flujo Principal   | <ol style="list-style-type: none"> <li>1. El administrador pulsa en “Escanear Entradas”.</li> <li>2. El administrador selecciona el evento del que quiere escanear las entradas.</li> <li>3. El administrador pulsa en “Escaneo Manual”.</li> <li>4. El administrador ingresa el código manual y pulsa “Confirmar”.</li> <li>5. El sistema devuelve el mensaje de validación del código por pantalla.</li> </ol> |
| Flujo Alternativo | - Si el código es incorrecto o inválido, se muestra un mensaje de error con el motivo por el cual no se permite la entrada.                                                                                                                                                                                                                                                                                      |
| Postcondiciones   | - Se valida el código de la entrada y se muestra el mensaje de validación.                                                                                                                                                                                                                                                                                                                                       |
| Resumen           | Permite a los administradores validar las entradas de un evento concreto mediante la introducción manual del código de la entrada.                                                                                                                                                                                                                                                                               |

Visual Paradigm Standard (jg00023(Universidad de Jaén))

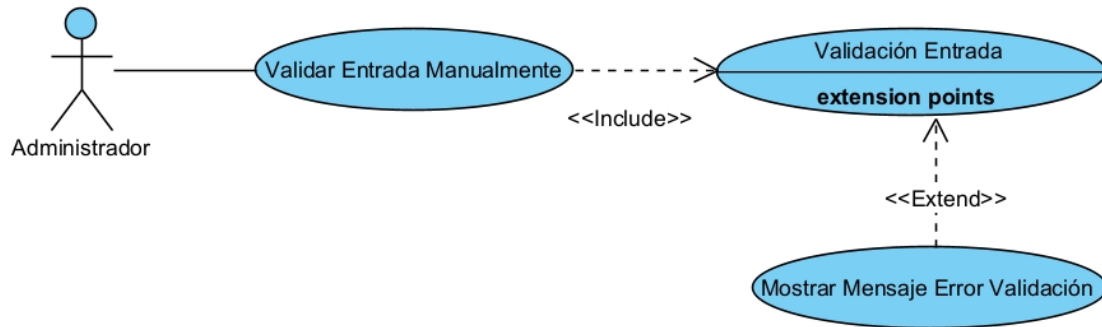


Diagrama de “CU-07” 9

|                   |                                                                                                                                                                                                                                                        |
|-------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Caso de Uso       | <b>Ver asistencia de eventos – “CU-08”</b>                                                                                                                                                                                                             |
| Actores           | Administrador                                                                                                                                                                                                                                          |
| Precondiciones    | - Debe existir un evento ya creado o acabado para poder seleccionarlo.                                                                                                                                                                                 |
| Flujo Principal   | 1. El administrador pulsa en “Asistencia”.<br>2. El administrador selecciona el evento del que quiera consultar la asistencia.<br>3. El administrador pulsa en “Consultar”.<br>4. El sistema muestra una lista con el conteo de asistentes del evento. |
| Flujo Alternativo | - Si el administrador selecciona un tipo de entrada, se mostrarán solo los asistentes con ese tipo de entrada                                                                                                                                          |
| Postcondiciones   | - Se muestra una lista del conteo de asistentes al evento seleccionado, con opción de filtrado por tipo de entrada.                                                                                                                                    |
| Resumen           | Permite a los administradores consultar las estadísticas de asistencia a un evento.                                                                                                                                                                    |

Visual Paradigm Standard (jg00023(Universidad de Jaén))

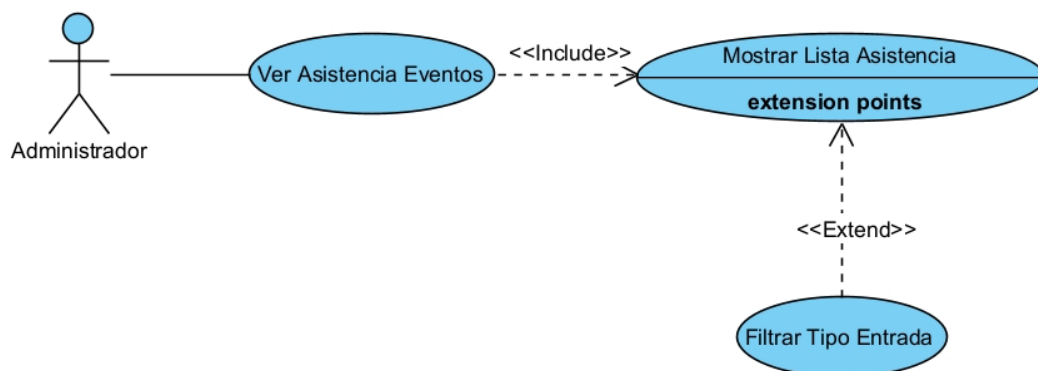


Diagrama de “CU-08” 10

|                   |                                                                                                                                                                                                                                                                                                                           |
|-------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Caso de Uso       | <b>Ver recaudación en taquilla de eventos – “CU-09”</b>                                                                                                                                                                                                                                                                   |
| Actores           | Administrador                                                                                                                                                                                                                                                                                                             |
| Precondiciones    | - Debe existir un evento ya creado o acabado para poder seleccionarlo.                                                                                                                                                                                                                                                    |
| Flujo Principal   | <ol style="list-style-type: none"> <li>1. El administrador pulsa en “Taquilla”.</li> <li>2. El administrador selecciona el evento del que quiera consultar la taquilla.</li> <li>3. El administrador pulsa en “Consultar”.</li> <li>4. El sistema muestra una lista con el recuento de la taquilla del evento.</li> </ol> |
| Flujo Alternativo | Ninguno.                                                                                                                                                                                                                                                                                                                  |
| Postcondiciones   | - Se muestra una lista de la recaudación total del evento seleccionado.                                                                                                                                                                                                                                                   |
| Resumen           | Permite a los administradores consultar los ingresos que se han obtenido en un evento evento.                                                                                                                                                                                                                             |

Visual Paradigm Standard(jg00023(Universidad de Jaén))

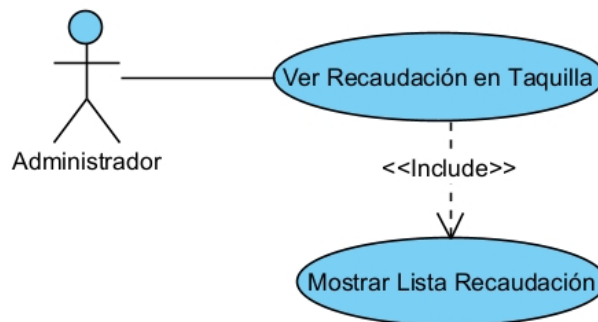


Diagrama de “CU-09” 11

|                   |                                                                                                                                                                                   |
|-------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Caso de Uso       | <b>Ver carnés de abonado – “CU-10”</b>                                                                                                                                            |
| Actores           | Usuario                                                                                                                                                                           |
| Precondiciones    | <ul style="list-style-type: none"> <li>- El usuario debe haber iniciado sesión previamente.</li> <li>- El usuario debe estar abonado y tener al menos un carné activo.</li> </ul> |
| Flujo Principal   | <ol style="list-style-type: none"> <li>1. El usuario pulsa en “Carné abonado”.</li> <li>2. El sistema muestra los diferentes abonos que tenga asociados el usuario.</li> </ol>    |
| Flujo Alternativo | - Si ha iniciado sesión, pero no tiene ningún carné activo se mostrará un mensaje diciendo que no hay carnés.                                                                     |
| Postcondiciones   | - Se muestra la información del abono, además del QR para poder escanearlo y el código para poder introducirlo manualmente.                                                       |
| Resumen           | Permite a los usuarios mostrar sus carnés de abonado con toda la información para el escaneo en la entrada de los eventos.                                                        |

Visual Paradigm Standard (jg00023(Universidad de Jaén))

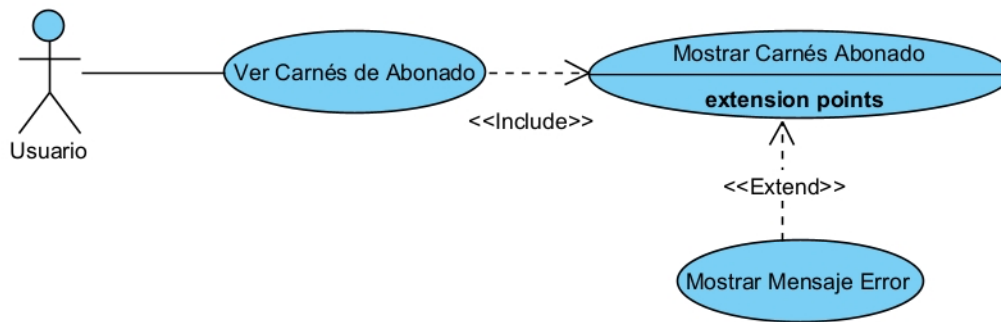


Diagrama de “CU-10” 12

|                   |                                                                                                      |
|-------------------|------------------------------------------------------------------------------------------------------|
| Caso de Uso       | <b>Ver clasificación de la liga – “CU-11”</b>                                                        |
| Actores           | Usuario                                                                                              |
| Precondiciones    | Ninguna.                                                                                             |
| Flujo Principal   | 1. El usuario pulsa en “Clasificación”.<br>2. El sistema muestra la clasificación actual de la liga. |
| Flujo Alternativo | Ninguno.                                                                                             |
| Postcondiciones   | - Se muestra un listado de la clasificación actual.                                                  |
| Resumen           | Permite a los usuarios consultar la clasificación actual de la liga.                                 |

Visual Paradigm Standard (jg00023(Universidad de Jaén))

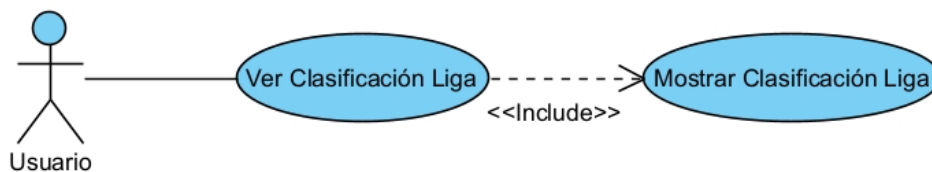


Diagrama de “CU-11” 13

|                   |                                                                                                     |
|-------------------|-----------------------------------------------------------------------------------------------------|
| Caso de Uso       | <b>Ver calendario de partidos – “CU-12”</b>                                                         |
| Actores           | Usuario.                                                                                            |
| Precondiciones    | Ninguna.                                                                                            |
| Flujo Principal   | 1. El usuario pulsa en “Calendario y partidos”.<br>2. El sistema muestra el calendario de partidos. |
| Flujo Alternativo | Ninguno.                                                                                            |
| Postcondiciones   | - Se muestra un listado del calendario de partidos.                                                 |
| Resumen           | Permite a los usuarios ver el calendario de partidos de la temporada.                               |



Visual Paradigm Standard (jg00023(Universidad de Jaén))

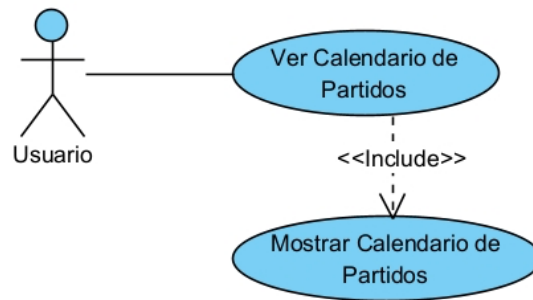


Diagrama de “CU-12” 14

|                   |                                                                                                       |
|-------------------|-------------------------------------------------------------------------------------------------------|
| Caso de Uso       | <b>Ver plantilla del equipo – “CU-13”</b>                                                             |
| Actores           | Usuario                                                                                               |
| Precondiciones    | Ninguna.                                                                                              |
| Flujo Principal   | 1. El usuario pulsa en “Plantilla”.<br>2. El sistema muestra al usuario la plantilla actual del club. |
| Flujo Alternativo | Ninguno.                                                                                              |
| Postcondiciones   | - Se muestra una lista con la plantilla actual del club.                                              |
| Resumen           | Permite a los usuarios consultar la plantilla actual del club con sus posiciones y dorsales.          |

Visual Paradigm Standard (jg00023(Universidad de Jaén))

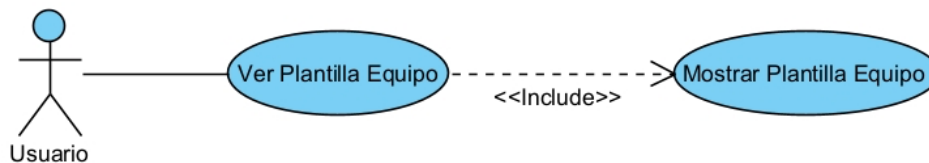


Diagrama de “CU-13” 15

|                   |                                                                                                                                    |
|-------------------|------------------------------------------------------------------------------------------------------------------------------------|
| Caso de Uso       | <b>Ver patrocinadores del equipo – “CU-14”</b>                                                                                     |
| Actores           | Usuario                                                                                                                            |
| Precondiciones    | Ninguna.                                                                                                                           |
| Flujo Principal   | 1. El usuario pulsa en “Patrocinadores”.<br>2. El sistema muestra al usuario una lista con los diferentes patrocinadores del club. |
| Flujo Alternativo | Ninguno.                                                                                                                           |
| Postcondiciones   | - Se muestra una lista con los patrocinadores actuales del club.                                                                   |
| Resumen           | Permite a los usuarios consultar los diferentes patrocinadores que apoyan actualmente al club.                                     |

Visual Paradigm Standard(jg0023(Universidad de Jaén))

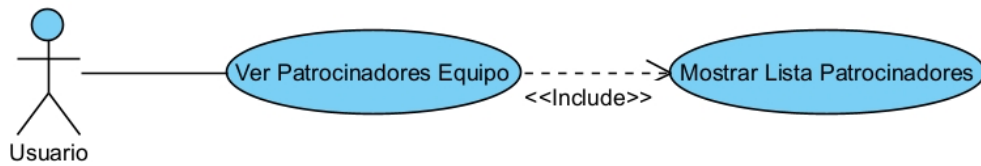


Diagrama de “CU-14” 16

|                   |                                                                                                                                   |
|-------------------|-----------------------------------------------------------------------------------------------------------------------------------|
| Caso de Uso       | <b>Ver contacto – “CU-15”</b>                                                                                                     |
| Actores           | Usuario.                                                                                                                          |
| Precondiciones    | Ninguna.                                                                                                                          |
| Flujo Principal   | 1. El usuario pulsa en “Contacto”.<br>2. El sistema muestra al usuario una lista con el contacto y las redes sociales del equipo. |
| Flujo Alternativo | Ninguno.                                                                                                                          |
| Postcondiciones   | - Se muestra una lista con los diferentes métodos de contacto del club además de las redes sociales.                              |
| Resumen           | Permite a los usuarios consultar los diferentes métodos de contacto del club además de las redes sociales.                        |

Visual Paradigm Standard(jg0023(Universidad de Jaén))

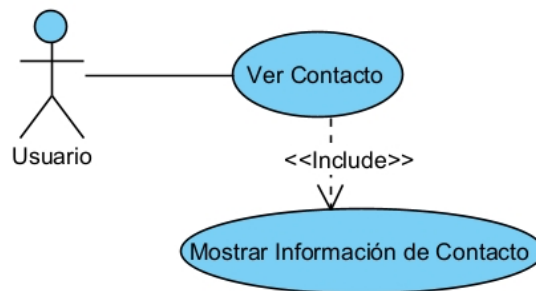


Diagrama de “CU-15” 17

Finalmente, tras presentar los diferentes casos de uso que tendremos en nuestro proyecto, presentamos el diagrama de frontera, compuesto de los actores y como interactúan con el sistema:



Diagrama de Frontera 18

## 2.4 Análisis posible arquitectura de la solución

Una vez vistos los casos de uso que implementaremos en la aplicación, analizaremos cómo será la posible estructura final de la aplicación, desde la aplicación hasta la base de datos.

Primero tendríamos la aplicación móvil, es decir, un cliente desde donde los usuarios van a interactuar con el sistema de diversas formas, por ejemplo, consultando noticias u obteniendo información de los partidos, y todo ello se hará a través de una interfaz de aplicación simple e intuitiva.

En segundo lugar, ese cliente tendría que enviar la información al servidor, que actuará como intermediario entre la aplicación y el resto del sistema. El servidor recibirá las peticiones del cliente, las interpretará y realizará la acción que necesite en función de los requerimientos.

En el servidor se encontrarían los servicios y la base de datos. Una vez recibidas las peticiones por parte del cliente, tendría que acceder a la base de datos,

para recoger o modificar los datos pertinentes, después los procesará, y devolverá una respuesta con la información que solicitó previamente el cliente.

Una representación esquemática y simple sería la siguiente:

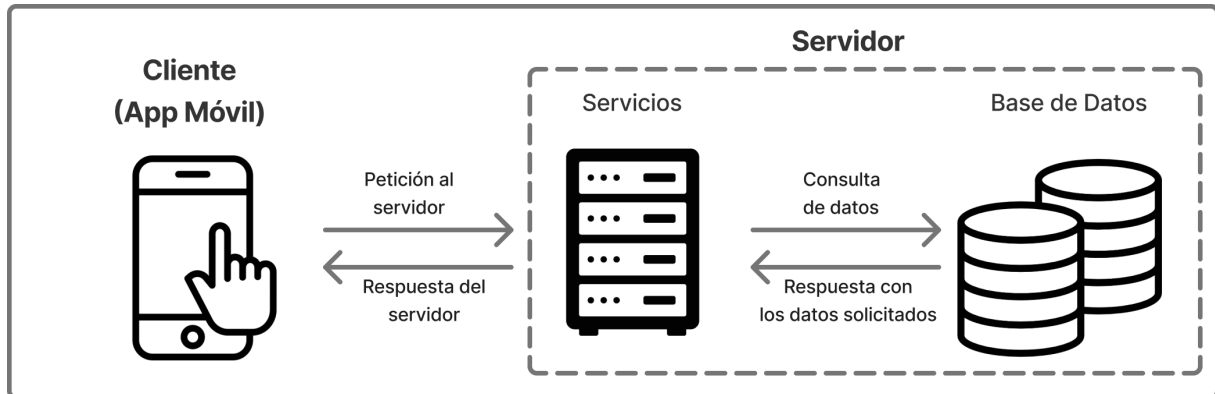


Diagrama posible arquitectura 19



### 3. Diseño del sistema

En la siguiente sección vamos a presentar el diseño del sistema que hemos planteado como solución para el desarrollo de la aplicación.

Se va a presentar el diseño elegido para la representación del modelo de datos que mejor se ha adaptado para la solución propuesta. De este modelo de datos, el cual está expresado usando el Modelo Entidad-Relación [4], se mostrarán las tablas resultantes de aplicar las distintas fases de normalización. Asimismo, vamos a mostrar los trabajos para la construcción de la interfaz gráfica. Para ello se han seguido las técnicas más modernas en el campo del desarrollo ágil mediante técnicas de UX/UI que permiten captar toda la información recogida en la fase de análisis y plasmarlas gráficamente. Gracias a esto, durante esta fase hemos ido prototipando diseños que han sido corregidos en tiempo real por parte del cliente, lo cual nos ha facilitado mucho el posterior desarrollo, puesto que la interfaz presentada ya se ajustaba a las necesidades de este. Por último, se ha realizado la guía de estilos para seguir en el desarrollo de la aplicación o si hay que realizar modificaciones en el futuro seguir la misma identidad visual.

## Escuela Politécnica Superior de Jaén



Seguidamente pasamos a describir las relaciones surgidas del modelo E-R

## RELACIONES DE LAS TABLAS

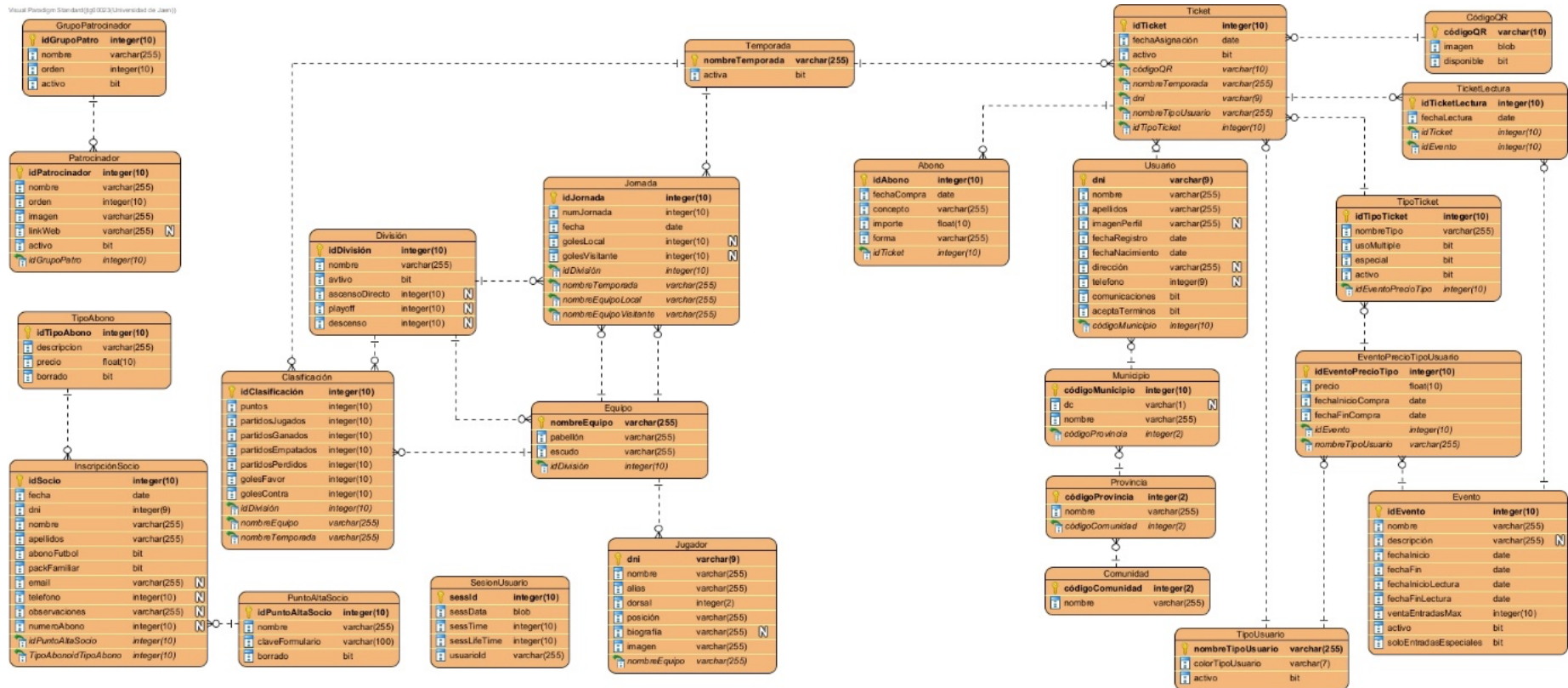
- R1 - “InscripciónSocio” tiene asociado “TipoAbono”: Cada inscripción que se hace de un socio está asociada a un tipo de abono específico.
- R2 - “InscripciónSocio” se realiza en “PuntoAltaSocio”: Cada inscripción que se hace de un socio se realizará en un punto de alta concreto.
- R3 - “Clasificación” pertenece a “División”: Cada clasificación que se hace sobre la liga pertenece a una división específica.
- R4 - “Clasificación” corresponde a “Temporada”: Cada clasificación que se hace sobre la liga corresponde a una temporada concreta.
- R5 - “Equipo” está en “Clasificación”: Cada equipo que compite en la división en una temporada está en la clasificación correspondiente.
- R6 - “Equipo” compite en “División”: Cada equipo compite en una división en una temporada concreta.
- R7 - “Equipo” disputa “Jornada”: Hay 2 equipos que disputan cada jornada, local y visitante.
- R8 - “Jugador” juega en “Equipo”: Un jugador juega para un único equipo en la competición.
- R9 - “Jornada” pertenece a “División”: Cada jornada pertenece a una división específica.
- R10 - “Jornada” forma parte de “Temporada”: Cada jornada forma parte de una temporada.
- R11 - “Ticket” válido en “Temporada”: Un ticket solo puede ser válido dentro de la temporada que tenga asociada.
- R12 - “Ticket” pertenece a “Abono”: Un ticket puede estar asociado a un abono adquirido por un socio.
- R13 - “CódigoQR” aparece en “Ticket”: Un código QR aparece en un ticket de entrada de evento.
- R14 - “Ticket” tiene asociado “Usuario”: Un ticket tiene asociado un único usuario que podrá usarlo.
- R15 - “Ticket” es de tipo “TipoTicket”: Un ticket tiene un tipo de ticket asignado.
- R16 - “Ticket” tiene asignado un “TipoUsuario”: Un ticket tiene asignado un tipo de usuario del usuario que tiene asociado.
- R17 - “TicketLectura” tiene un “Ticket”: Cada ticket de lectura tiene un ticket al que está vinculado.
- R18 - “Usuario” vive en “Municipio”: Cada usuario tiene asociado un municipio donde vive.
- R19 - “Municipio” pertenece a “Provincia”: Cada municipio pertenece a una provincia.



- R20 - “Provincia” pertenece a “Comunidad”: Cada provincia pertenece a una comunidad autónoma.
- R21 - “EventoPrecioTipoUsuario” tiene un precio asociado a “TipoUsuario”: Se establece un precio asociado para cada tipo de usuario en un evento concreto.
- R22 - “EventoPrecioTipoUsuario” pertenece a “Evento”: Cada configuración de precios está asociada a un evento concreto.
- R23 - “TicketLectura” ocurre durante “Evento”: Una lectura de un ticket ocurre durante un evento determinado.
- R24 - “TipoTicket” tiene asociado “EventoPrecioTipoUsuario”: Cada tipo de ticket tiene asociados unos precios en el evento para los distintos usuarios.
- R25 - “Patrocinador” pertenece a “GrupoPatrocinador”: Cada patrocinador pertenece a un grupo de patrocinadores en función de su tipo de patrocinio.

### 3.2 Tablas del sistema

Finalmente, tras presentar el modelo de datos basado en E-R y sus relaciones, vamos a presentar las tablas resultantes de aplicar el proceso de normalización de tablas descrito en [4] para realizar el análisis y gestión de datos. Dichas tablas serán posteriormente pasadas a un SGBD, en nuestro caso, MySQL [17].



Tablas finales del sistema 21

### 3.3 Diseño UX/UI aplicación móvil

A continuación, vamos a presentar los diseños UX/UI de la aplicación móvil utilizando las técnicas de Storyboards [20] y Wireframes [21]. Los Wireframes [21] son esquemas o planos de una pantalla o página. Éstos representan cómo se organiza el contenido, botones, menús, imágenes, etc. No tienen diseño final (colores bonitos ni imágenes reales), solo estructura. Los Storyboards [20] son como historietas que cuentan una historia. Sirven para mostrar el flujo de acciones: qué pasa primero, qué pasa después, paso por paso. Se enfocan en el recorrido del usuario o de la historia.

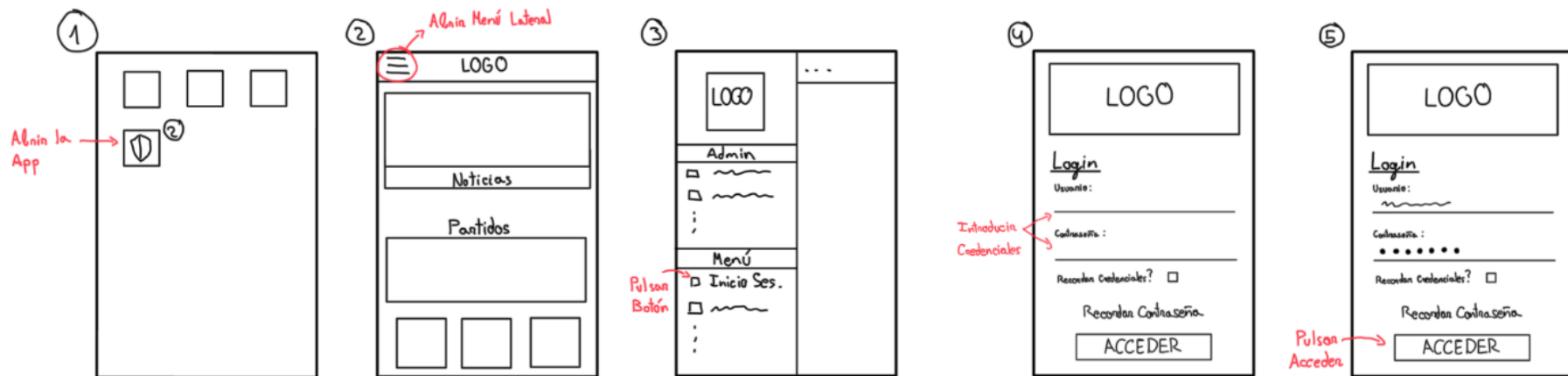
La importancia de este paso es fundamental en la construcción sólida de la aplicación, ya que, a partir de los storyboards, los wireframes y sus descripciones, somos capaces de plasmar la idea del cliente y los requerimientos en un diseño de funcionalidades que explique clara y concisamente lo que se pretende llevar a cabo. Gracias a esta técnica, se ahorran muchas horas de trabajo, ya que siempre se valida y retroalimenta con el cliente hasta cumplir con todas sus expectativas de requerimientos.

Tras presentar los storyboards y wireframes, veremos la guía de estilos que hemos establecido para el diseño, colores y tipos de letras que tendrá la aplicación, siempre teniendo en cuenta los requerimientos que imponen en sus tiendas Google y Apple [6][7].

### 3.3.1 StoryBoards

#### 3.3.1.1 Storyboard de cómo un usuario inicia sesión:

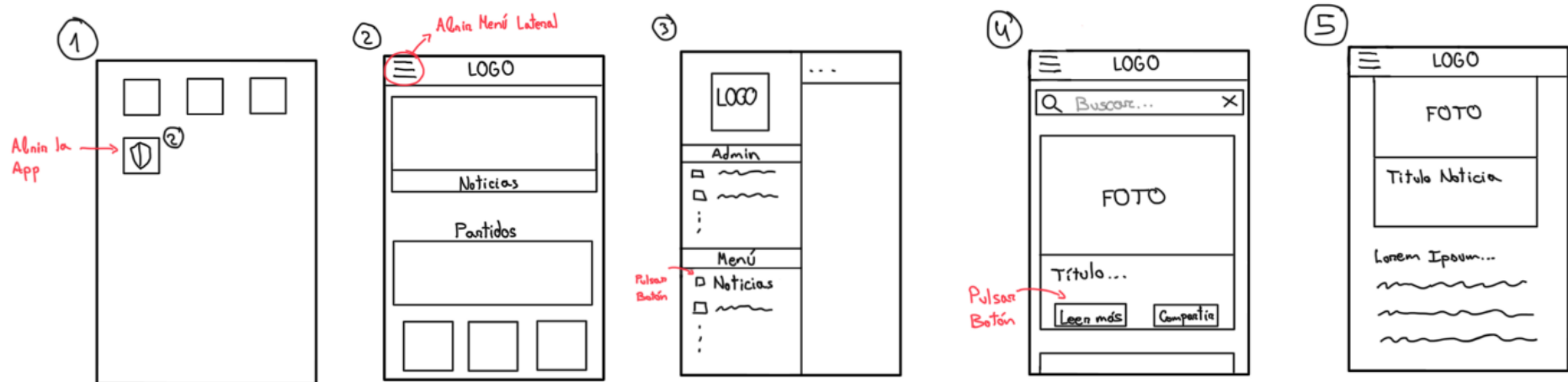
1. Escena 1: Usuario abre la app → Ve a la página de inicio.
2. Escena 2: Usuario abre el menú lateral.
3. Escena 3: Usuario pulsa en “Iniciar Sesión”.
4. Escena 4: Usuario introduce sus credenciales.
5. Escena 5: Usuario pulsa el botón “Acceder”.
6. Escena 6: Sesión iniciada por el usuario.



Storyboard inicio sesión 22

### 3.3.1.2 Storyboard de cómo un usuario revisa una noticia:

1. Escena 1: Usuario abre la app → Ve a la página de inicio.
2. Escena 2: Usuario abre el menú lateral o hace scroll hacia abajo.
3. Escena 3: Usuario pulsa en “Noticias”.
4. Escena 4: Usuario se desplaza por el listado y escoge una noticia que quiera ver, entonces pulsa “Leer más”.
5. Escena 6: Pantalla de los detalles de la noticia.



Storyboard ver noticias 23

### 3.3.1.3 Storyboard de cómo un usuario comparte una noticia:

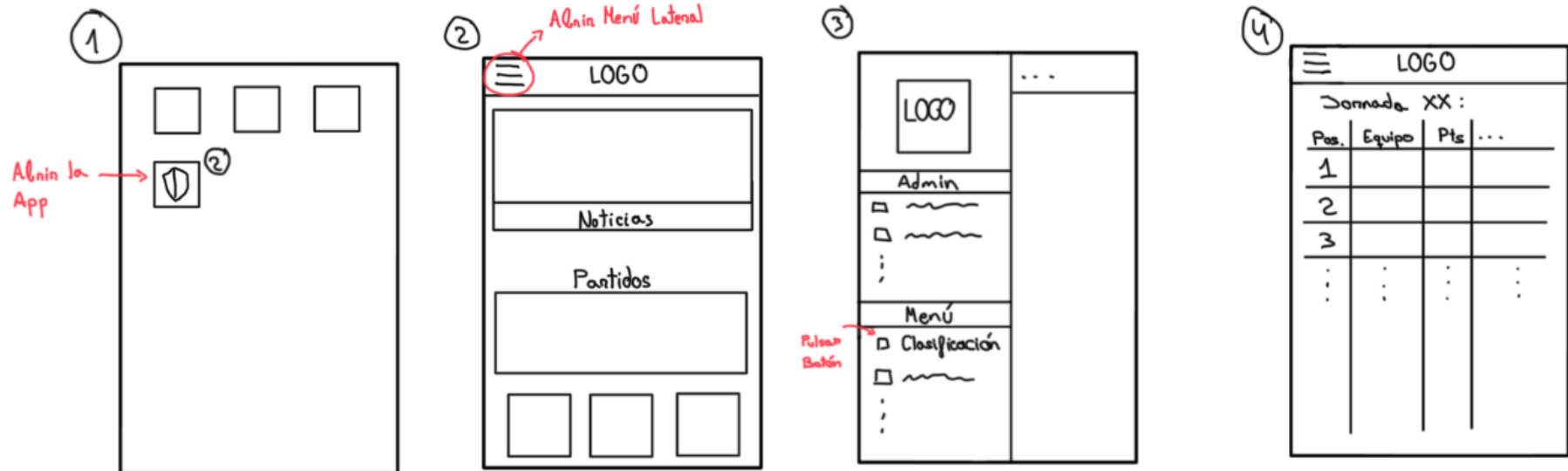
1. Escena 1: Usuario abre la app → Ve a la página de inicio.
2. Escena 2: Usuario abre el menú lateral o hace scroll hacia abajo.
3. Escena 3: Usuario pulsa en “Noticias”.
4. Escena 4: Usuario se desplaza por el listado y escoge una noticia que quiera compartir.
5. Escena 5: Usuario pulsa en “Compartir”.
6. Escena 6: Pantalla con las opciones del teléfono para compartir.



Storyboard compartir noticias 24

### 3.3.1.3 Storyboard de cómo un usuario revisa la clasificación:

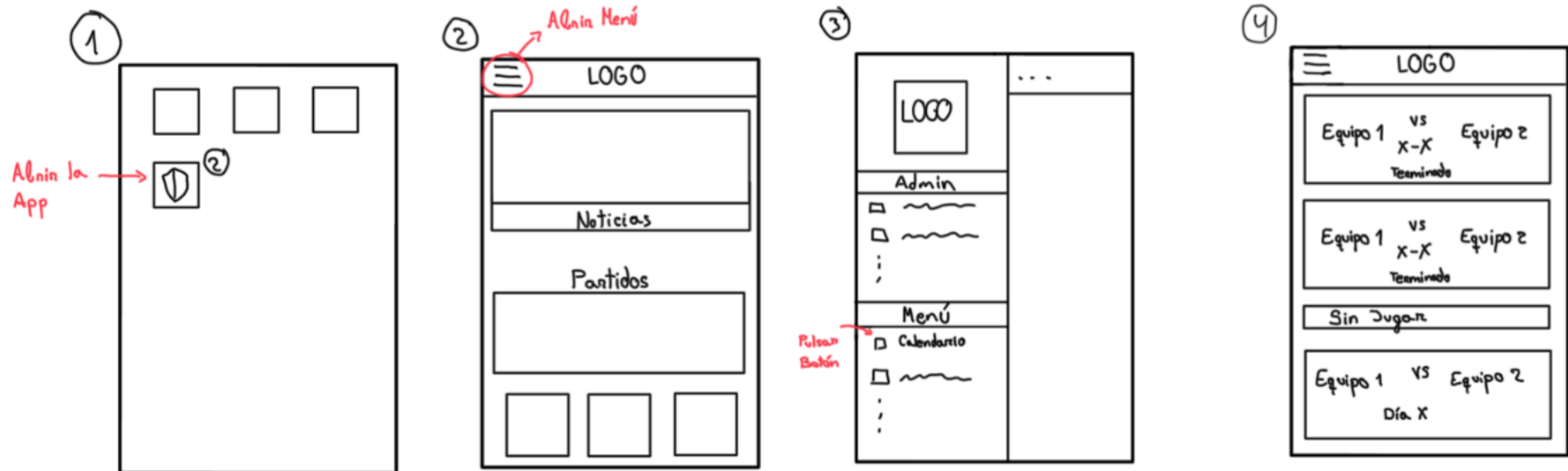
1. Escena 1: Usuario abre la app → Ve a la página de inicio.
2. Escena 2: Usuario abre el menú lateral o hace scroll hacia abajo.
3. Escena 3: Usuario pulsa en “Clasificación”.
4. Escena 4: Pantalla de la clasificación de la liga.



Storyboard clasificación 25

### 3.3.1.4 Storyboard de cómo un usuario revisa el calendario:

1. Escena 1: Usuario abre la app → Ve a la página de inicio.
2. Escena 2: Usuario abre el menú lateral o hace scroll hacia abajo.
3. Escena 3: Usuario pulsa en “Calendario y Resultados”.
4. Escena 4: Pantalla del calendario y resultados de la liga.

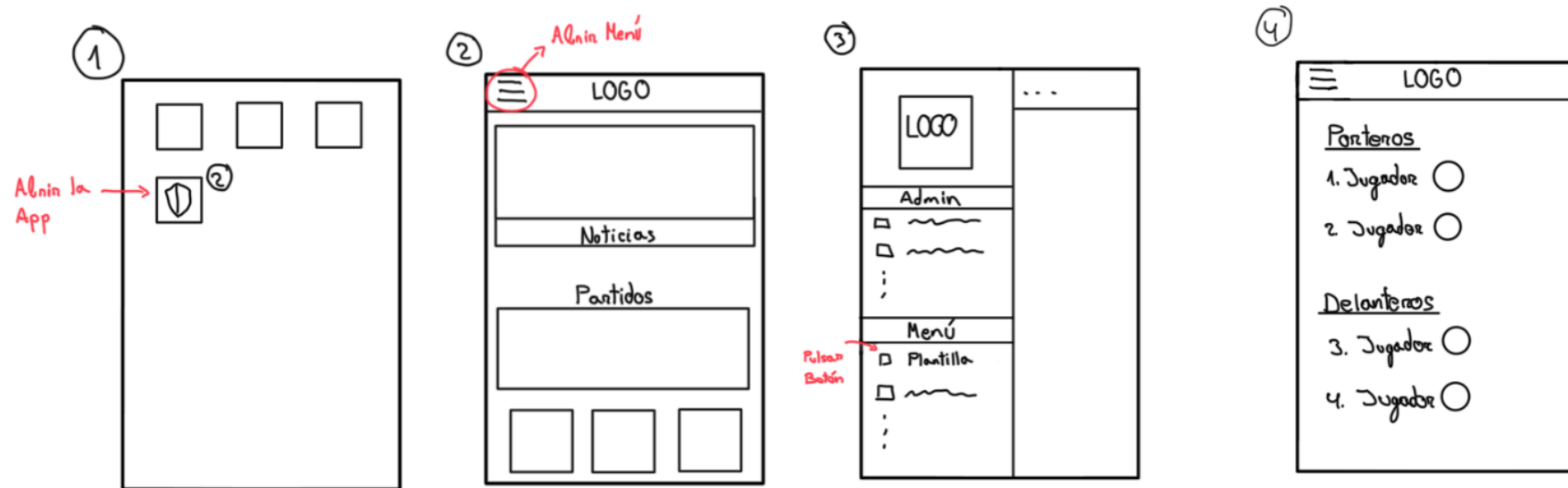


Storyboard calendario 26



### 3.3.1.5 Storyboard de cómo un usuario consulta la plantilla activa del equipo:

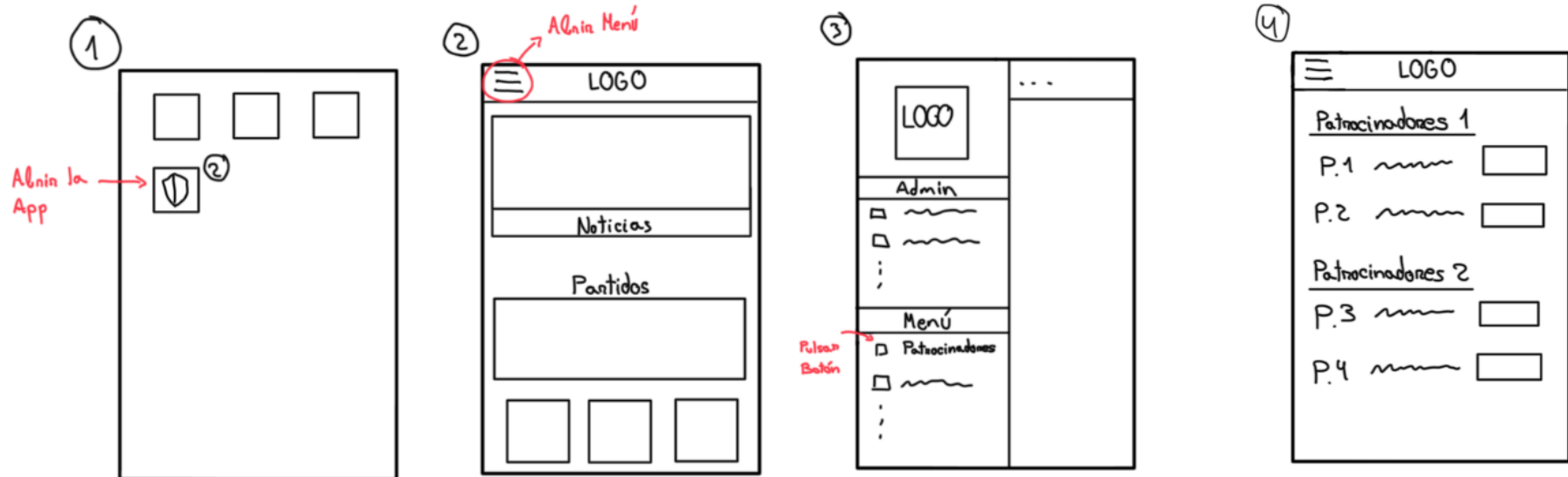
1. Escena 1: Usuario abre la app → Ve a la página de inicio.
2. Escena 2: Usuario abre el menú lateral.
3. Escena 3: Usuario pulsa en "Plantilla".
4. Escena 4: Pantalla de los detalles de la plantilla del equipo.



Storyboard plantilla 27

### 3.3.1.6 Storyboard de cómo un usuario consulta los patrocinadores del equipo:

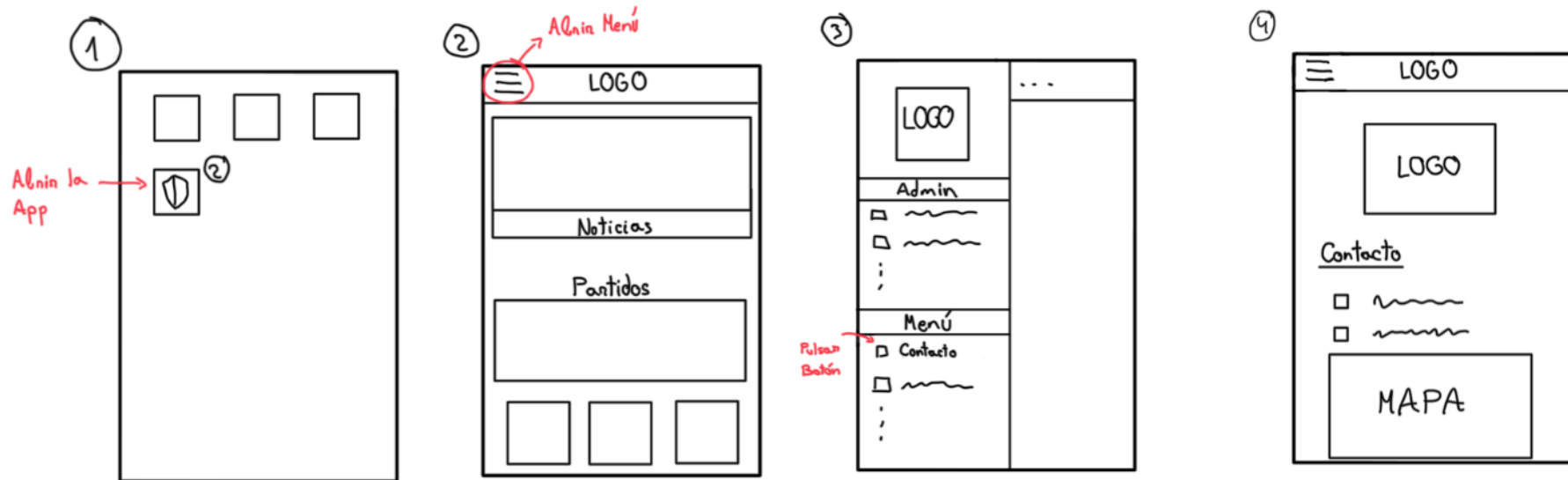
1. Escena 1: Usuario abre la app → Ve a la página de inicio.
2. Escena 2: Usuario abre el menú lateral.
3. Escena 3: Usuario pulsa en “Patrocinadores”.
4. Escena 4: Pantalla de los patrocinadores del equipo.



Storyboard patrocinadores 28

### 3.3.1.7 Storyboard de cómo un usuario consulta la información de contacto:

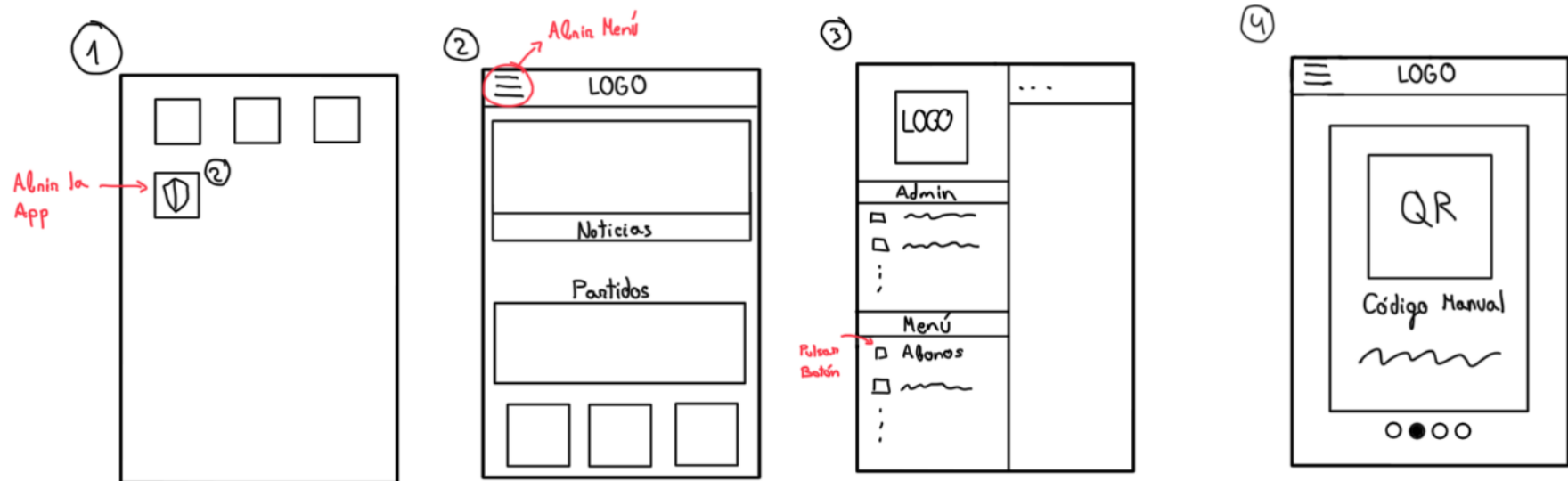
1. Escena 1: Usuario abre la app → Ve a la página de inicio.
2. Escena 2: Usuario abre el menú lateral.
3. Escena 3: Usuario pulsa en "Contacto".
4. Escena 4: Pantalla con información del contacto del equipo.



Storyboard contacto 29

### 3.3.1.8 Storyboard de cómo un usuario accede a su carné de abonado:

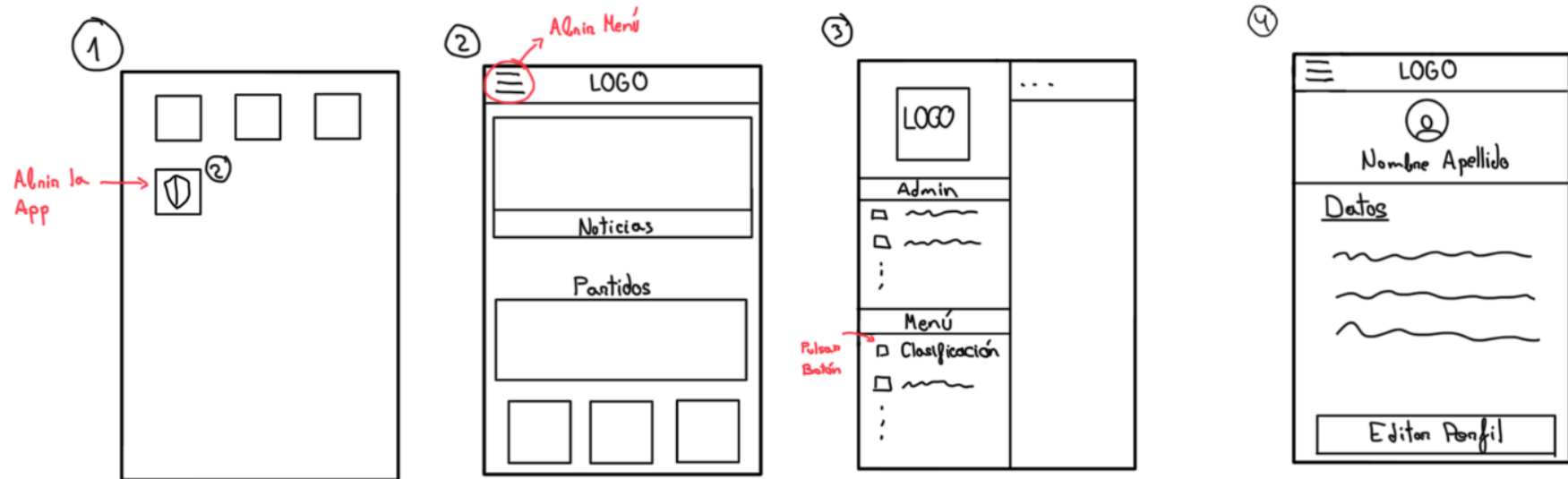
1. Escena 1: Usuario abre la app → Ve a la página de inicio.
2. Escena 2: Usuario inicia sesión.
3. Escena 3: Usuario abre el menú lateral o hace scroll hacia abajo.
4. Escena 4: Usuario pulsa en “Carné Abonado”.
5. Escena 5: Pantalla con los diferentes abonos del usuario.



Storyboard carné abonado 30

### 3.3.1.9 Storyboard de cómo un usuario consulta su perfil de usuario:

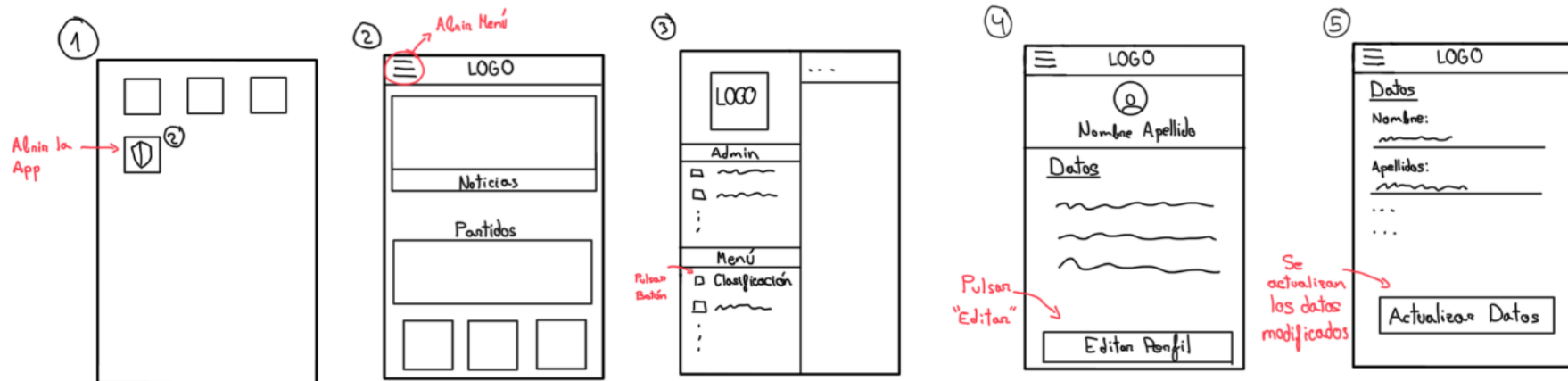
1. Escena 1: Usuario abre la app → Ve a la página de inicio.
2. Escena 2: Usuario inicia sesión.
3. Escena 3: Usuario abre el menú lateral.
4. Escena 4: Usuario pulsa en “Mi Perfil”.
5. Escena 5: Pantalla con los datos del perfil del usuario.



Storyboard perfil 31

### 3.3.1.10 Storyboard de cómo un usuario edita sus datos de perfil de usuario:

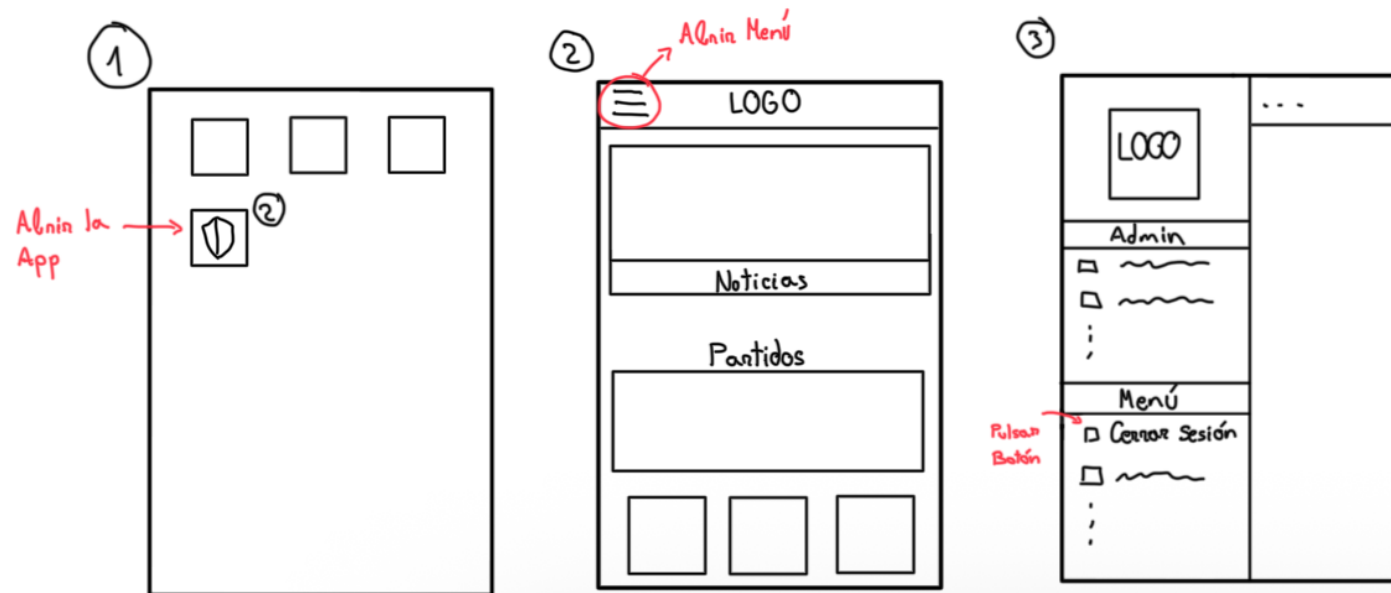
1. Escena 1: Usuario abre la app → Ve a la página de inicio.
2. Escena 2: Usuario inicia sesión.
3. Escena 3: Usuario abre el menú lateral.
4. Escena 4: Usuario pulsa en “Mi Perfil”.
5. Escena 5: Usuario pulsa “Editar Perfil”.
6. Escena 6: Usuario modifica los datos que quiera de su perfil.
7. Escena 7: Usuario pulsa “Actualizar Perfil”.
8. Escena 8: Pantalla con los nuevos datos del perfil del usuario.



Storyboard editar perfil 32

### 3.3.1.11 Storyboard de cómo un usuario cierra sesión:

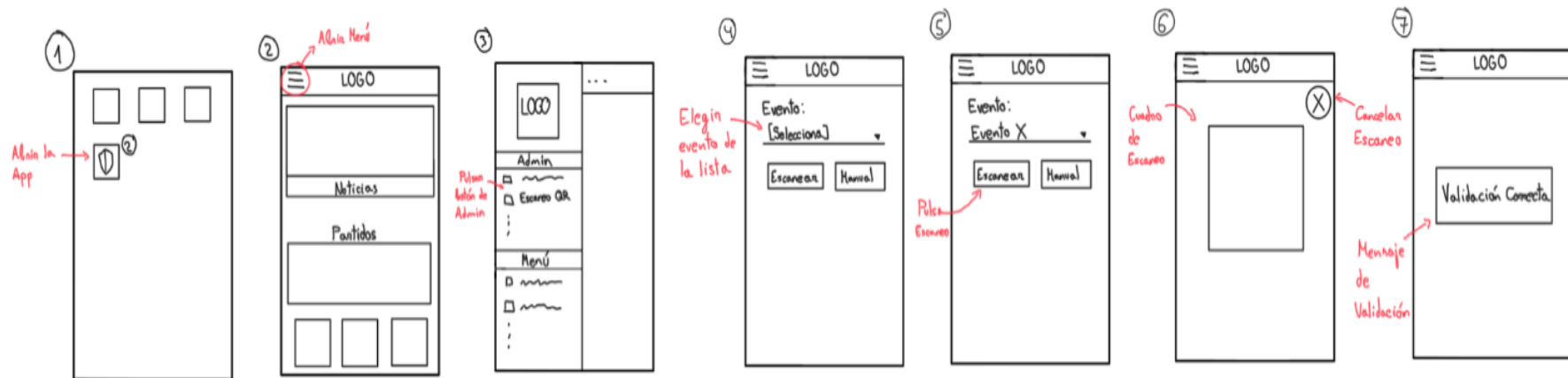
1. Escena 1: Usuario abre la app → Ve a la página de inicio
2. Escena 2: Usuario inicia sesión.
3. Escena 3: Usuario abre el menú lateral.
4. Escena 4: Usuario pulsa en “Cerrar Sesión”.
5. Escena 5: Pantalla de inicio y aviso de sesión cerrada.



Storyboard cerrar sesion 33

### 3.3.1.12 Storyboard de cómo un administrador escanea las entradas:

1. Escena 1: Usuario abre la app → Ve a la página de inicio.
2. Escena 2: Usuario inicia sesión como administrador.
3. Escena 3: Usuario abre el menú lateral.
4. Escena 4: Usuario pulsa en “Escanear Entradas”.
5. Escena 5: Usuario escoge un evento de la lista de eventos activos.
6. Escena 6: Usuario pulsa “Escanear”.
7. Escena 7: Usuario enfoca con la cámara la entrada o abono que quiera escanear.
8. Escena 8: Salta un aviso de validación o error en la pantalla de escaneo.

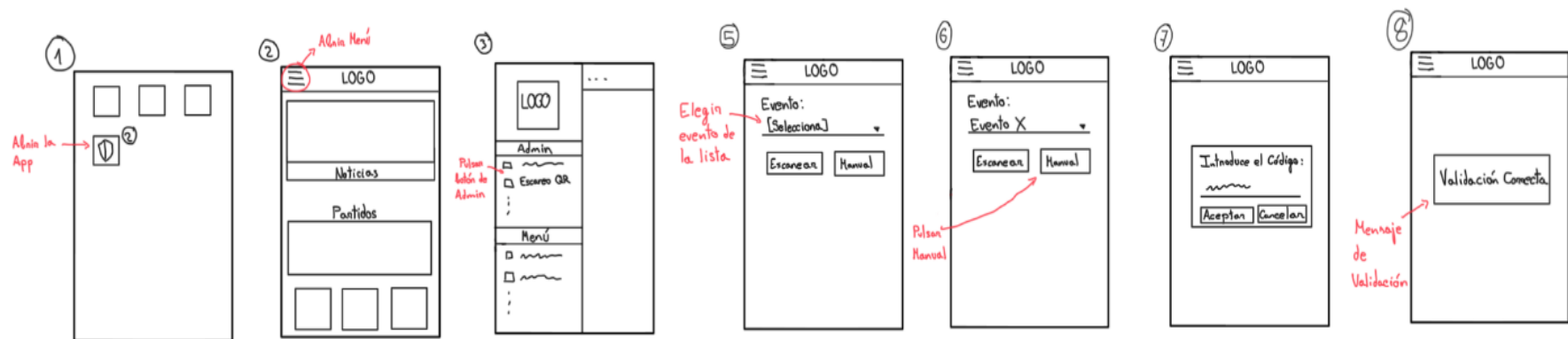


Storyboard escaneo QR 34



### 3.3.1.13 Storyboard de cómo un administrador introduce el código de una entrada:

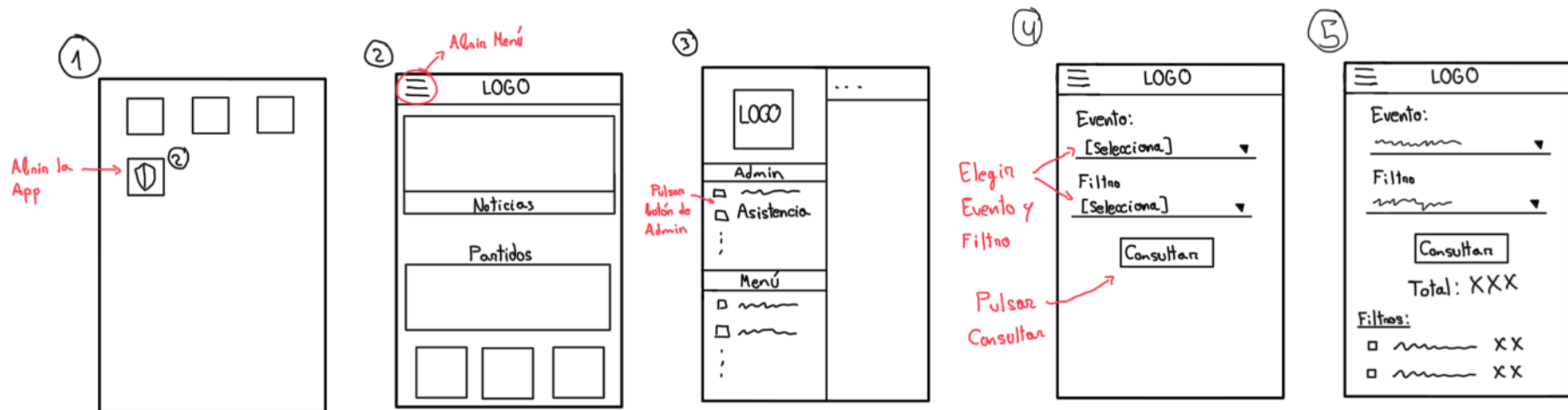
1. Escena 1: Usuario abre la app → Ve a la página de inicio.
2. Escena 2: Usuario inicia sesión como administrador.
3. Escena 3: Usuario abre el menú lateral.
4. Escena 4: Usuario pulsa en “Escanear Entradas”.
5. Escena 5: Usuario escoge un evento de la lista de eventos activos.
6. Escena 6: Usuario pulsa “Manual”.
7. Escena 7: Usuario introduce manualmente el código de la entrada.
8. Escena 8: Salta un aviso de validación o error en la pantalla.



Storyboard QR manual 35

### 3.3.1.14 Storyboard de cómo un administrador consulta la asistencia a un evento.

1. Escena 1: Usuario abre la app → Ve a la pantalla de inicio.
2. Escena 2: Usuario inicia sesión como administrador.
3. Escena 3: Usuario abre el menú lateral.
4. Escena 4: Usuario pulsa en “Asistencia”.
5. Escena 5: Usuario escoge un evento de la lista de eventos.
6. Escena 6: Usuario escoge si quiere aplicar un filtro.
7. Escena 7: Usuario pulsa “Consultar”.
8. Escena 8: Pantalla con la lista de la asistencia al evento que se ha consultado.



Storyboard asistencia 36

### 3.3.1.15 Storyboard de cómo un administrador consulta la taquilla de un evento.

1. Escena 1: Usuario abre la app → Ve a la pantalla de inicio.
2. Escena 2: Usuario inicia sesión como administrador.
3. Escena 3: Usuario abre el menú lateral.
4. Escena 4: Usuario pulsa en “Taquilla”.
5. Escena 5: Usuario escoge un evento de la lista de eventos.
6. Escena 6: Usuario pulsa “Consultar”.
7. Escena 7: Pantalla con la lista de la recaudación del evento que se ha consultado.



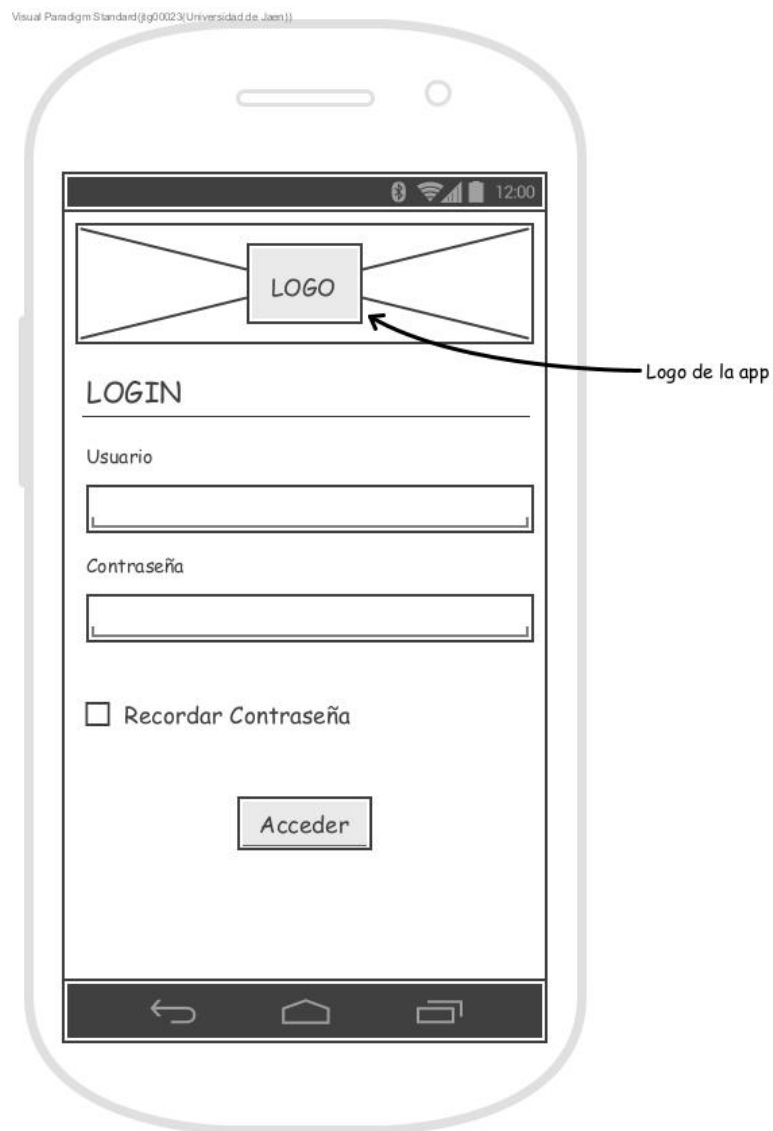
Storyboard taquilla 37

Una vez realizados los storyboards y validados con la entidad, nos disponemos a presentar los wireframes creados para cumplir con los requerimientos recogidos y los casos de uso de la aplicación móvil

### 3.3.2 Wireframes

- Login:

En el siguiente wireframe podemos ver la pantalla de inicio de sesión o “Login”. Tiene una interfaz sencilla, se muestra el logo de la App en la parte superior, y justo debajo el formulario con la opción de recordar la contraseña para futuras ocasiones.



Wireframe de “Login” 38

- Registro:

En el siguiente wireframe, podemos ver la pantalla de “Registro” de nuevos usuarios. Similar a la pantalla de inicio de sesión, se compone del logo en la parte superior, y un sencillo formulario con los datos más detallados, además del botón obligatorio de “aceptar la política de privacidad”.

Visual Paradigm Standard (jg00023@Universidad de Jaén)

Introduce tus datos

Nombre y Apellidos

Email

DNI

Contraseña

Confirma Contraseña

☐ Aceptar política de privacidad

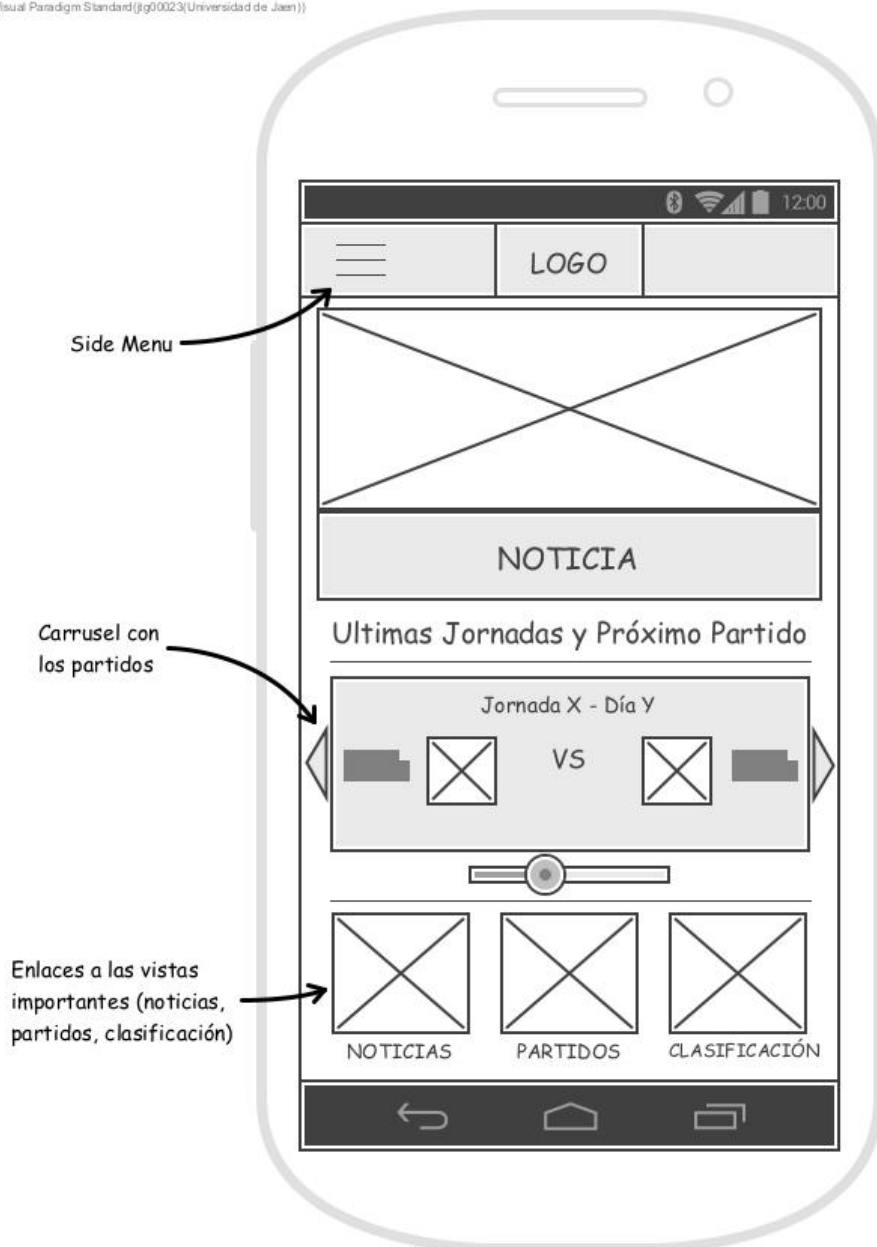
Acceder

Wireframe de “Registro” 39

- Home:

En el siguiente wireframe podemos ver la pantalla “Home”, esta vista será lo primero que se vea al abrir la aplicación por primera vez. Contiene una barra superior que permanecerá estática en el resto de pantallas junto con el menú lateral o “Side Menu”. Contiene un carrusel con las últimas noticias y otro con los partidos. Además, también tiene iconos en grande de las vistas más visitadas para poder desplazarse con más facilidad.

Visual Paradigm Standard (jg00023(Universidad de Jaén))

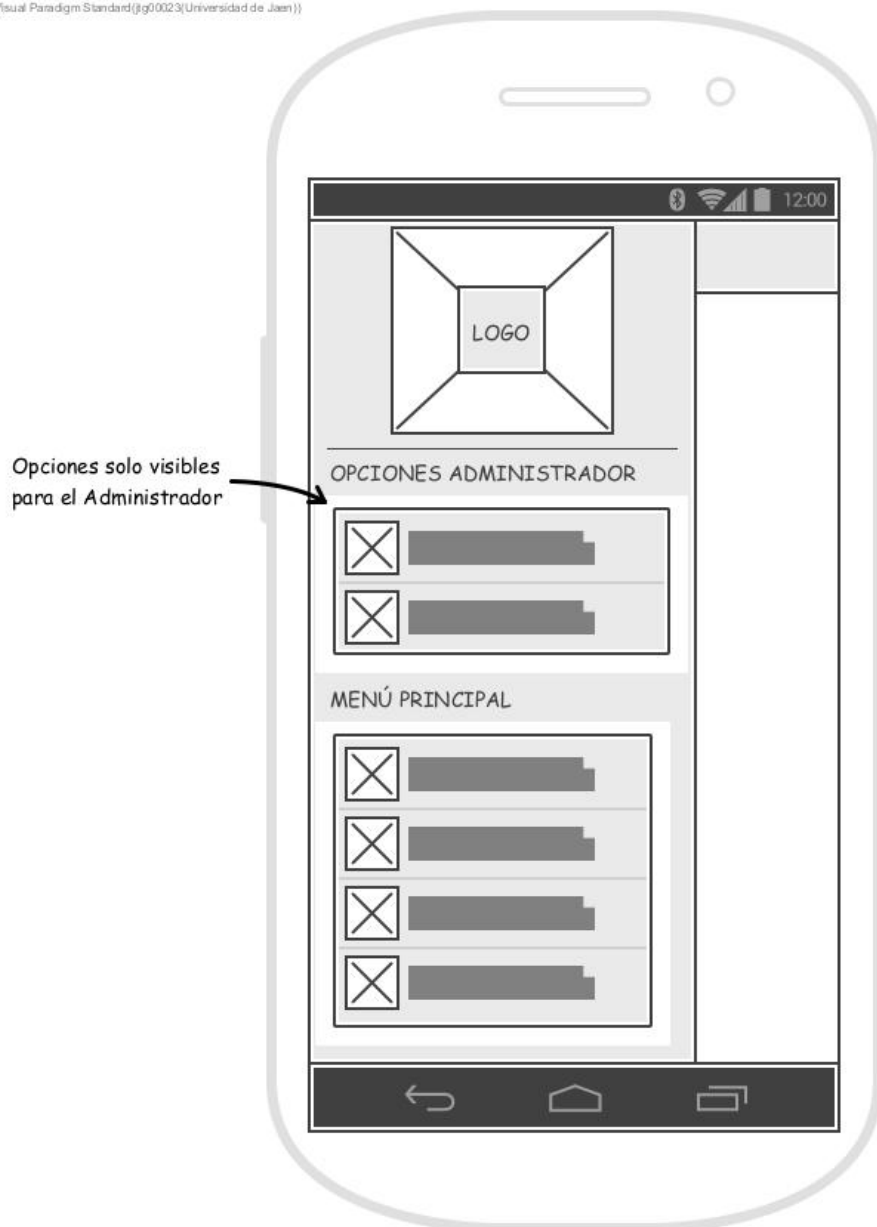


Wireframe de “Home” 40

- Side Menu:

En el siguiente wireframe podemos ver la pantalla de cómo se vería el “Side Menu” desplegado. Este menú se podría desplegar en prácticamente cualquiera de las vistas, y contiene las diferentes opciones de navegación de la aplicación, dependiendo de si el rol del usuario con el que se inicia sesión es de administrador o solo usuario.

Visual Paradigm Standard (jg00023(Universidad de Jaén))

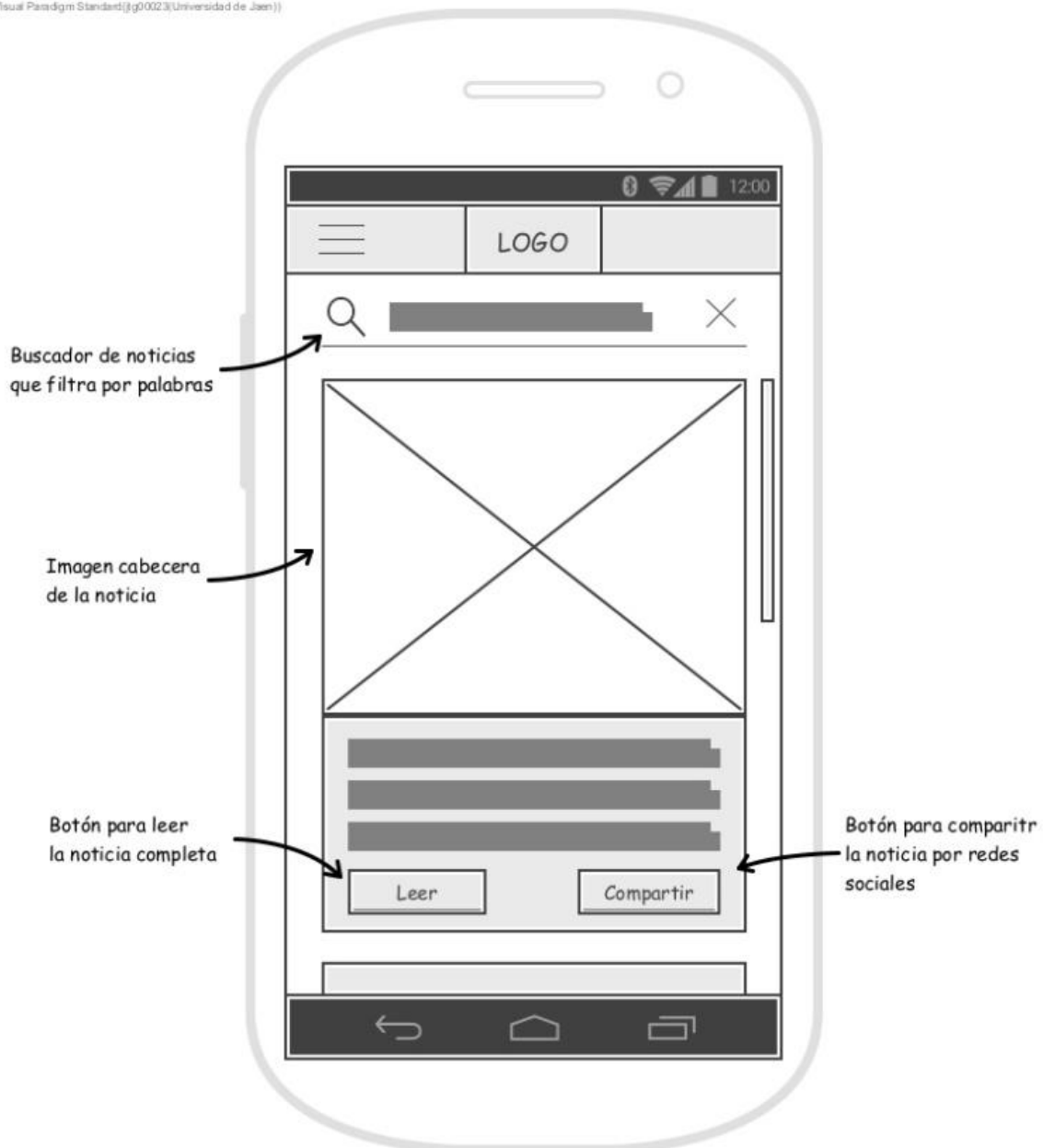


Wireframe de “Side Menu” 41

- Noticias:

En el siguiente wireframe podemos ver la pantalla “Noticias”. En la parte superior contiene una barra de búsqueda que permite filtrar las noticias. Cada noticia aparece con su imagen de cabecera, su título y 2 botones de acción, uno para compartir la noticia por redes sociales, y otro para leer los detalles de la noticia. Las diferentes noticias se van generando en un scroll infinito, lo que permite la carga dinámica de las siguientes noticias.

Visual Paradigm Standard(jg00023/Universidad de Jaén)



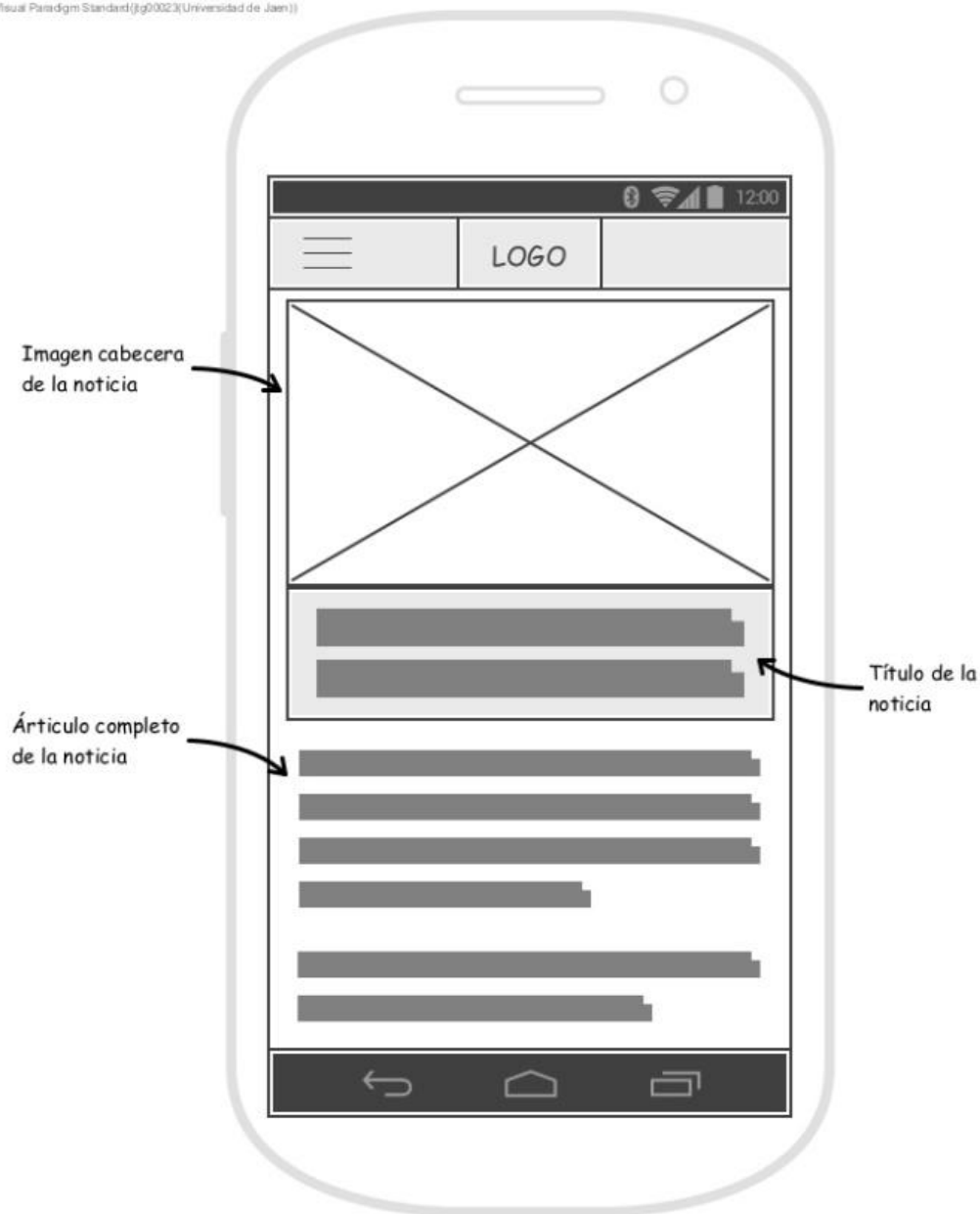
Wireframe de “Noticias” 42



- Detalle Post:

En el siguiente wireframe podemos ver la pantalla de “Detalle Post”. En la parte superior podemos observar la imagen completa de la cabecera de la noticia junto con el título, y justo debajo el artículo completo de la noticia.

Visual Paradigm Standard (jg00023(Universidad de Jaén))

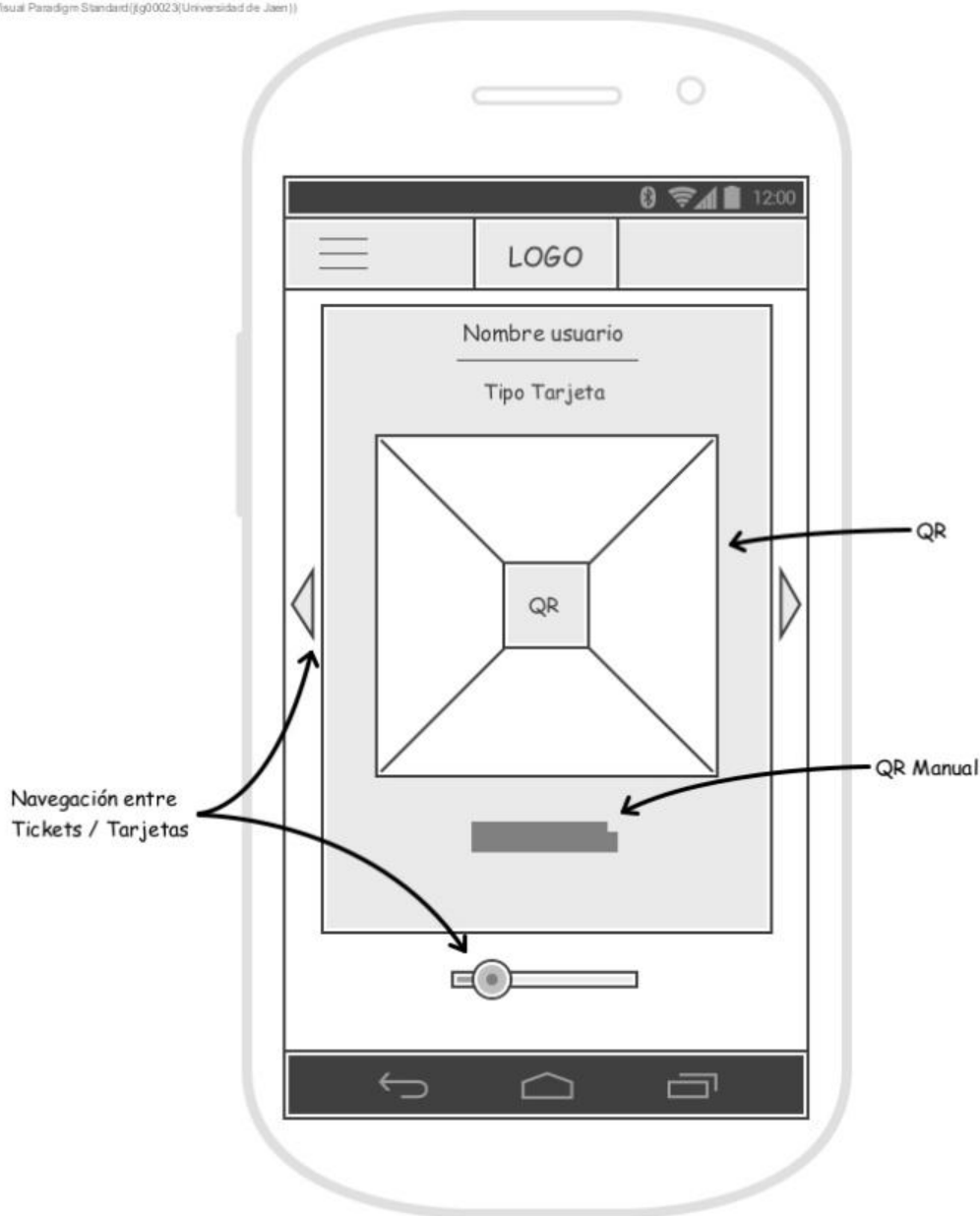


Wireframe de “Detalle Post” 43

- **Carné Abonado:**

En el siguiente wireframe podemos ver la pantalla de “Carné Abonado”. Se muestran las tarjetas/abonos de los que dispone el usuario. Dentro de cada tarjeta podemos observar el tipo de tarjeta y el QR que permite escanear dicha entrada. La navegación entre tarjetas se hace deslizando hacia derecha e izquierda.

Visual Paradigm Standard (jg00023/Universidad de Jaén)

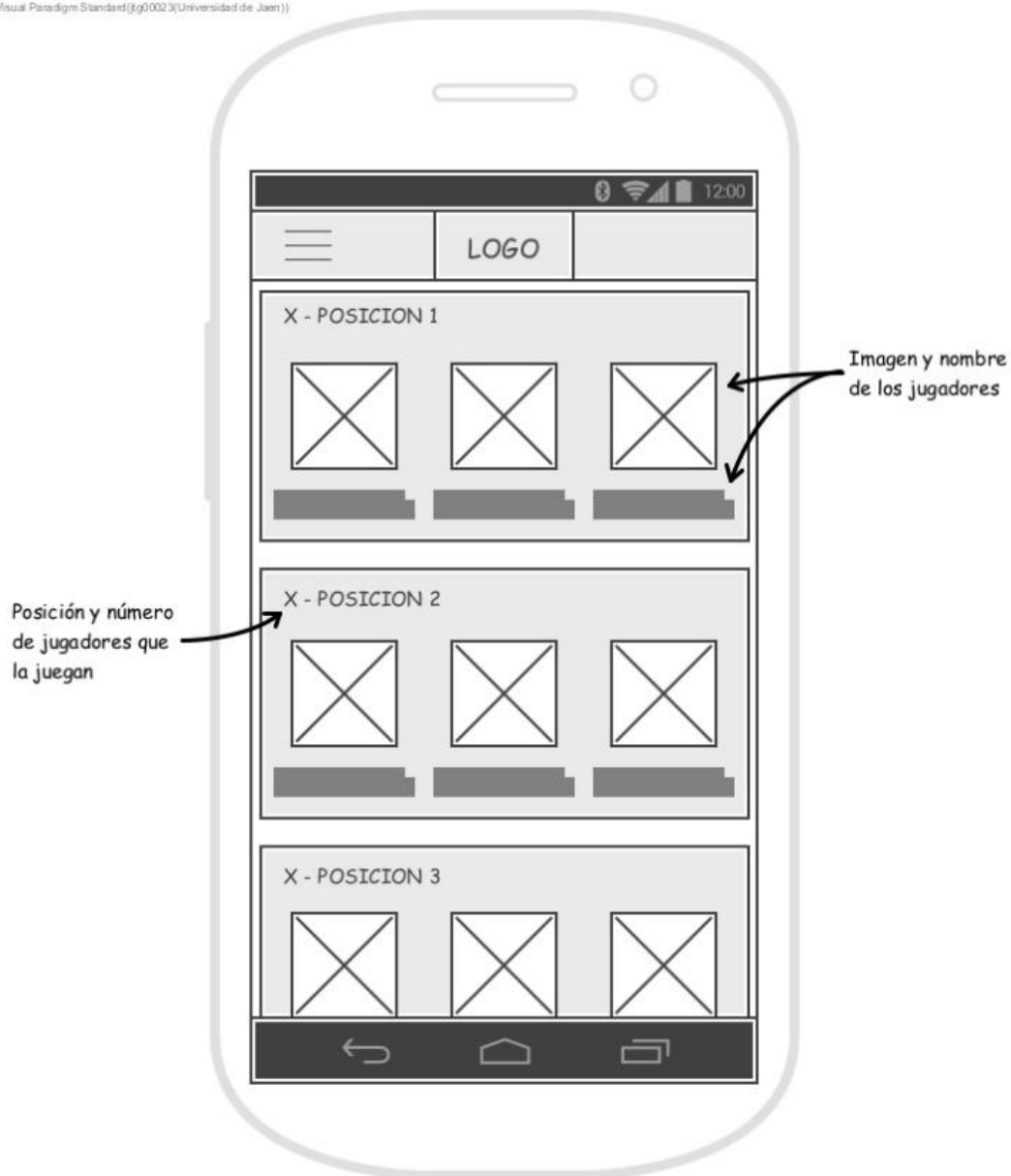


Wireframe de “Carné Abonado” 44

- Plantilla:

En el siguiente wireframe podemos ver la pantalla de “Plantilla”. Nos muestra la información de la actual plantilla del equipo (nombre, dorsal...) y lo ordena en función de la posición en la que juegan.

Visual Paradigm Standard (jg00023@Universidad de Jaén))

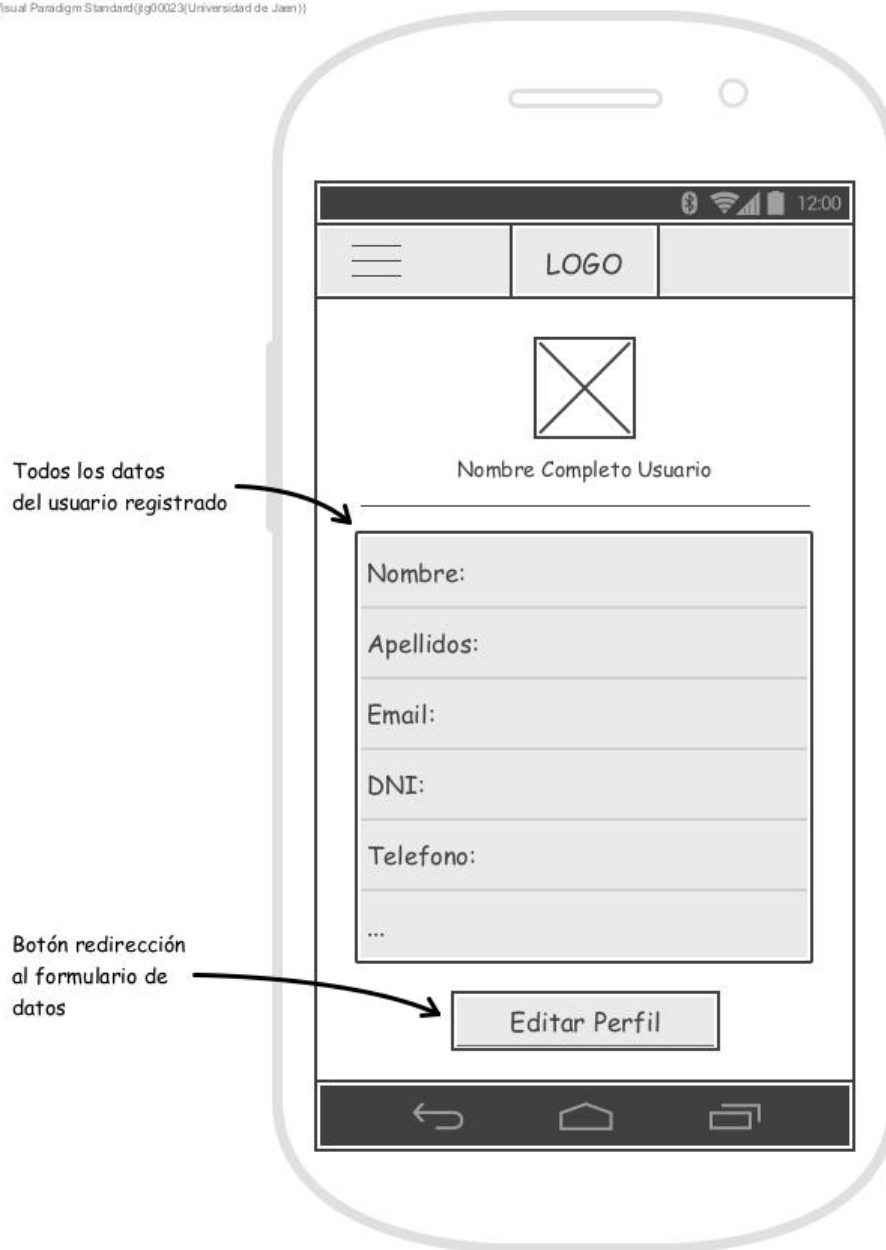


Wireframe de “Plantilla” 45

- Mi Perfil:

En el siguiente wireframe podemos ver la pantalla de “Mi Perfil”. Se muestra la información del usuario registrado, junto con una foto adjunta que haya subido y un botón que abre el formulario para editar dicho perfil. Es necesario haber iniciado sesión previamente para poder acceder a esta vista, si no se ha iniciado sesión antes, se redirecciona a la vista de “Login”.

Visual Paradigm Standard (jg0023@Universidad de Jaén)



Wireframe de “Mi Perfil” 46

- Actualizar Perfil:

En el siguiente wireframe podemos ver la pantalla de “Actualizar Perfil”. Se muestra el formulario con los datos completos del usuario y permite modificar los que se desee. Una vez modificados los datos el botón de confirmar los guarda.

Visual Paradigm Standard (jg00023(Universidad de Jaén))

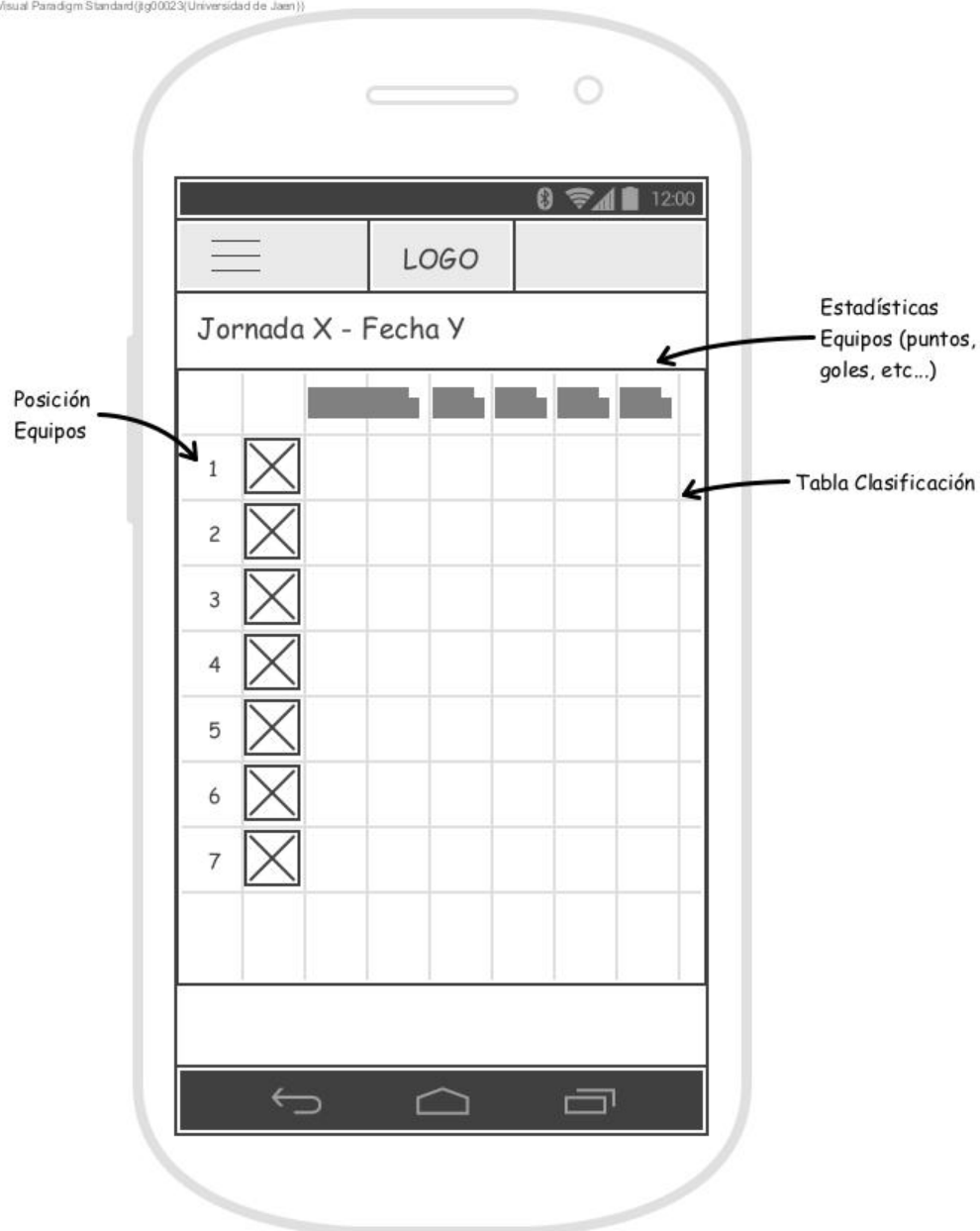
The wireframe illustrates a mobile application screen for updating a user's profile. The screen is enclosed in a rounded rectangle representing a smartphone. At the top, there is a status bar with icons for signal, Wi-Fi, battery, and the time 12:00. Below the status bar is a header bar with a hamburger menu icon on the left, a 'LOGO' placeholder in the center, and a blank space on the right. The main content area features a large square placeholder with an 'X' inside, representing a profile picture. Below this is the label 'Nombre Completo Usuario'. The form consists of several input fields: 'Nombre', 'Apellidos', 'Email', and 'Contraseña'. There is an ellipsis (...) between the 'Email' and 'Contraseña' fields. Below the 'Contraseña' field is a 'Confirmar Contraseña' field. At the bottom of the form is a 'Confirmar' button. Two annotations with arrows point to the form: one points to the 'Nombre' field with the text 'Formulario que permite cambiar todos los datos (o alguno solo)', and the other points to the 'Confirmar' button with the text 'Botón que confirma los datos nuevos'. The bottom of the screen shows a standard Android navigation bar with back, home, and recent apps icons.

Wireframe de “Actualizar Perfil” 47

- Clasificación:

En el siguiente wireframe podemos ver la pantalla “Clasificación”. Nos muestra una tabla con estadísticas detalladas de la actual clasificación de la temporada activa, así como la posición en la que se encuentran los equipos en el momento actual de la temporada.

Visual Paradigm Standard (jg00023(Universidad de Jaén))

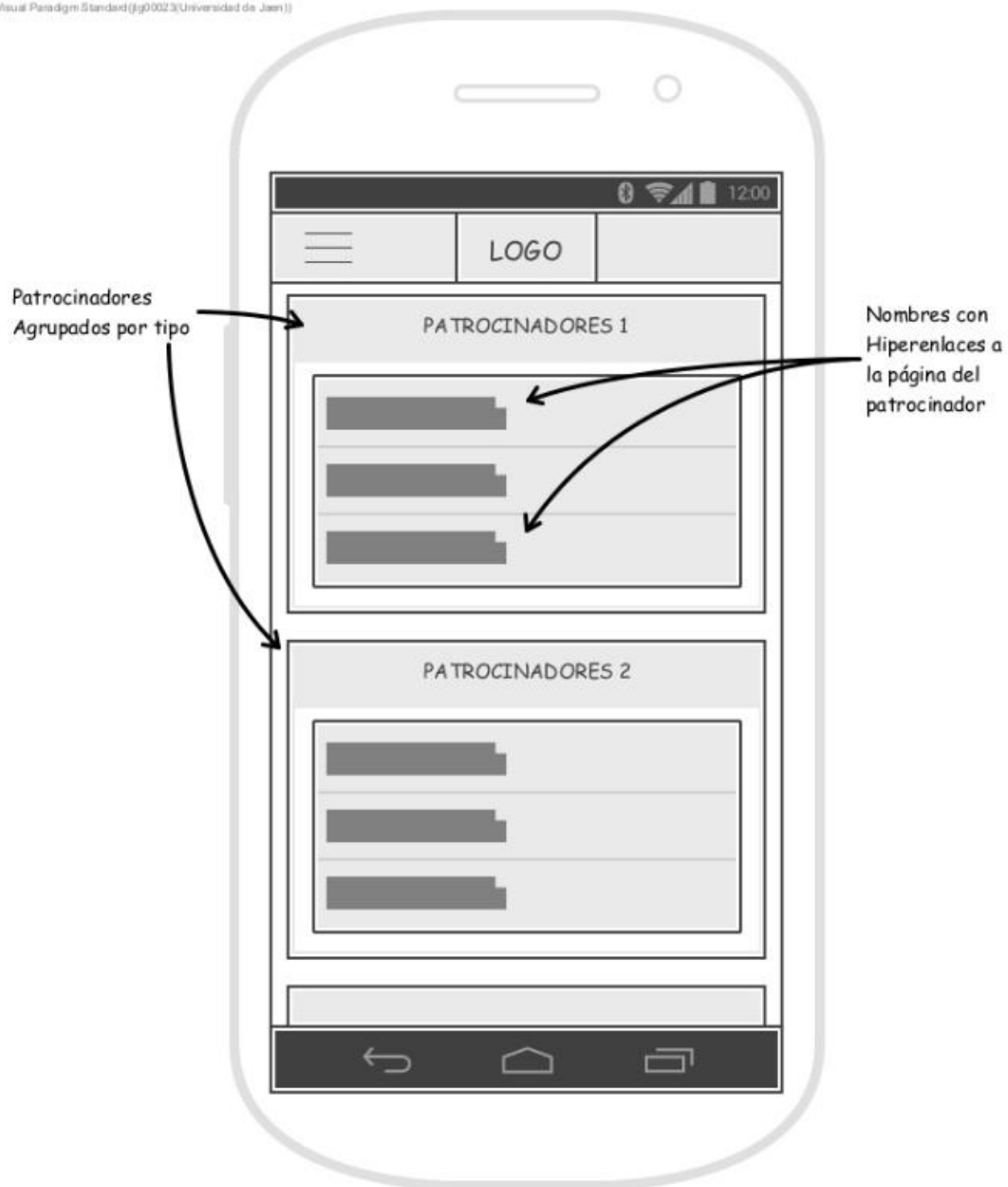


Wireframe de “Clasificación” 48

- Patrocinadores:

En el siguiente wireframe podemos ver la pantalla “Patrocinadores”. Nos muestra una vista sencilla de los diferentes patrocinadores que tiene el equipo agrupados por las diferentes categorías.

Visual Paradigm Standard (jg00023/Universidad de Jaén))

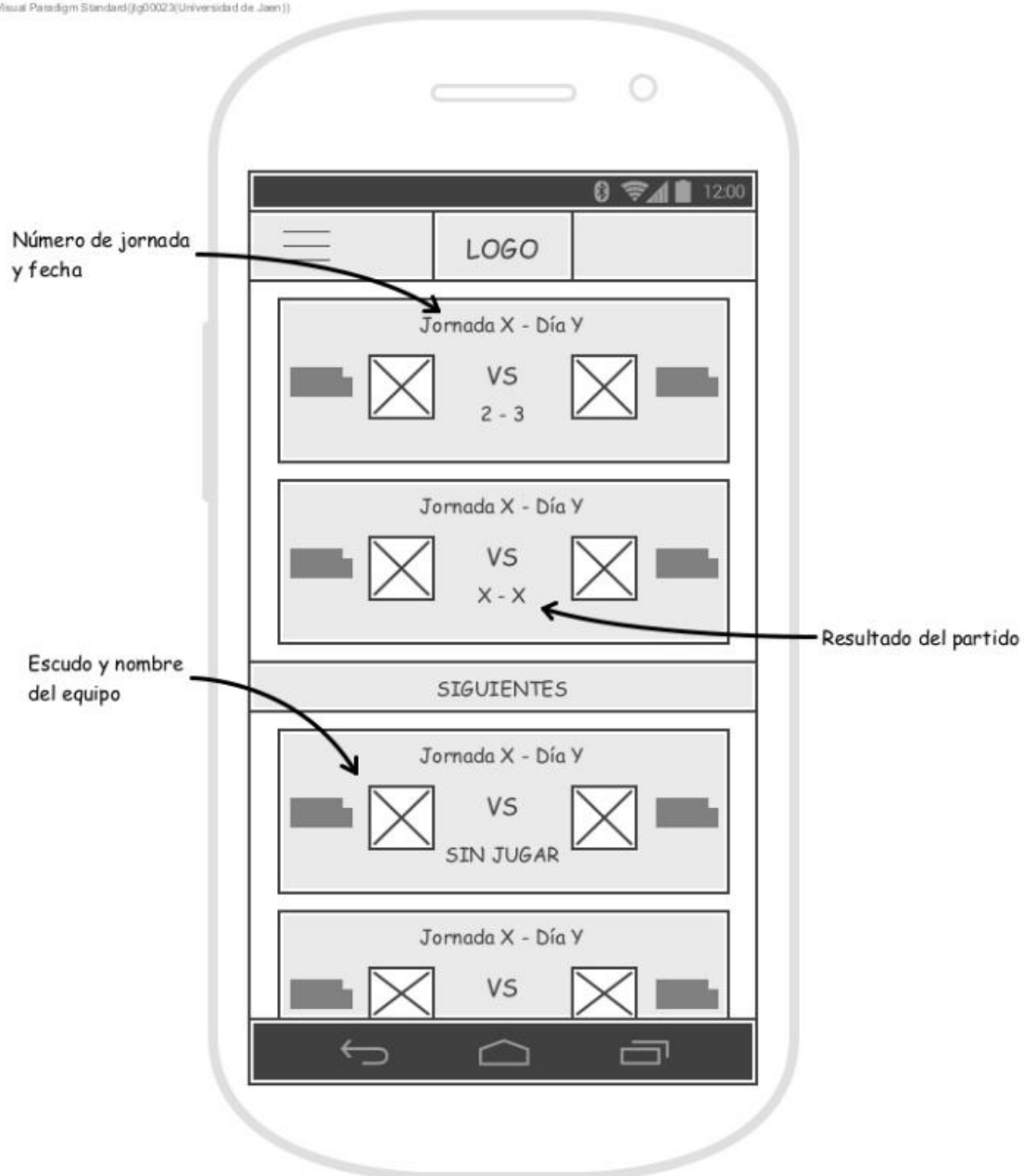


Wireframe de “Patrocinadores” 49

- Calendario y Resultados:

En el siguiente wireframe podemos ver la pantalla “Calendario y Resultados”. Nos muestra una vista detallada del calendario de partidos del equipo en esta temporada activa, haciendo una diferenciación clara entre los partidos ya jugados y los que están por venir.

Visual Paradigm Standard(jg00023(Universidad de Jaén))



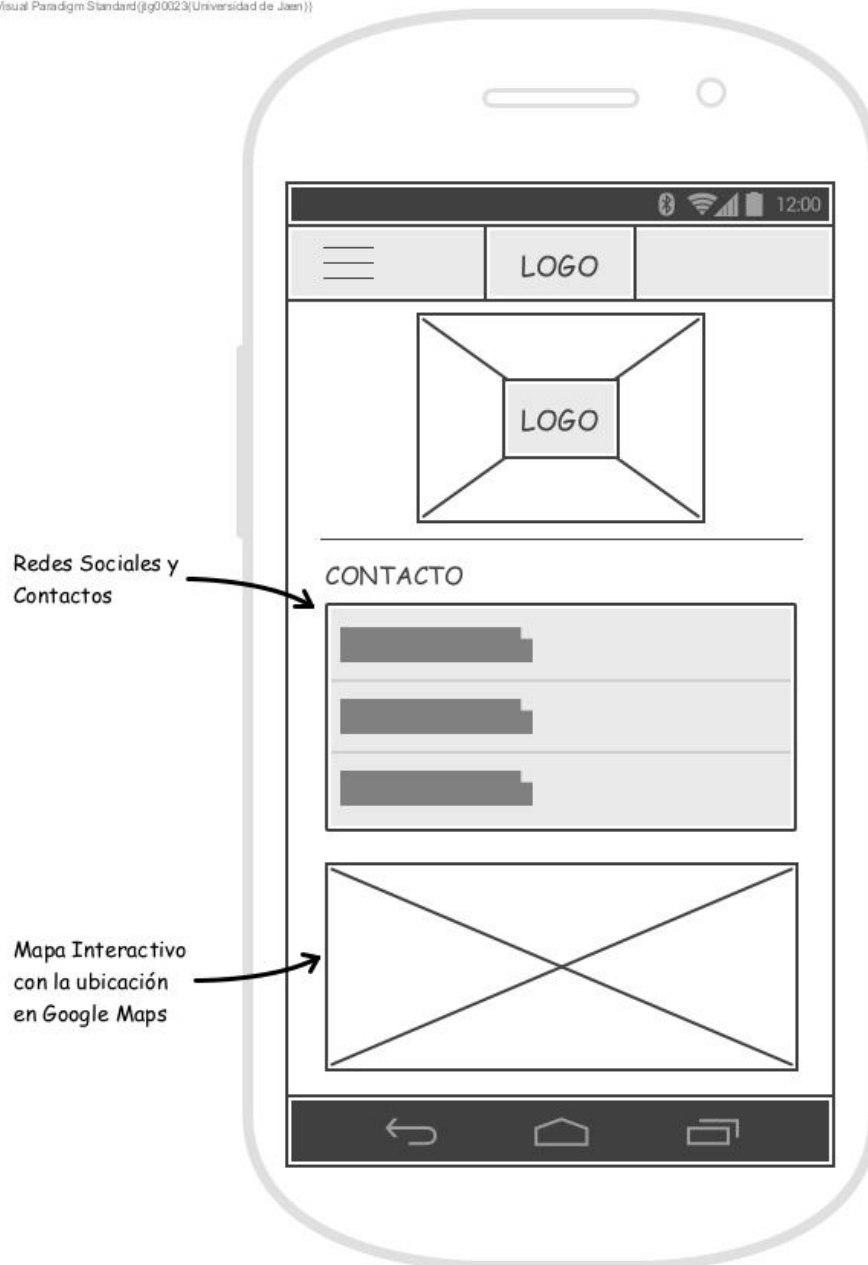
Wireframe de “Calendario y Resultados” 50



- Contacto:

En el siguiente wireframe podemos ver la pantalla “Contacto”. Nos muestra los datos básicos de contacto del equipo (por ejemplo, redes sociales), además de mostrarnos la ubicación de la sede oficial del club en un mapa interactivo de Google Maps.

Visual Paradigm Standard (jg00023(Universidad de Jaén))

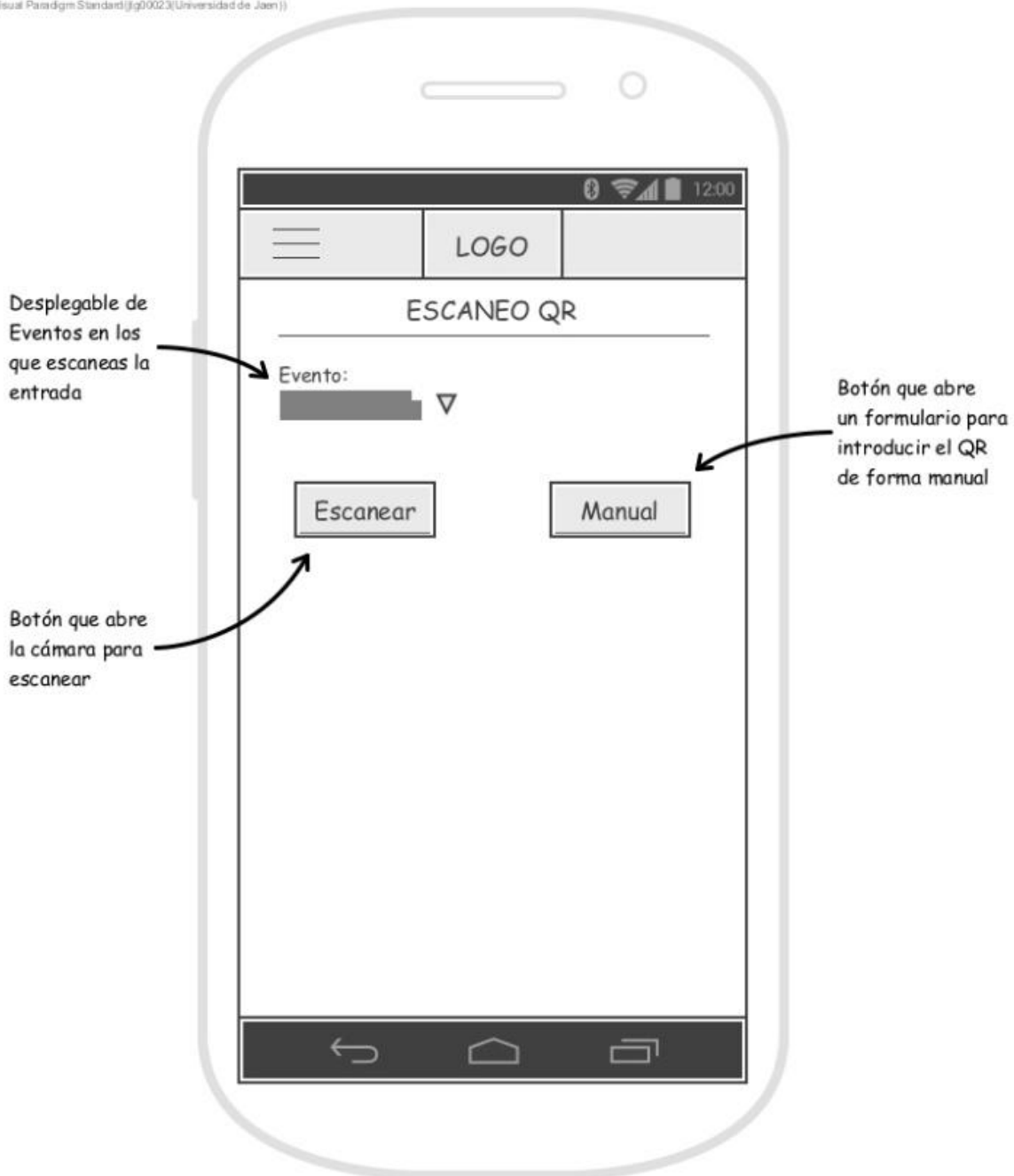


Wireframe de “Contacto” 51

- Escaneo QR:

En el siguiente wireframe podemos ver la pantalla “Escaneo QR”. Una vista solo permitida para el rol *Administrador*, que permitirá en los días de partido poder escanear las entradas (ya sean físicas, o mostradas en el propio móvil con la aplicación) y acceder al estadio. El escaneo puede ser mediante la cámara, pero también tendrá una opción de introducir el código manual.

Visual Paradigm Standard (jg00023(Universidad de Jaén))

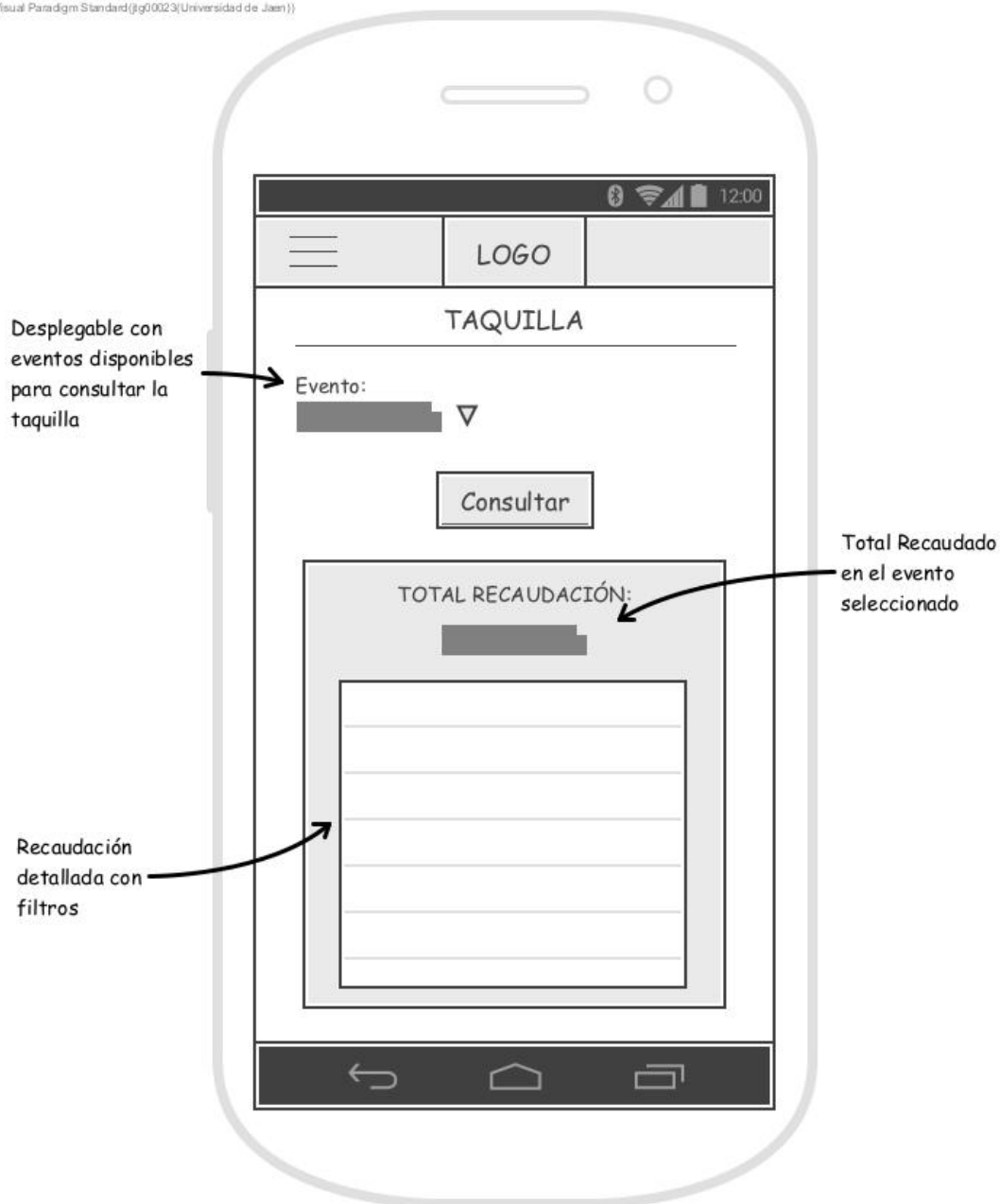


Wireframe de “Escaneo QR” 52

- Taquilla:

En el siguiente wireframe podemos ver la pantalla “Taquilla”. Una vista solo permitida para el rol *Administrador*, que permitirá conocer el dinero recaudado el día de el evento que seleccionemos. Mostrará también un filtro con la especificación sobre de donde ha salido ese dinero (entradas, abonos, etc...).

Visual Paradigm Standard (jg00023(Universidad de Jaén))

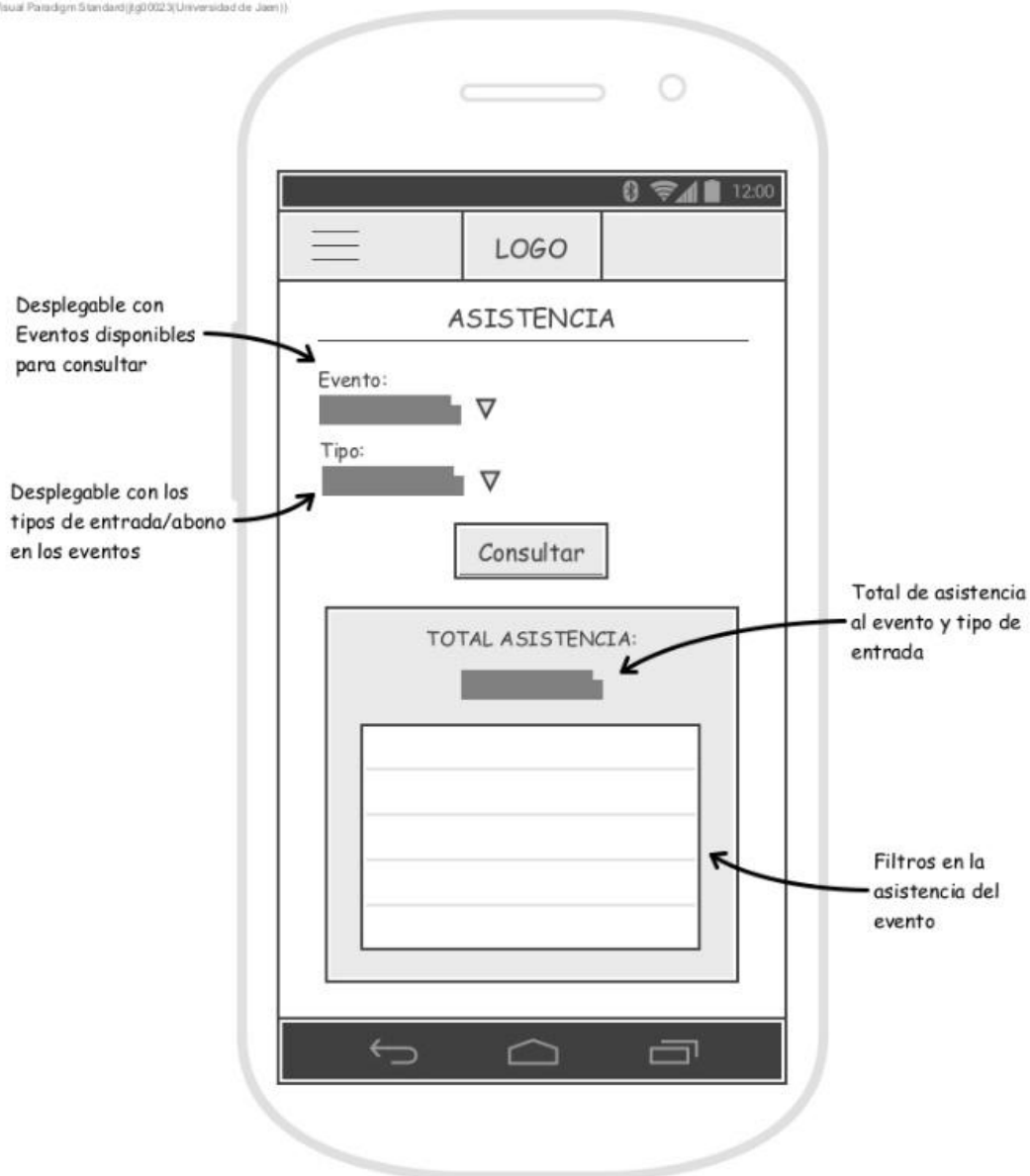


Wireframe de “Taquilla” 53

- Asistencia:

En el siguiente wireframe podemos ver la pantalla “Asistencia”. Una vista solo permitida para el rol *Administrador*, que nos permitirá conocer el número de personas que han asistido a un partido concreto, mostrándonos además detalles sobre el tipo de asistencia que ha habido (número de abonados, número de personas sin abono, juveniles, etc...).

Visual Paradigm Standard (jgo0023@Universidad de Jaén)



Wireframe de “Asistencia” 54

### 3.3.3 Guía de estilo

A continuación, presentaremos la guía de estilo, que ha sido realizada en una pizarra de la herramienta “Figma”:

## Paleta de Colores

Color Principal



Hex: #C70101

RGB: 199, 1, 1

Color Secundario



Hex: #F4F4F4

RGB: 244, 244, 244

Color de Apoyo



Hex: #222222

RGB: 34, 34, 34

Texto Principal



Hex: #F4F4F4

RGB: 244, 244, 244

Texto Secundario



Hex: #C70101

RGB: 199, 1, 1

Texto de Cuerpo



Hex: #6D7A83

RGB: 109, 122, 131

Texto de Detalle



Hex: #209C42

RGB: 32, 156, 66

Color de Error



Hex: #E60B0B

RGB: 230, 11, 11

Color de Éxito



Hex: #32DB64

RGB: 50, 219, 100

Pizarra Figma “paleta de colores” 55

# Tipografía

## Fuente

# EVOGRIA

|             |             |             |             |             |             |             |             |             |             |             |
|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| A<br>U-0041 | B<br>U-0042 | C<br>U-0043 | D<br>U-0044 | E<br>U-0045 | F<br>U-0046 | G<br>U-0047 | H<br>U-0048 | I<br>U-0049 | J<br>U-004A | K<br>U-004B |
| <b>A</b>    | <b>B</b>    | <b>C</b>    | <b>D</b>    | <b>E</b>    | <b>F</b>    | <b>G</b>    | <b>H</b>    | <b>I</b>    | <b>J</b>    | <b>K</b>    |
| L<br>U-004C | M<br>U-004D | N<br>U-004E | O<br>U-004F | P<br>U-0050 | Q<br>U-0051 | R<br>U-0052 | S<br>U-0053 | T<br>U-0054 | U<br>U-0055 | V<br>U-0056 |
| <b>L</b>    | <b>M</b>    | <b>N</b>    | <b>O</b>    | <b>P</b>    | <b>Q</b>    | <b>R</b>    | <b>S</b>    | <b>T</b>    | <b>U</b>    | <b>V</b>    |
| W<br>U-0057 | X<br>U-0058 | Y<br>U-0059 | Z<br>U-005A |             |             |             |             |             |             |             |
| <b>W</b>    | <b>X</b>    | <b>Y</b>    | <b>Z</b>    |             |             |             |             |             |             |             |

## Encabezados

Encabezados H1 → 2.4 REM

Encabezados H4 / Subtítulo → 1 REM

Encabezados H2 → 1.6 REM

Párrafo / Botones → 0.875 REM

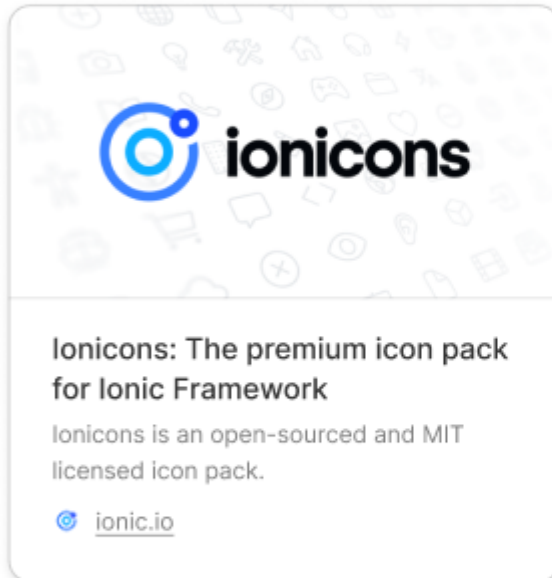
Encabezados H3 → 1.2 REM

Pizarra Figma “Fuente” 56

# Iconografía

## Librería

Ionicons 7.1.0: <https://ionic.io/ionicons>



Ejemplos:



Pizarra Figma "Iconografía" 57

## Botones

### Tamaño de los botones



### Botones Link



Pizarra Figma "Botones" 58

## Logotipos

### Logo Escudo



### Logo de Cabecera



Pizarra Figma "Logotipos" 59





## 4. Implementación del sistema

En la siguiente sección vamos a presentar los trabajos llevados a cabo para la construcción de la herramienta.

### 4.1 Herramientas utilizadas para la implementación

A continuación, procederemos a enumerar y describir las diferentes herramientas que hemos usado a lo largo del desarrollo de la aplicación, entre las cuales se incluyen diferentes frameworks [8][9], IDEs [10][11][12] y servicios [13].

Frameworks:

- Symfony (versión 3.4.49) [8]: “Symfony es un framework de trabajo para el lenguaje de PHP, de código abierto, que se utiliza en el desarrollo de aplicaciones web. Tiene una arquitectura modular que se basa en componentes y permite la reutilización de los mismos, por lo que es una gran opción para el desarrollo de un backend y su integración con servicios en la nube”.

Tal y como se indica en [8], “La arquitectura de Symfony se basa en el patrón de diseño “Modelo-Vista-Controlador” [22], lo que le permite una clara separación entre la lógica de negocio y la interacción con el usuario, lo cual es muy útil para que nuestro código sea más organizado, sostenible y escalable. La parte del Modelo se encarga de la gestión de datos y la lógica de negocio, que es donde se definen las entidades de nuestra aplicación, y la integración con la base de datos. La Vista se encarga de la presentación y la interfaz de usuario. Para la creación de las vistas, Symfony usa el motor de plantillas Twig, que permite crear vistas de forma sencilla y eficiente, además de tener una sintaxis muy sencilla e intuitiva, lo que facilita la personalización del aspecto visual. El Controlador es aquello que actúa de intermediario entre el modelo y la vista, este recibe las solicitudes del usuario, y se comunica con el modelo para poder procesar los datos y enviárselos a la vista para que se los pueda mostrar al usuario.”

Por último, cabe destacar, tal y como se indica en [14] que “Symfony también facilita la construcción de APIs, las cuales son esenciales para que la conexión entre las aplicaciones móviles y los servicios en la nube se realicen correctamente.”



Symfony

Logo Symfony 2.60

Como apoyo para el desarrollo de los servicios y mejor comprensión del framework hemos utilizado uno de los libros más destacados en la bibliografía [14].

- Ionic (versión 7.2.0) [9]: “Ionic es un framework de código abierto, al igual que Symfony, que está basado en tecnologías web tales como HTML, SCSS, TS (typescript). Ionic nos permite desarrollar aplicaciones móviles híbridas tanto para iOS como para Android. Se integra con los framework de frontend más populares como Angular, Vue y React, en este desarrollo estaremos usando Angular, que está basado en una arquitectura modular y orientada a componentes”.

En [9] se indica que, “la arquitectura de Angular se basa en módulos, que a su vez agrupan componentes, y cada componente está compuesto por una plantilla HTML, una hoja de estilos SCSS y un archivo TS que controla la lógica de la aplicación”.

Al igual que Symfony, Ionic también facilita la conexión con los servicios en la nube mediante APIs, lo que permite una sincronización a tiempo real de los datos. Además, como se indica en [9][15], “Ionic está basado en Capacitor, que es una plataforma que permite a los desarrolladores integrar funcionalidades en las aplicaciones tales como el uso de la cámara (el cual usaremos para el escaneo de QR en nuestra aplicación) o el GPS entre otras”.



Logo IONIC 2.61

En Ionic hemos trabajado con las tecnologías de typescript y java para las cuales, para una mejor comprensión y control de las mismas, nos hemos apoyado en la bibliografía más destacada de las mismas [15][16].

## IDEs:

- PhpStorm (versión 2024.2.3) [10]: El IDE que usaremos durante la mayor parte de la implementación y el desarrollo de la aplicación será PhpStorm, un IDE desarrollado por JetBrains que, a pesar de ser de pago, podremos usarlo gracias a la licencia que nos proporciona la universidad. Como se indica en [10], “está diseñado específicamente para el desarrollo en PHP, pero también es compatible con tecnologías web tales como HTML, SCSS y typescript, por lo que lo usaremos tanto para el backend con Symfony como para el frontend con Ionic”.



Logo PhpStorm 2.62

- Android Studio (versión 2024.2.1) [11]: Usaremos Android Studio, un IDE oficial de Google, para el desarrollo y la prueba de la aplicación en dispositivos Android (teléfonos, relojes, tablets...). Este IDE es completamente gratuito y está basado en IntelliJ IDEA, también de JetBrains, y nos proporcionará herramientas para el desarrollo en Kotlin, Java y Javascript. Android Studio también nos proporciona un emulador, que nos permitirá probar nuestra aplicación en diferentes dispositivos y versiones de Android, aunque algunas funcionalidades no pueden probarse con este emulador, y deberemos utilizar un dispositivo físico real.

Android Studio también nos proporciona las herramientas necesarias para la generación de los archivos APK y AAB (Android App Bundle), que serán necesarias para la correcta subida de la aplicación a la Google Play Store como veremos más adelante.



IDE Android Studio 2.63

- XCode [12]: “XCode es el IDE oficial de Apple para la compilación, prueba y despliegue de aplicaciones en dispositivos iOS (iPhone, iPads...)”. Es un IDE gratuito, pero tiene la limitación de que solo puede usarse en dispositivos macOS y, al igual que Android Studio, nos permitirá probar nuestra aplicación a

través de emuladores. En este caso, tal y como se indica en [12] “la herramienta de emulación se llama iOS Simulator, y nos permite probar la aplicación en diferentes modelos de Apple”. También nos permitirá generar el archivo IPA (iOS AppStore Package), que será necesario para la subida de la aplicación a la App Store como veremos más adelante.



Logo XCode IDE 2.64

Servidor Web para los microservicios:

- Nginx [18] es un servidor web/Proxy inverso ligero de alto rendimiento (Actualmente compite con Apache Server por dominar el mercado). Es software libre y de código abierto, con Licencia BSD simplificada. Es un sistema multiplataforma (GNU/Linux, BSD, Solaris, Mac OS X, Windows etc).



Logo NGINX 65

Herramientas de control de versiones:

- BitBucket[13]: Como sistema de control de versiones hemos usado BitBucket, una herramienta para alojamiento de código basada en Git, la cual nos facilitará el desarrollo de nuevas funcionalidades sin afectar al código en producción, mediante su sistema de ramas y commits.



Logo Bitbucket 2.66

## 4.2 Arquitectura implementada

En este apartado, vamos a desarrollar como está montada la arquitectura que soporta nuestra aplicación.

La arquitectura de la aplicación está basada en un modelo cliente-servidor, donde el cliente, que en este caso es nuestra aplicación móvil, se comunica con nuestro servidor, que utiliza Nginx [18] como servidor web y este a su vez se comunica con Symfony [8] como framework de backend. El servidor se encargará de recibir las peticiones externas y redirigirlas a Symfony, el cual procesa la lógica de negocio. Y por última, el backend con Symfony se conecta a nuestra base de datos, que en este caso es MySQL [17].

El cliente está desarrollado en Ionic [9] con Angular mediante nuestra aplicación, desde la cual los usuarios interactúan con el sistema a través de la interfaz, y realizan peticiones HTTP que tendrá que resolver el servidor, peticiones como iniciar sesión o consultar noticias, entre otras.

El servidor backend está desarrollado con el framework Symfony, el cual explicaremos más adelante, y utiliza la versión de PHP 7.4.24. Aquí es donde se maneja toda la lógica de negocio, Symfony se encarga de interpretar las peticiones que recibe, redirigirlas al controlador correspondiente que las vaya a tratar, acceder a la base de datos si es necesario para la petición, y generar las respuestas en formato JSON, que acabarán llegando al cliente de vuelta.

El servidor está montado sobre un entorno de producción con Nginx [18] como servidor web, que actúa como intermediario entre el cliente y el backend. La función del servidor web es recibir las peticiones que el cliente hace desde la aplicación móvil, y redirigírselas al backend para que pueda procesarlas correctamente.

Por último, la base de datos está montada con MySQL[17], y es donde se almacenarán todos los datos necesarios en el sistema, como por ejemplo usuarios, equipos, eventos, y todo lo relacionado con ellos. Symfony nos permite mapear las entidades del sistema (Usuario, Equipo, Eventos...) a través de anotaciones dentro de las clases. Estas anotaciones nos permiten definir correctamente los atributos y las relaciones entre entidades que se pasarán a las tablas en la base de datos.

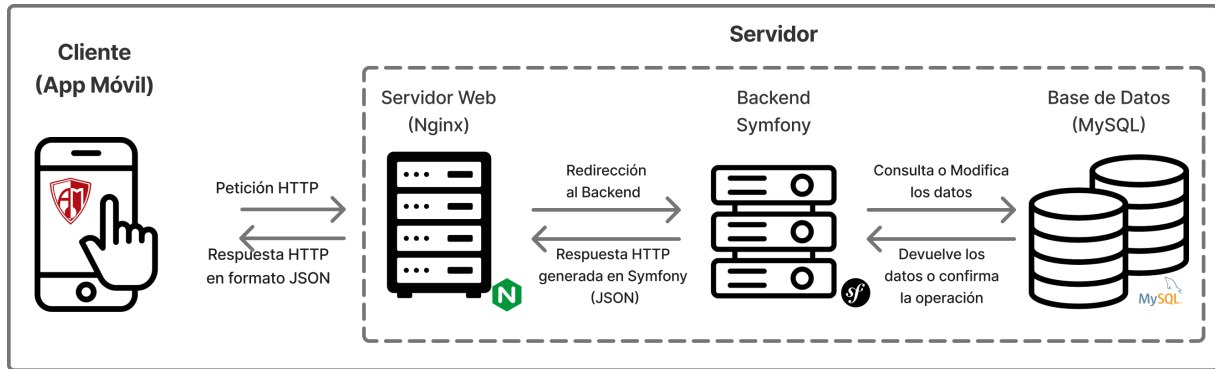


Diagrama arquitectura implementada 67

### 4.3 Servicios implementados

- Escaneo de QR: La implementación del escaneo de entradas a través de un código QR fue uno de los puntos clave durante el desarrollo de la aplicación. Esta funcionalidad es de vital importancia para poder llevar a cabo el conteo de entradas en un partido, y así evitar posibles fraudes, dado que cada abonado lleva su entrada única en el abono que tiene en la aplicación, y una vez escaneada para un evento quedaría registrada, haciendo que, si se vuelve a escanear el mismo QR, se muestre un error al administrador diciendo que ya ha sido escaneada previamente.

Para poder llevar a cabo este servicio, fue necesario un plugin compatible con Ionic y capacitor, que nos facilitase el escaneo de QR y además fuese compatible con la arquitectura de los dispositivos y con la estructura del proyecto. Tras probar varias opciones de plugin que fueron descartadas por conflictos con el resto de dependencias ya existentes en el proyecto, se acabó optando por el plugin “capacitor-mlkit”, que nos proporcionaba todas las funcionalidades que requeríamos.

Para la implementación en el código de este plugin fue necesario añadir unas cuantas funciones para poder integrarlo correctamente. Por ejemplo, la función “verificarPermisos()”, que pide los permisos si no están ya solicitados en la aplicación, y muestra una ventana de error si estos permisos se deniegan.

```

98     private async verificarPermisos(): Promise<boolean> {
99         try {
100             let status :PermissionStatus = await BarcodeScanner.checkPermissions()
101
102             // Si el permiso no está concedido, solicitamos permisos
103             if (status.camera !== "granted") {
104                 status = await BarcodeScanner.requestPermissions()
105             }
106
107             if (status.camera === "granted") {
108                 return true
109             } else {
110                 console.error("Permiso de cámara denegado")
111                 await this.globalVars.mostrarToast("Permiso de cámara denegado", "bottom", 3000)
112                 return false
113             }
114         } catch (error) {
115             console.error("Error al verificar permisos:", error)
116             await this.globalVars.mostrarToast("Error al verificar permisos", "bottom", 3000)
117             return false
118         }
119     }

```

Figura código de “Escaneo QR” 68

Pero la función principal en este servicio es “escaneoContinuo()”, que ayudada de otras funciones auxiliares como “comprobarCodigo()”, es la que permite que el escaneo se mantenga después de cada entrada, facilitando así la tarea del administrador, y deja un pequeño tiempo de espera para que no haya duplicados en el escaneo y se puedan mostrar correctamente los mensajes de validación o error (lo cual fue uno de los problemas principales a solucionar una vez se empezó a probar la aplicación en hardware físico).

```

142     // Evitar escaneos duplicados en un corto periodo de tiempo
143     if (codigo === this.lastScannedCode && currentTime - this.lastScanTime < this.SCAN_COOLDOWN) {
144         return
145     }
146
147     this.scanPaused = true
148
149     this.lastScannedCode = codigo
150     this.lastScanTime = currentTime
151
152     console.log("Código escaneado:", codigo)
153     const codigoValido :boolean = await this.comprobarCodigo(codigo)
154     this.ultimoEscaneoEsConCamara = true
155

```

Figura código de “Escaneo QR” 69

- Mostrar plantilla a tiempo real: Otro de los servicios que se han implementado ha sido el de mostrar la plantilla actual del equipo de forma dinámica, mostrando los jugadores que actualmente están marcados como activos en el backend. De esta forma no sería necesario cambiar la vista de jugadores cada vez que en una nueva temporada se actualice la plantilla, bastaría con incluir o borrar los jugadores en la zona de administración del backend para que se cambie automáticamente en la aplicación. Esta funcionalidad se implementa a través de una API que consulta todos



los jugadores activos en la base de datos, los diferencia por posición, y dentro de cada posición los ordena de manera ascendente, para que la aplicación pueda mostrarlos correctamente y de forma actualizada.

```
public function obtenerPlantilla($serializer, Equipo $equipo)
{
    $em = $this->getEntityManager();
    $eq = [];

    // todo: ordenar por dorsal
    $eq['portero'] = $em->getRepository('AppBundle:Jugador')->findBy(['equipo' => $equipo, 'posicion' => 'portero'], ['dorsal' => 'ASC']);
    $eq['cierre'] = $em->getRepository('AppBundle:Jugador')->findBy(['equipo' => $equipo, 'posicion' => 'cierre'], ['dorsal' => 'ASC']);
    $eq['ala'] = $em->getRepository('AppBundle:Jugador')->findBy(['equipo' => $equipo, 'posicion' => 'ala'], ['dorsal' => 'ASC']);
    $eq['pivot'] = $em->getRepository('AppBundle:Jugador')->findBy(['equipo' => $equipo, 'posicion' => 'pivot'], ['dorsal' => 'ASC']);
    $eq['cuerpo técnico'] = $em->getRepository('AppBundle:Jugador')->findBy(['equipo' => $equipo, 'posicion' => 'cuerpo técnico'], ['dorsal' => 'ASC']);

    $eq = $serializer->serialize([
        'ok' => true,
        'data' => $eq
    ], 'json', SerializationContext::create()->enableMaxDepthChecks()->setGroups(['api-publica'])->setSerializeNull(true));
}
```

Figura código de “Escaneo QR” 70

- Mostrar jornadas de la temporada actualizadas: A la hora de listar las jornadas de la presente temporada, se quería que los resultados de los diferentes partidos se fuesen actualizando en tiempo real conforme los resultados se suban al backend. Para lograr esto se decidió que, a través de una API que permita obtener las jornadas (ya terminadas, y próximas) para que en la aplicación podamos diferenciarlas mostrando el resultado de las ya terminadas, y la fecha y hora de las que están por venir.

```
$jornadasTerminadas = $em->createQueryBuilder()
->select( select: 'j')
->from( from: 'AppBundle:Jornada', alias: 'j')
->andWhere('j.temporada = :temporada')
->andWhere('j.golesLocal IS NOT NULL AND j.golesVisitante IS NOT NULL')
->setParameters($parametros)
->orderBy( sort: 'j.numJornada', order: 'ASC')
->getQuery()
->getResult();

$jornadasNoTerminadas = $em->createQueryBuilder()
->select( select: 'j')
->from( from: 'AppBundle:Jornada', alias: 'j')
->andWhere('j.temporada = :temporada')
->andWhere('j.golesLocal IS NULL AND j.golesVisitante IS NULL')
->setParameters($parametros)
->orderBy( sort: 'j.numJornada', order: 'ASC')
->getQuery()
->getResult();
```

Figura código de “Mostrar Jornadas” 71

- Mostrar asistencia de los eventos: Uno de los servicios fundamentales para la administración del club es la revisión de la asistencia a un evento. Esta funcionalidad le permite a los administradores y gestores del club conocer el dato en tiempo real de los asistentes a un evento determinado. Para poder implementarlo, a través de una API se recogen de la base de datos los datos relacionados con las entradas

escaneadas en un evento concreto, así como la información asociada a dichas entradas y/o abonos. Esta información se consulta de forma dinámica, permitiendo así mostrar al número de asistentes al evento, y el número de asistentes asociados a cada tipo de entrada.

```
public async listarTaquilla() {
    this.cargandoRegistros = true;

    try {
        const respuesta: any = await this.apiService.consulta(
            this.globalVars.urlListarTaquilla,
            'POST',
            { idEvento: this.eventoSeleccionado, idTipo: this.tipoSeleccionado, tipo: 'A' },
            true
        );

        console.log('respuesta: ', respuesta);
        if (respuesta.ok) {
            if (respuesta.body && respuesta.body.ok) {
                this.registros = respuesta.body.data;
                this.calcularTotalCantidad();
            } else {
                await this.globalVars.mostrarToast(respuesta.body.error, 'bottom', 3000);
            }
        } else {
            await this.globalVars.mostrarToast('Se ha producido un error', 'bottom', 3000);
        }
    } catch (error) {
```

Figura código de “Mostrar Asistencia” 72

- Mostrar recaudación en taquilla de los eventos: De igual manera que lo hacemos con la asistencia, queremos conocer la recaudación en taquilla para un evento. Para ello, no es muy diferente a el proceso que se hace con la asistencia, de hecho, usamos la misma API, pero cambiando el tipo seleccionado, para que, en lugar de devolvernos el número de asistentes, nos devuelva la cantidad recaudada, que depende también del tipo de entrada.

```
public async listarTaquilla() {
    this.cargandoRegistros = true;

    try {
        const respuesta: any = await this.apiService.consulta(
            this.globalVars.urlListarTaquilla,
            'POST',
            { idEvento: this.eventoSeleccionado, tipo: "T" },
            true
        );

        console.log('respuesta: ', respuesta);
        if (respuesta.ok) {
            if (respuesta.body && respuesta.body.ok) {
                this.registros = respuesta.body.data;
                this.calcularTotal();
            } else {
                await this.globalVars.mostrarToast(respuesta.body.error, 'bottom', 3000);
            }
        } else {
            await this.globalVars.mostrarToast('Se ha producido un error', 'bottom', 3000);
        }
    } catch (error) {
```

Figura código de “Mostrar Taquilla” 73

- Mostrar noticias: Un servicio clave para los aficionados y usuarios del club, es la visualización de noticias relacionadas con el club. Esta funcionalidad permite a los usuarios estar informado sobre cualquier anuncio o artículo acerca del club para la implementación de este servicio se conecta a través de una API con la web oficial del club, donde se encuentran todas las noticias ordenadas por categorías, y se traen dichas noticias con todos sus datos asociados.

```
public async listarPosts(body: any) {
  try {
    const respuesta: any = await this.apiService.consulta(this.globalVars.uriListarNoticias, 'POST', body, false);
    console.log('respuesta: ', respuesta);

    if (respuesta.ok) {
      if (respuesta.body && respuesta.body.ok) {
        this.registros = this.registros.concat(respuesta.body['data']);
        this.paginadorRegistros = respuesta.body['meta'];
        console.log('registros:', this.registros);
        this.registrosCargados = true;
      } else {
        await this.globalVars.mostrarToast(respuesta.body.error, 'bottom', 3000);
      }
    } else {
      await this.globalVars.mostrarToast('Se ha producido un error', 'bottom', 3000);
    }
  } catch (error) {
    console.error('Error in listarPosts:', error);
    await this.globalVars.mostrarToast('Se ha producido un error inesperado', 'bottom', 3000);
  }
}
```

Figura código de “Mostrar Noticias” 74

## 5. Pruebas del software

Llegados a este punto, y tras haber desarrollado la aplicación, nos hemos centrado en realizar diferentes pruebas al software para asegurar su seguridad, su calidad y que realmente realiza las operaciones para las que fue diseñado. Para ello, se van a realizar pruebas de análisis de código estático para asegurar la seguridad y calidad del código y pruebas manuales que sigan todos los flujos posibles marcados en los casos de uso, las cuales nos aseguran que la aplicación hace realmente lo que se le pide.

### 5.1 Pruebas de análisis de código estático: Seguridad y calidad

En esta sección de pruebas, vamos a realizar una batería de pruebas con la ayuda del software SonarQube. SonarQube [19] es una herramienta de análisis estático de código que nos permite detectar los errores, vulnerabilidades y problemas de mantenimiento que pueda tener nuestro código mediante el análisis de algunas métricas, lo cual nos permite realizar un seguimiento detallado del proyecto y mejorar la calidad de nuestro código.



Logo SonarQube 75

Para configurar SonarQube primero descargamos la versión “community build”, que es la versión gratis y de código abierto. Después, basta con lanzar el servidor y una vez iniciado acceder a la interfaz web a través del puerto asignado “localhost:9000”. Tras entrar a la interfaz web e introducir nuestras credenciales, tendremos que elegir como analizar nuestro repositorio, en nuestro caso lo vamos a hacer de manera local desde la carpeta del propio proyecto, pero nos permite hacerlo mediante plataformas GitLab, GitHub o Bitbucket.

#### Analysis Method

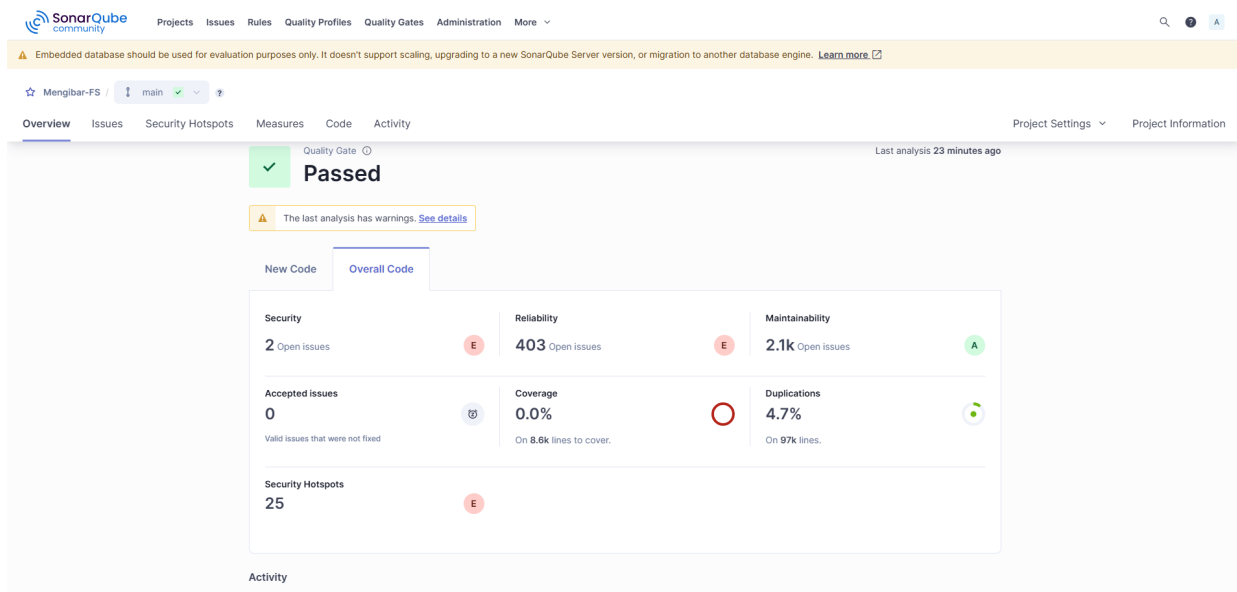
Use this page to manage and set-up the way your analyses are performed.

#### How do you want to analyze your repository?

|                                                                                                                                  |                                                                                                          |                                                                                                                  |
|----------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------|
|  With Jenkins                                 |  With GitHub Actions  |  With Bitbucket Pipelines   |
|  With GitLab CI                               |  With Azure Pipelines | <b>Other CI</b><br>SonarQube Community Build integrates with your workflow no matter which CI tool you're using. |
| <b>Locally</b><br>Use this for testing or advanced use-case. Other modes are recommended to help you set up your CI environment. |                                                                                                          |                                                                                                                  |

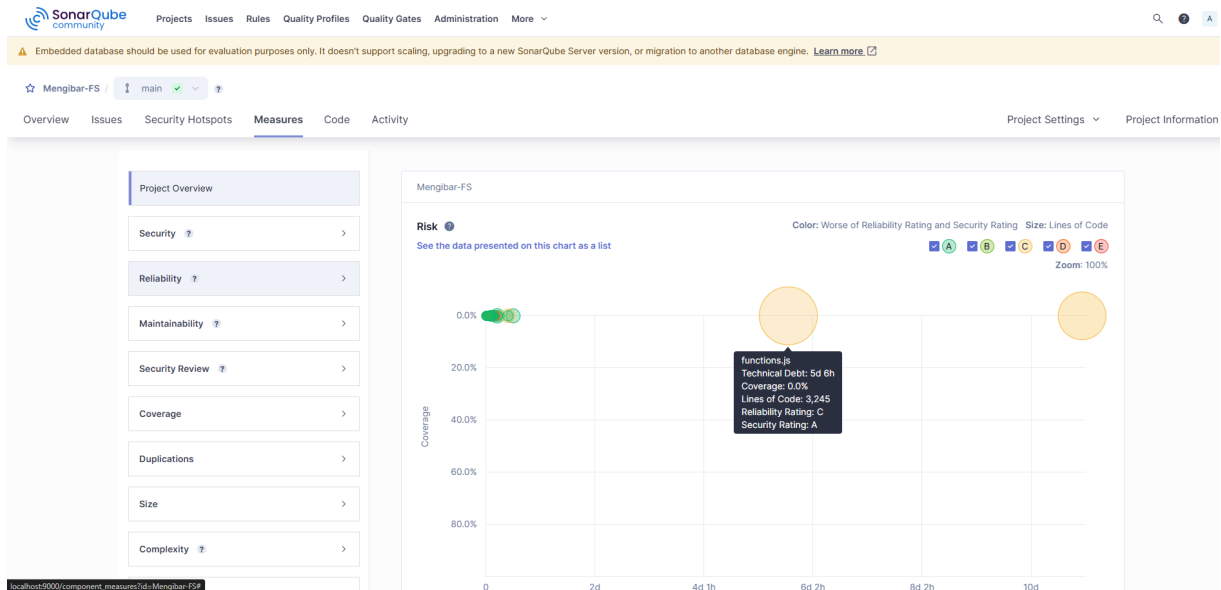
Figura Inicio SonarQube 5.76

Tras hacer el análisis del proyecto, nos mostrará un informe detallado con diferente información acerca del proyecto, y de forma gráfica, nos mostrará cuantos problemas y de que tipo tenemos en nuestro proyecto. El informe nos muestra diferentes las diferentes métricas que se han evaluado, y la puntuación que se le da al proyecto en cada una de ellas, como por ejemplo seguridad o mantenibilidad entre otras. También nos permite ver directamente cuales son los errores de cada métrica, y donde se encuentran, especificándonos el archivo y la línea concretos.



**Figura SonarQube en funcionamiento 5.77**

Además, SonarQube nos proporciona algunas herramientas muy útiles como gráficos en los que nos muestra la cantidad de errores, la importancia, e incluso la cantidad de tiempo estimada que nos llevaría solucionarlos, junto con especificaciones varias de los mismos, lo cual es muy útil para saber por dónde empezar.



**Figura SonarQube en funcionamiento 78**

Tras realizar el primer análisis podemos observar que nos aparecen algunos errores varios y notas bajas en las métricas de seguridad, fiabilidad y cobertura de código, y por otra parte nos aparecen buenas notas en las métricas de mantenibilidad y duplicación de código. Aun así, la nota general del proyecto nos aparece como aprobada (Quality Gate: passed), esto quiere decir que a pesar de los errores el proyecto sería totalmente funcional y estaría listo para pasar a la fase de producción sin ningún problema.

Después de arreglar algunos errores, ejecutamos un segundo análisis y podemos ver cómo se actualizan y se muestran las nuevas notas de cada métrica, además de una gráfica que nos dice cuántos errores se han corregido desde el último análisis. A su vez, si después del análisis aparece un nuevo error con el cambio, se puede ver cuál ha sido el problema en la sección “new code”. Por ejemplo, aquí hemos arreglado los 2 errores graves que teníamos de seguridad junto con algún error leve de fiabilidad, y ambas métricas nos han subido en nota.

Puesto que el código que tenemos es en parte heredado, nos bastará con solucionar los errores más importantes y que nos proporcionan malas notas en las métricas, ya que la gran mayoría de los errores que aparecen, no son errores que influyan en el funcionamiento normal de la aplicación. Estos errores se podrían corregir más adelante si se quiere hacer un seguimiento de calidad a fondo de la aplicación, enfocándose en estos detalles y con mucho más tiempo para ello.

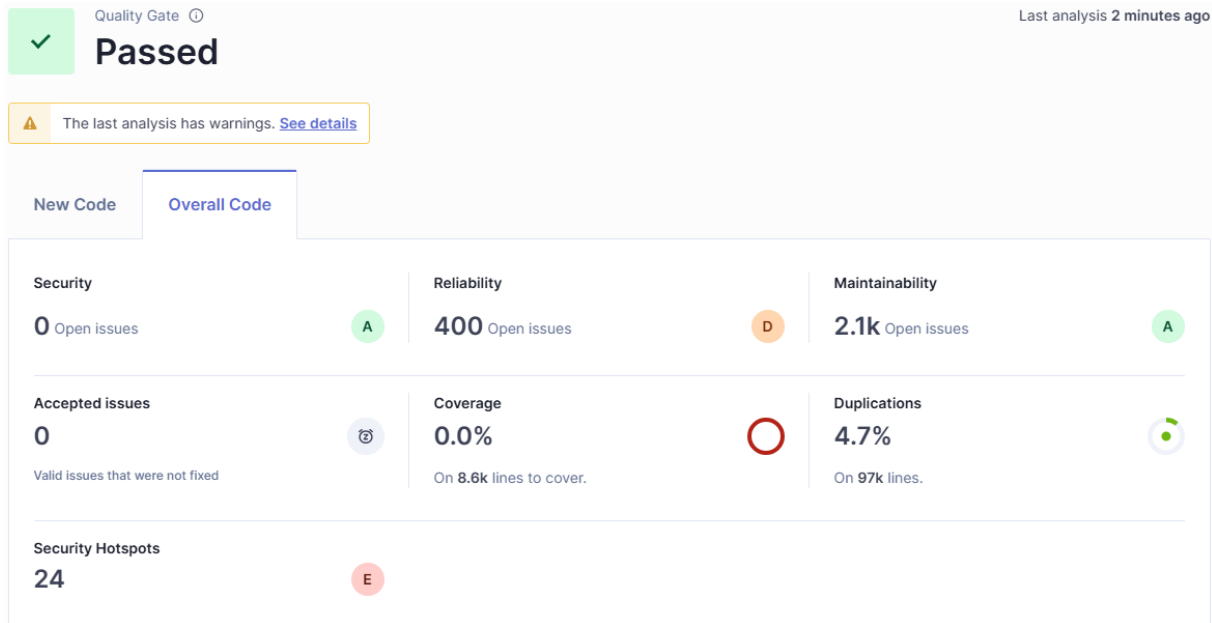


Figura SonarQube resultados 79

|                                        |                                     |
|----------------------------------------|-------------------------------------|
| April 2, 2025 at 12:27 PM              | Quality Gate: <span>✓</span> Passed |
| <b>NOT PROVIDED</b>                    |                                     |
| New analysis: <span>↘</span> -2 Issues |                                     |
| April 2, 2025 at 12:25 PM              | Quality Gate: <span>✗</span> Failed |
| New analysis: <span>↗</span> +1 Issues |                                     |
| April 2, 2025 at 12:16 PM              | Quality Gate: <span>✓</span> Passed |
| New analysis: <span>↘</span> -1 Issues |                                     |
| April 2, 2025 at 12:13 PM              | Quality Gate: <span>✗</span> Failed |
| New analysis: <span>↔</span> +0 Issues |                                     |
| April 2, 2025 at 12:10 PM              | Quality Gate: <span>✗</span> Failed |
| New analysis: <span>↘</span> -1 Issues |                                     |

[See full history of analyses](#)

Figura SonarQube resultados 80

También analizamos la parte front del proyecto, es decir, la aplicación móvil con Ionic. Como podemos observar en las métricas de este análisis, este tiene muchos menos problemas de inicio, solo unos pocos de fiabilidad, pero ninguno de seguridad, que serían los más urgentes y graves de reparar. Aun así, aquí también nos aparece la nota general como aprobada, ya que estos errores no impiden el correcto funcionamiento de la aplicación.

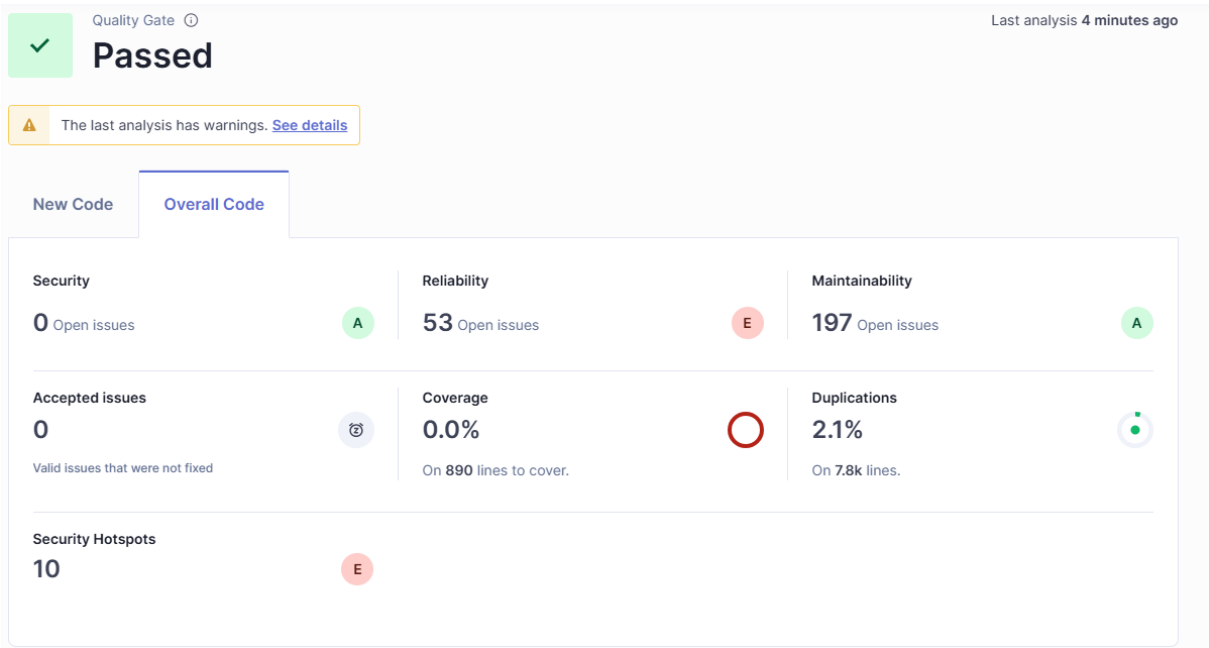


Figura SonarQube resultados 81

De igual manera también solucionamos los errores más importantes de este proyecto para darle unas mejores métricas.

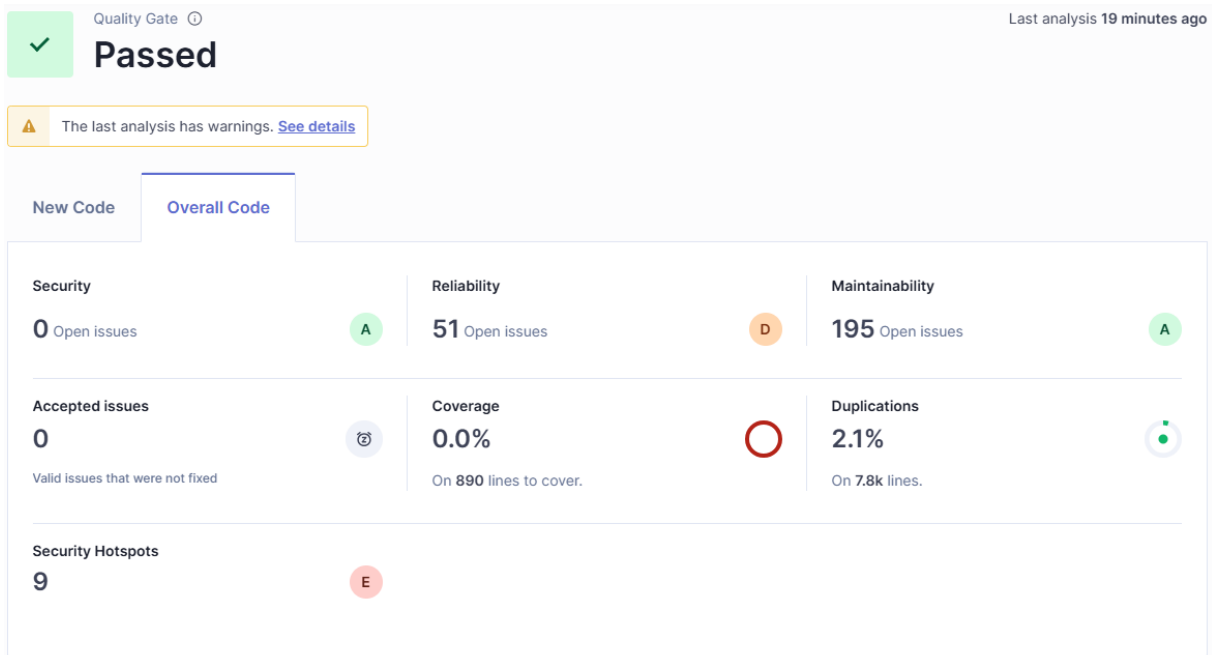


Figura SonarQube resultados reparados 82

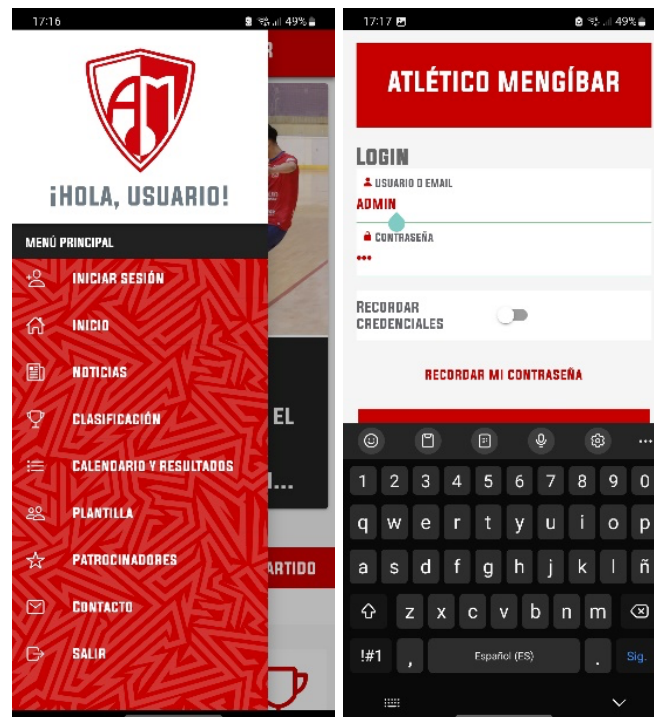


## 5.2 Pruebas manuales del software

En esta sección vamos a llevar a cabo pruebas manuales al software siguiendo los casos de uso planteados, para ver el correcto funcionamiento de la aplicación y sus servicios asociados.

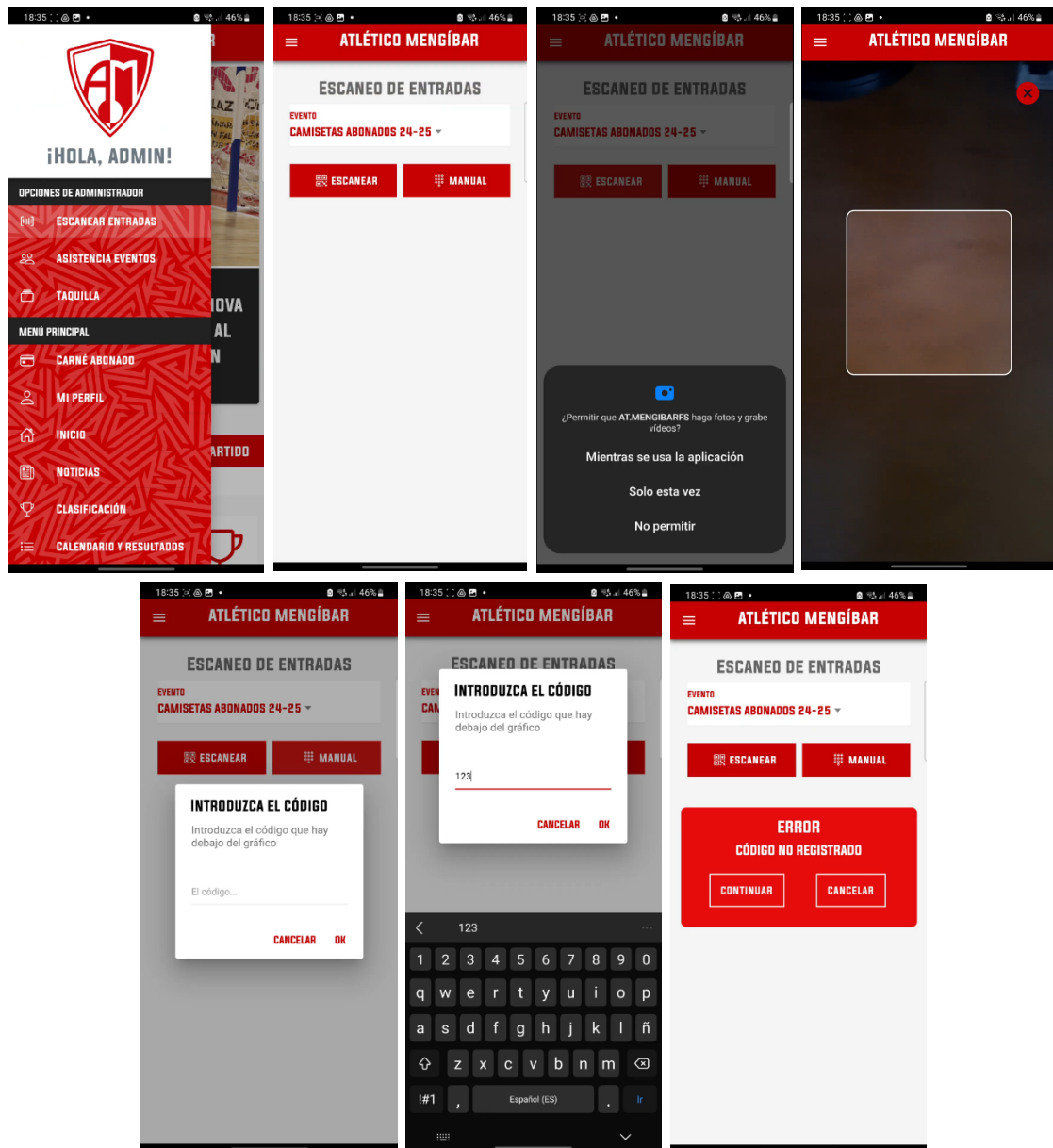
Probaremos la aplicación ya descargada en un dispositivo Android, en este caso un Samsung "Galaxy S20+ 5G". Para seguir un orden en las pruebas, probaremos con todos los casos de uso ya descritos anteriormente en el documento:

1. Inicio de sesión: Abrimos la aplicación y sin tener la sesión iniciada, abrimos el menú lateral y pulsamos el botón "Iniciar sesión", después introducimos las credenciales correctas y pulsamos en "Acceder" y comprobamos que se haya iniciado la sesión correctamente.



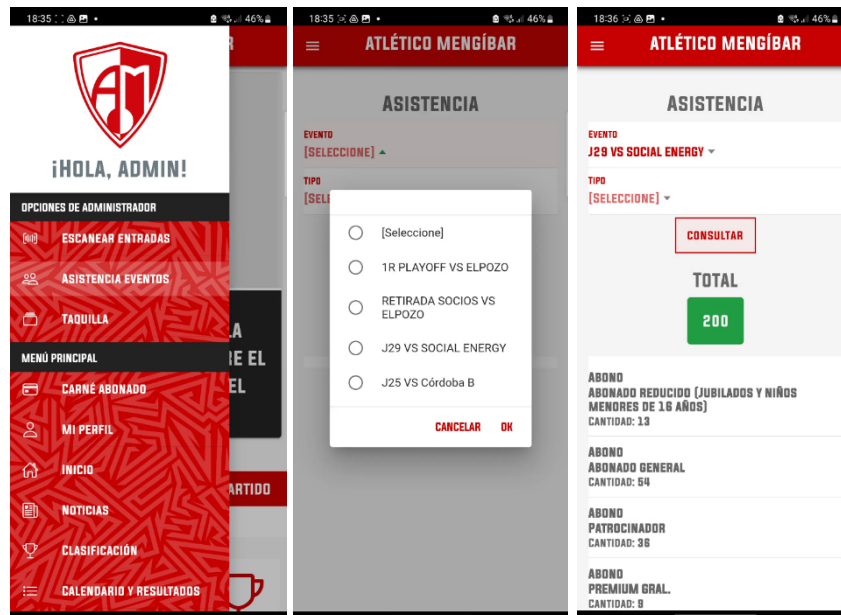
Pruebas manuales de Inicio de sesión 83

- Escaneo de Entradas: Abrimos la aplicación e iniciamos sesión como administrador, entonces, abrimos el menú lateral y pulsamos “Escanear Entradas”, se nos abrirá la vista de escaneo de entradas, y seleccionamos un evento. Después, pulsamos el botón de escaneo y el dispositivo nos pide los permisos necesarios para poder abrir la cámara, tras aceptar dichos permisos se abrirá la cámara con el escaneo de QR preparado. También probamos el escaneo manual introduciendo un código de prueba “123” que no debería funcionar, y efectivamente, nos da un error de validación en la entrada.



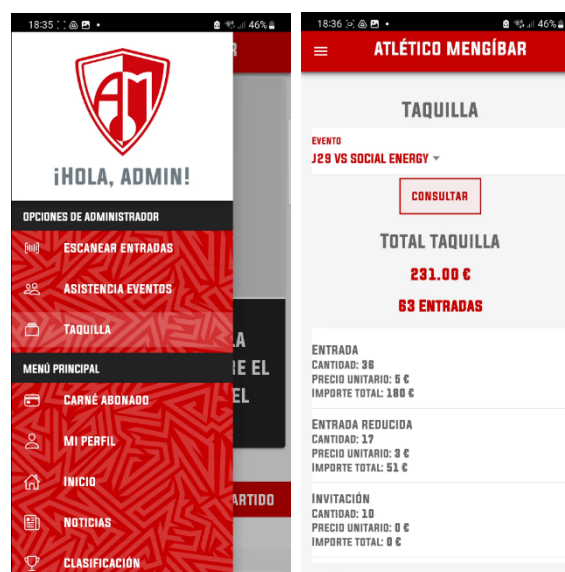
Pruebas manuales Escaneo de entradas 84

3. Asistencia a un evento: Abrimos la aplicación e iniciamos sesión como administrador, entonces abrimos el menú lateral y pulsamos “Asistencia Eventos”, se nos abrirá la vista de la asistencia, donde podremos seleccionar un evento concreto y un filtro que podremos aplicarle. Tras pulsar el botón “Consultar” se nos muestra una lista con los diferentes tipos de asistencia al evento.



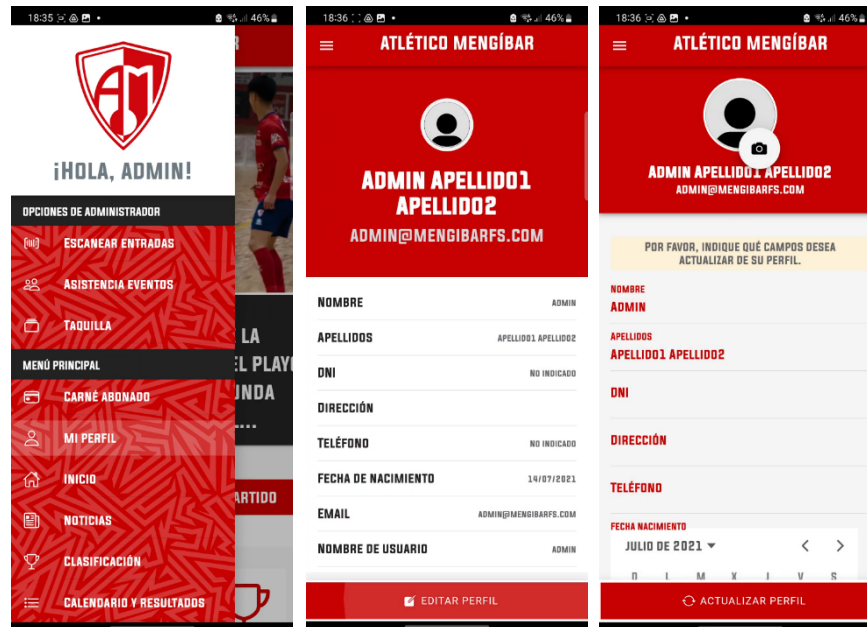
Pruebas manuales Asistencia 85

4. Taquilla de un evento: Abrimos la aplicación e iniciamos sesión como administrador, entonces abrimos el menú lateral y pulsamos “Taquilla”, se nos abrirá la vista de la taquilla, donde podemos seleccionar un evento concreto. Tras pulsar el botón “Consultar” se nos muestra una lista con la recaudación de la taquilla y de donde proviene esa recaudación



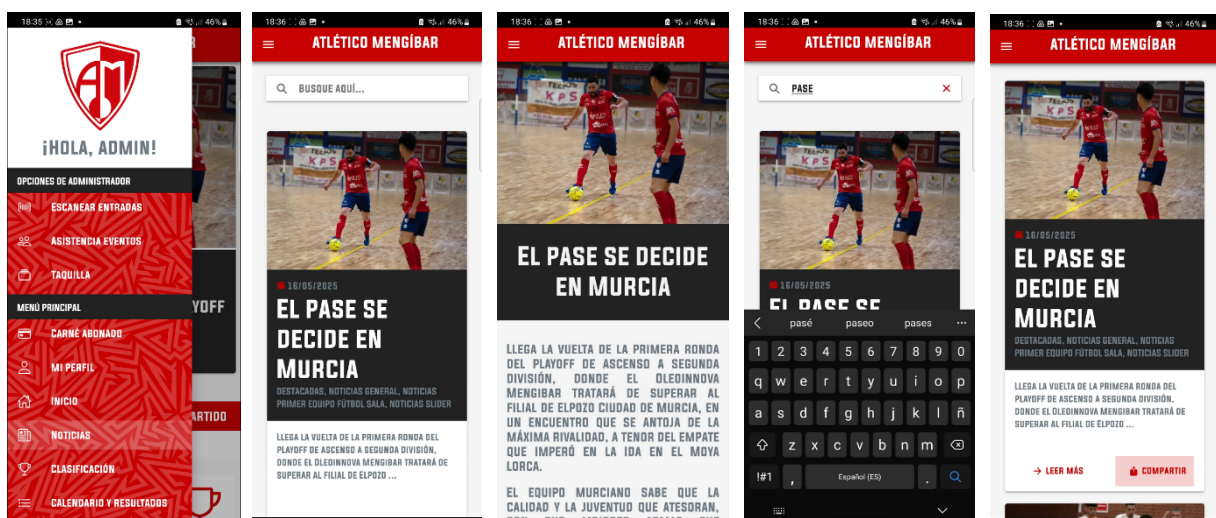
Pruebas manuales Taquilla 86

5. Editar perfil de usuario: Abrimos la aplicación e iniciamos sesión como administrador o usuario, entonces abrimos el menú lateral y pulsamos “Mi perfil”, se nos abrirá la vista del perfil de usuario con los diferentes datos introducidos en el registro. Si queremos cambiar algún dato pulsamos en “editar perfil” y nos llevará a la pantalla de edición de perfil, donde podremos modificar algunos de los datos.



Pruebas manuales Editar perfil 87

6. Ver noticias: Abrimos la aplicación y navegamos a la pestaña de “Noticias”, se nos abrirá una vista donde podremos navegar por las diferentes noticias, ver los detalles de una noticia concreta, compartir el enlace de la noticia, o escribir en el buscador y filtrar las diferentes noticias por palabras clave.



Pruebas manuales noticias 88

7. Clasificación: Abrimos la aplicación y navegamos a la pestaña “Clasificación”, tras pulsar el botón, nos mostrará una clasificación en tiempo real en la que podremos scrollear hacia abajo para revisarla entera, y debajo de la clasificación se mostrará una leyenda básica para el sistema de colores.



| POS | EQUIPO                      | PTS | PJ | PG | PE | PP | DG |
|-----|-----------------------------|-----|----|----|----|----|----|
| 1   | ELPOZO CIUDAD DE MURCIA     | 73  | 30 | 23 | 4  | 3  | 75 |
| 2   | CD AVANZA JAÉN              | 69  | 30 | 22 | 3  | 5  | 69 |
| 3   | XEREZ TOYOTA NIMAUTO        | 68  | 30 | 21 | 5  | 4  | 64 |
| 4   | OLEOINNOVA MENGIBAR FS      | 67  | 30 | 21 | 4  | 5  | 35 |
| 5   | JAÉN PARAÍSO INTERIOR F.S.B | 54  | 30 | 16 | 6  | 8  | 35 |
| 6   | CD VIRGILI CÁDIZ "A"        | 51  | 30 | 16 | 3  | 11 | 20 |
| 7   | C.D. MALACITANO FUTSAL      | 50  | 30 | 15 | 5  | 10 | 13 |
| 8   | C.D. CÓRDOBA FUTSAL         | 47  | 30 | 14 | 5  | 11 | 24 |

Pruebas manuales Clasificación 89

8. Calendario y partidos: Abrimos la aplicación y navegamos a la pestaña de “Calendario y resultados”, tras pulsar el botón, nos mostrará la lista de los partidos divididos en los ya jugados y sin jugar, y podremos scrollear hacia abajo en orden cronológico para revisar el calendario.

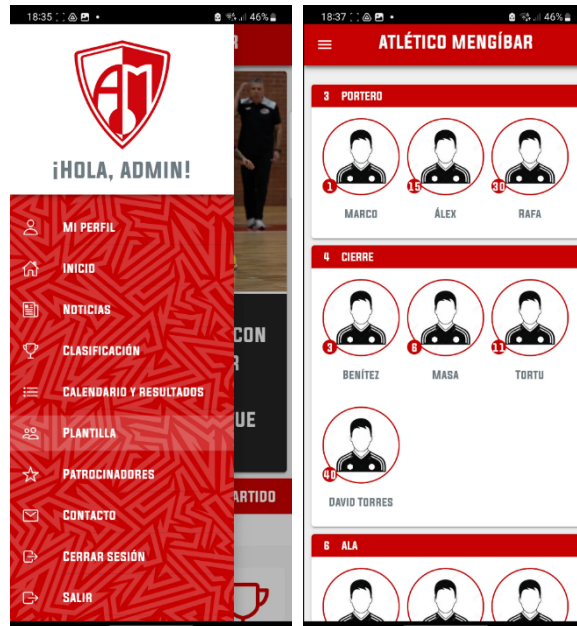


| FECHA   | HORA | EQUIPO 1                 | VS    | EQUIPO 2               | RESULTADO |
|---------|------|--------------------------|-------|------------------------|-----------|
| 14 SEP. |      | XEREZ TOYOTA NIMAUTO     | 4 - 0 | OLEOINNOVA MENGIBAR FS | TERMINADO |
| 21 SEP. |      | OLEOINNOVA MENGIBAR FS   | 4 - 3 | NAWTER BLANCA FS       | TERMINADO |
| 28 SEP. |      | REAL SOCIEDAD LA PANTERA | 1 - 4 | OLEOINNOVA MENGIBAR FS | TERMINADO |
| 5 OCT.  |      | OLEOINNOVA MENGIBAR FS   | 6 - 3 | C.D. BUJALANCE F.S.    | TERMINADO |

Pruebas manuales Calendario y partidos 90

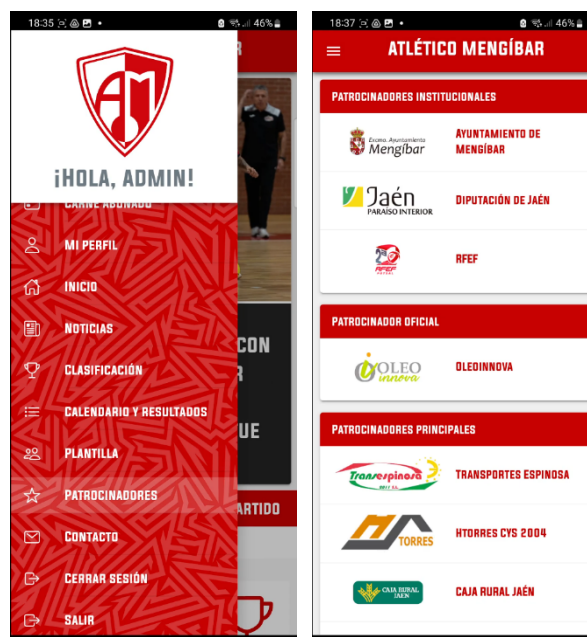


9. Plantilla: Abrimos la aplicación y navegamos a la pestaña de “Plantilla”, tras pulsar el botón, nos mostrará la lista de la plantilla actual activa del equipo divididos por posición, podremos scrollear hacia abajo para revisar los diferentes jugadores y sus fotos.



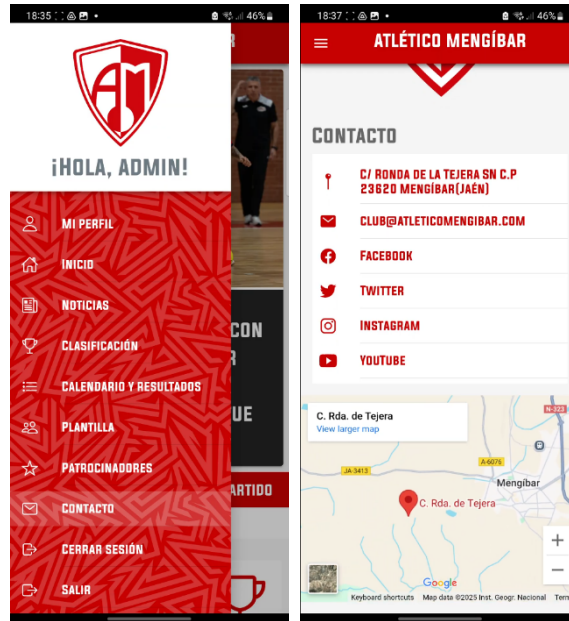
Pruebas manuales plantilla 91

10. Patrocinadores: Abrimos la aplicación y navegamos a la pestaña de “Patrocinadores”, tras pulsar el botón, nos mostrará la lista de los patrocinadores actuales del equipo, divididos por el tipo de patrocinio que tienen. Si hacemos clic en cualquiera de los patrocinadores (nombre o foto) nos llevará directamente a la página web asociada al patrocinador (por ejemplo, la página del ayuntamiento de Mengíbar)



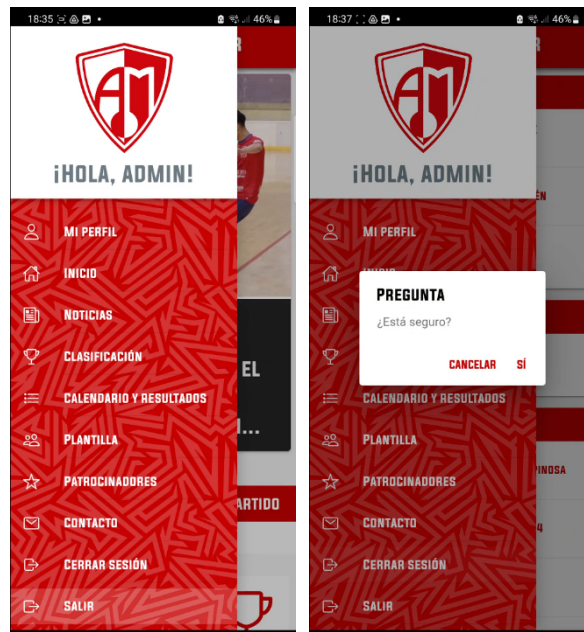
Pruebas manuales Patrocinadores 92

11. Contacto: Abrimos la aplicación y tras abrir el menú lateral y pulsamos “Contacto”, se nos abrirá la vista de la información de contacto. Podremos revisar los diferentes enlaces de redes sociales que nos mandarán a las aplicaciones externas de las redes sociales, y también nos encontraremos con un mapa interactivo, que podremos mover y visualizar pulsando con 2 dedos en la pantalla.



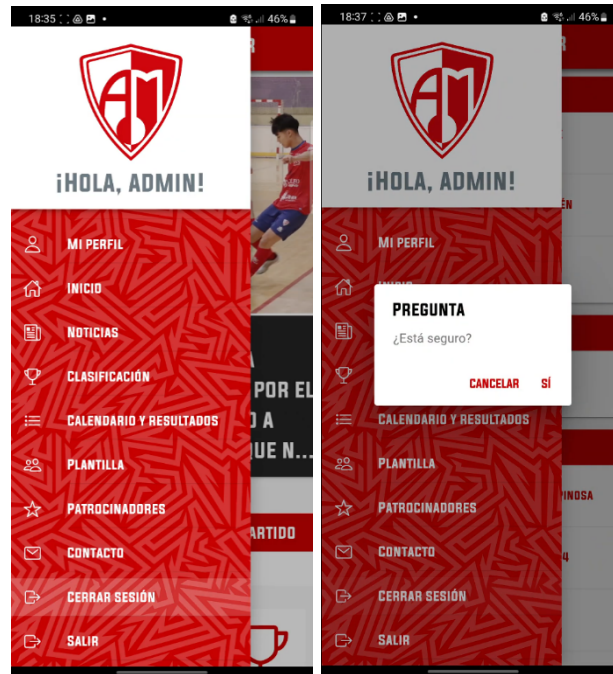
Pruebas manuales Contacto 93

12. Salir de la aplicación: Tras tener la aplicación abierta, para salir únicamente basta con abrir el menú lateral y pulsar el botón de “salir” que está abajo del todo y aceptamos el pop-up que aparece que nos pregunta si estamos seguros de salir, tras pulsar “Si” se cerrará la aplicación.



Pruebas manuales Salir 94

13. Cerrar sesión: Para cerrar sesión previamente habrá que haber iniciado sesión, ya que este botón no aparecerá si no está la sesión iniciada. Bastará con abrir el menú lateral y scrollear hasta el botón de “Cerrar sesión”, nos aparecerá el pop-up que pregunta si estamos seguros de salir, y tras pulsar “Sí” se cerrará la sesión de la aplicación, entonces se volverá a la pantalla de inicio, pero con la sesión ya cerrada.



Pruebas manuales Cerrar sesión 95





## 6. Conclusiones y trabajos futuros

Tras finalizar este trabajo fin de grado, me gustaría destacar la grata oportunidad de poner en práctica cada uno de los conocimientos adquiridos en cada una de las asignaturas impartidas a lo largo del grado, tanto en la gestión de proyectos como en métodos y tecnologías de desarrollo software.

Haciendo una revisión del trabajo realizado durante estos meses de desarrollo del mismo, como conclusión técnica me gustaría destacar el aprendizaje realizado durante el trabajo tanto en la creación de apps móviles, como en el desarrollo de funcionalidades mediante servicios. He tenido que aprender a crear nuevas funcionalidades para software ya existente basado en servicios y con tecnología actualmente obsoleta, la cual ha habido que actualizar también. Por otro lado, también he realizado un aprendizaje de un nuevo framework que permite construir aplicaciones móviles para los dos sistemas operativos más representativos del mercado y sus distintas peculiaridades para subir las apps a sus tiendas.

Por otro lado, en cuanto a los objetivos planteados al inicio del proyecto, una vez finalizado todo el trabajo y realizando una valoración justa, se pueden dar por cumplidos dentro del plazo estimado de tiempo gestionando las posibles variaciones de tiempo de una forma rápida y concisa.

Creo que las aplicaciones desarrolladas y los servicios van a ser de gran utilidad para el club y espero en el futuro poder seguir colaborando con ellos ampliando nuevas funcionalidades a la aplicación.

Como trabajos futuros, podríamos plantear realizar un sistema de recomendación basado en análisis predictivo de asistencia de público partiendo de los datos existentes en la base de datos. De esta manera, el club podría realizar una correcta previsión de impresión de entradas, compras en el Bar y de stock de camisetas, bufandas etc.



## Anexos

### A. Manual instalación y configuración servicios web en servidor

En este apartado describiremos el proceso que hemos seguido para la instalación y configuración de los servicios web necesarios para el correcto funcionamiento del backend de la aplicación. El sistema se ha desarrollado, como ya hemos comentado en anteriores ocasiones, con Symfony, por lo que es necesario disponer de un entorno que nos permita ejecutar aplicaciones PHP con acceso a la base de datos.

Dado que el proyecto ya tenía una base de datos creada previamente, y las migraciones ya se habían ejecutado, por lo tanto, no fue necesario crear las tablas mediante symfony. Aun así, para algunas clases y tablas fue necesario modificaciones en la base de datos, por lo tanto, se tuvieron que realizar nuevas migraciones en symfony, y estas se hicieron mediante los comandos “make:migration” y “doctrine:migrations:migrate”.

También configuramos adecuadamente el archivo “.env”, que es donde se especifican las variables de entorno del proyecto, como el nombre y la dirección de la base de datos, la contraseña y demás datos sensibles. Esta configuración es esencial para que la aplicación pueda comunicarse correctamente con la base de datos.

Para facilitar la configuración del entorno y evitar algunos problemas de dependencias, se optó por usar Docker, que con su tecnología de contenedores nos permite empaquetar todo el entorno en archivos de configuración. Gracias a Docker, podremos levantar la aplicación sin preocuparnos de instalar manualmente PHP, MySQL, Nginx o cualquiera de las extensiones necesarias.

En primer lugar, hubo que definir el archivo docker-compose.yml, que es el archivo que describe todos los servicios que necesita nuestro backend. Para ilustrar el proceso, en las siguientes capturas mostraremos un ejemplo de archivo docker-compose.yml similar al de nuestro proyecto. Las capturas se han tomado de un proyecto de prueba, ya que no podemos poner las del proyecto real por temas de privacidad y datos sensibles, pero sigue manteniendo la estructura principal de los servicios:

```
1 version: '3.5'
2
3 services:
4   symfony-skeleton-app:
5     container_name: symfony-skeleton-app
6     build:
7       context: ./docker
8       args:
9         UID: $U_ID
10    volumes:
11      - ./:/var/www/html
12      ##> XDEBUG 3 ##
13      # Use your client IP here
14      # Linux: run "ip a | grep docker0"
15      # Windows (with WSL2) and Mac: host.docker.internal
16    environment:
17      XDEBUG_CLIENT_HOST: host.docker.internal
18      XDEBUG_CLIENT_PORT: 9003
19      PHP_IDE_CONFIG: serverName=symfony-skeleton-server
20    ports:
21      - '1000:80'
22    networks:
23      - symfony-skeleton-app-network
24    depends_on:
25      - symfony-skeleton-app-mysql
26
27   symfony-skeleton-app-mysql:
28     container_name: symfony-skeleton-app-mysql
29     image: mysql:8.2.0
30     ports:
31       - '3336:3306'
32     environment:
33       MYSQL_DATABASE: symfony-skeleton-app
34       MYSQL_ROOT_PASSWORD: root
35     volumes:
36       - symfony-skeleton-app-mysql-data:/var/lib/mysql
37     networks:
38       - symfony-skeleton-app-network
39     command: [ 'mysqld', '--character-set-server=utf8mb4', '--collation-server=utf8mb4_unicode_ci' ]
40
```

Captura docker-compose.yml 96

```

41 symfony-skeleton-app-mailer:
42   container_name: symfony-skeleton-app-mailer
43   image: schickling/mailcatcher
44   ports:
45     - '1025:1025'
46     - '1080:1080'
47   networks:
48     - symfony-skeleton-app-network
49
50 ###> doctrine/doctrine-bundle ###
51 database:
52   image: postgres:${POSTGRES_VERSION:-16}-alpine
53   environment:
54     POSTGRES_DB: ${POSTGRES_DB:-app}
55     # You should definitely change the password in production
56     POSTGRES_PASSWORD: ${POSTGRES_PASSWORD:-!ChangeMe!}
57     POSTGRES_USER: ${POSTGRES_USER:-app}
58   volumes:
59     - database_data:/var/lib/postgresql/data:rw
60     # You may use a bind-mounted host directory instead, so that it is harder to accidentally remove the volume and lose all your data!
61     # - ./docker/db/data:/var/lib/postgresql/data:rw
62 ###< doctrine/doctrine-bundle ###
63
64 networks:
65   symfony-skeleton-app-network:
66     name: symfony-skeleton-app-network
67
68 volumes:
69   symfony-skeleton-app-mysql-data:
70   database_data:

```

Captura docker-compose.yml 97

En este archivo se definen los servicios de “php”, “nginx” y la base de datos.

Una vez definido el entorno de ejecución, el siguiente paso fue adaptar el archivo .env del proyecto Symfony para que usase los datos del contenedor de base de datos. La línea correspondiente quedaría algo así:

“DATABASE\_URL = ”mysql://symfony:secret@db:3306/symfony\_db”

Con esto, Symfony ya sería capaz de comunicarse con el de la base de datos, utilizando el y las contraseñas asignados.

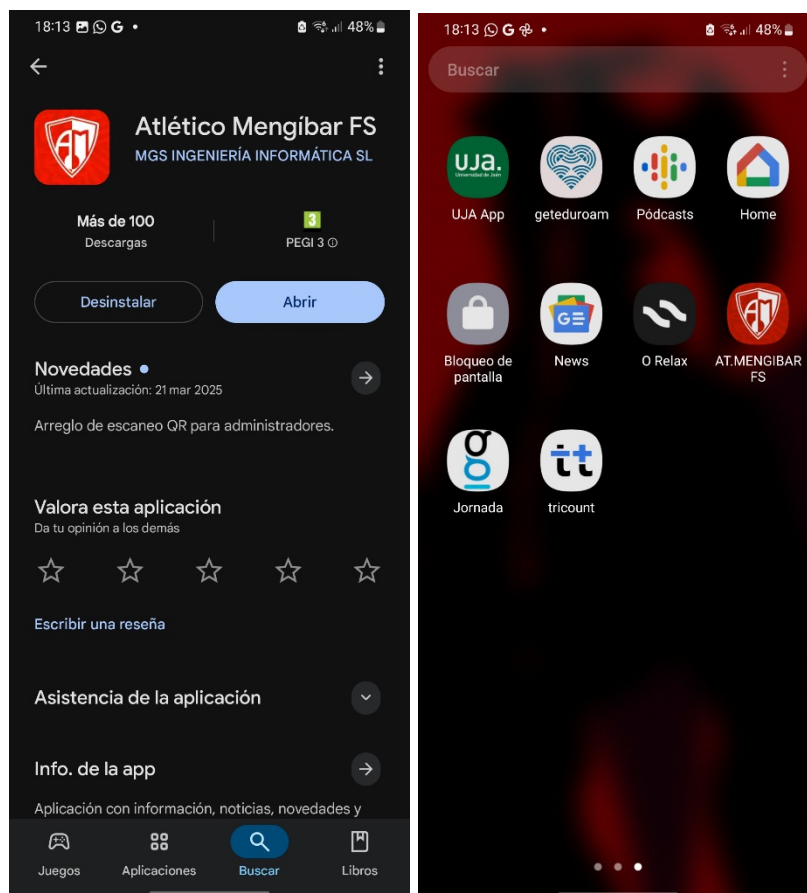
Una vez tengamos todos los archivos bien configurados, bastaría con ejecutar el comando para iniciar todo el entorno de desarrollo, que en este caso es “docker-compose up -d”. Tras unos pocos segundos de carga, podremos acceder a la aplicación desde el navegador a través de <http://localhost:8080>, donde ya podremos comprobar que el backend de Symfony está en funcionamiento, y podremos continuar con las tareas de desarrollo, prueba y despliegue.

## B. Manual instalación y funcionamiento Aplicación móvil

En el siguiente anexo veremos los requisitos y pasos necesarios para llevar a cabo una correcta instalación de la aplicación móvil, y como ponerla en funcionamiento.

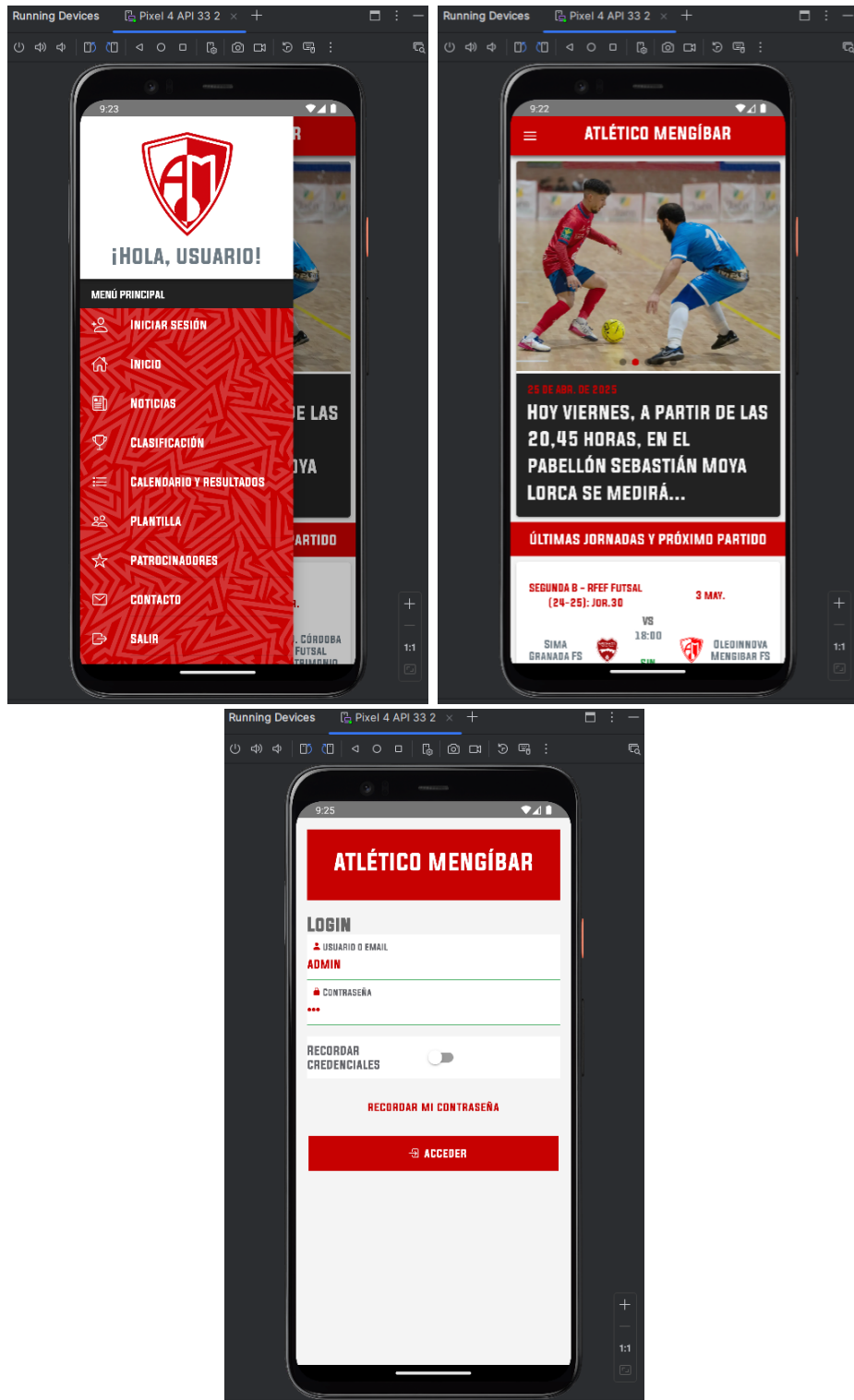
En primer lugar, será necesario que dispongamos de un dispositivo móvil con las características necesarias, es decir, en el caso de nuestra aplicación los requerimientos mínimos son un sistema operativo compatible, tanto iOS como Android valdría porque la aplicación está subida en ambas tiendas, en el caso de iOS la versión mínima compatible con nuestra aplicación sería iOS 18, y en el caso de Android la versión 5.1, que abarca la gran mayoría de dispositivos activos a día de hoy. También será necesario disponer de un espacio mínimo en el almacenamiento del dispositivo, en torno a unos 26 MB.

Si cumplimos estos requisitos el siguiente paso será tan simple como buscar la ficha de la aplicación en la tienda correspondiente al sistema operativo (Google Play Store o Apple Store), y darle al botón de “Instalar”. Una vez instalada la aplicación, basta con abrirla y nos enviará directamente a la vista “Home” donde ya podremos manejarnos libremente y navegar con normalidad.



## B.1 Hacer Login

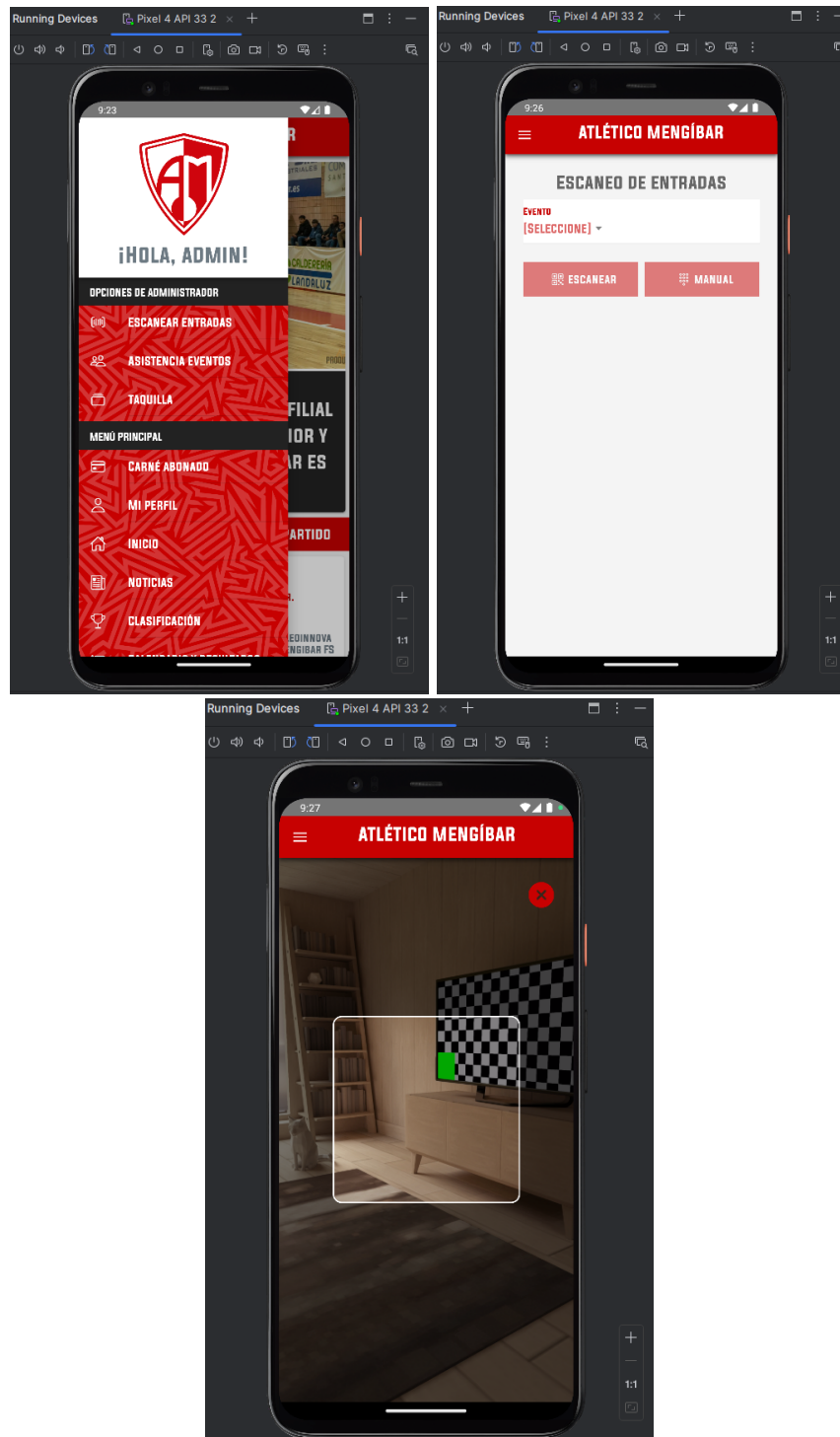
Para hacer login el usuario deberá tener una cuenta previamente creada y registrada, y bastará con abrir la aplicación, pulsar el menú lateral, pulsar el botón de “Iniciar sesión”, y después introducir las credenciales y pulsar “Acceder”.





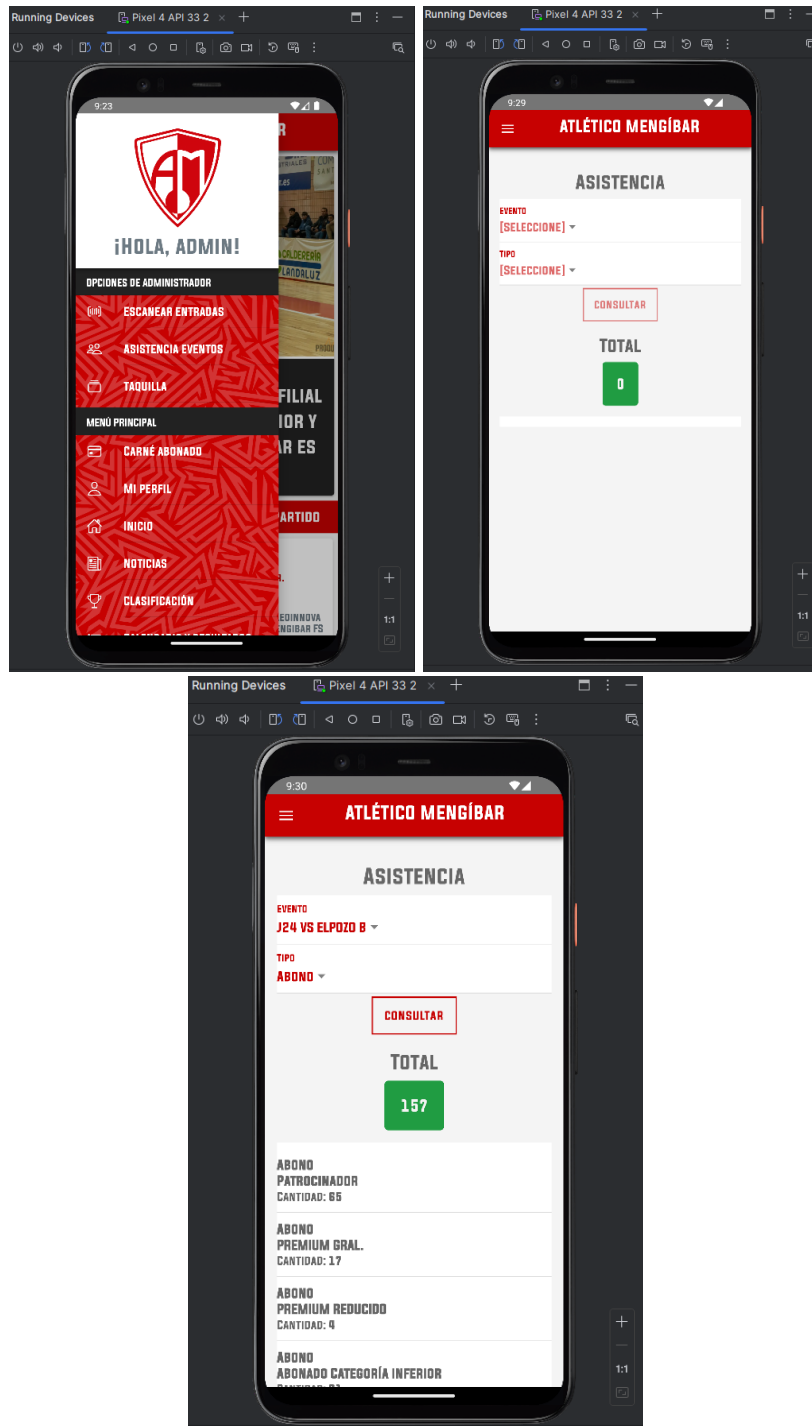
## B.2 Escanear QR

Para escanear un código QR el usuario deberá abrir la aplicación e iniciar sesión con su cuenta de administrador previamente, una vez iniciada la sesión bastará con abrir el menú lateral, pulsar en el botón de “Escanear QR”, seleccionar un evento activo para el cual escanear las entradas, y pulsar “Escanear” para proceder al escaneo de los QR.



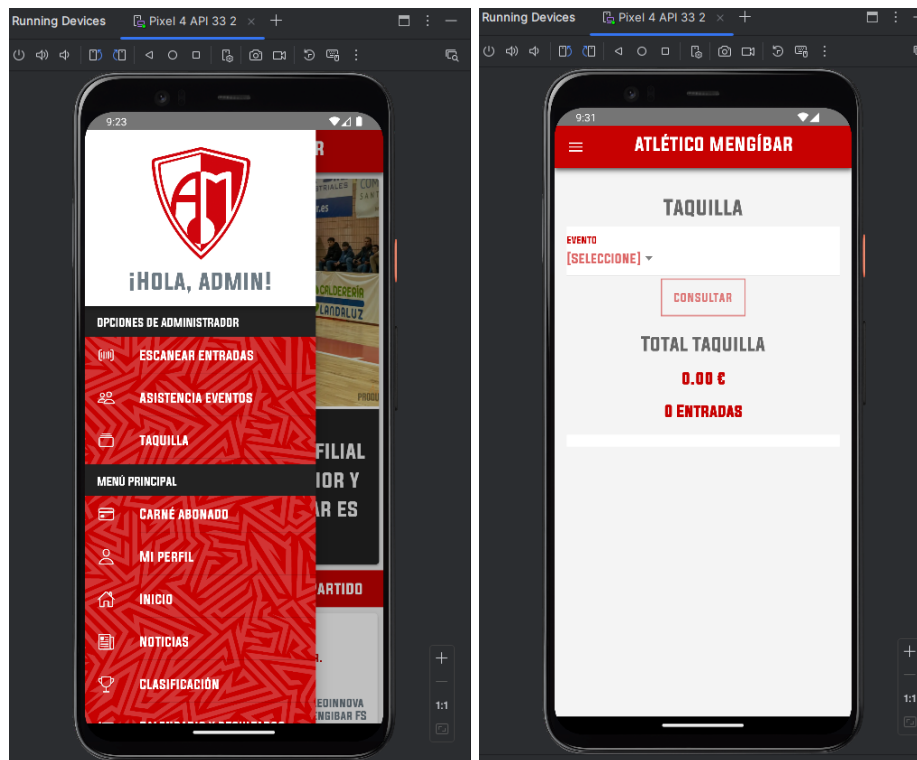
### B.3 Ver asistentes

Para revisar los números de asistencia de un evento concreto el usuario deberá abrir la aplicación e iniciar sesión con su cuenta de administrador previamente, una vez iniciada la sesión bastará con abrir el menú lateral, pulsar en el botón “Asistencia”, seleccionar un evento pasado o activo, y pulsar “Consultar” para mostrar el listado de la asistencia.



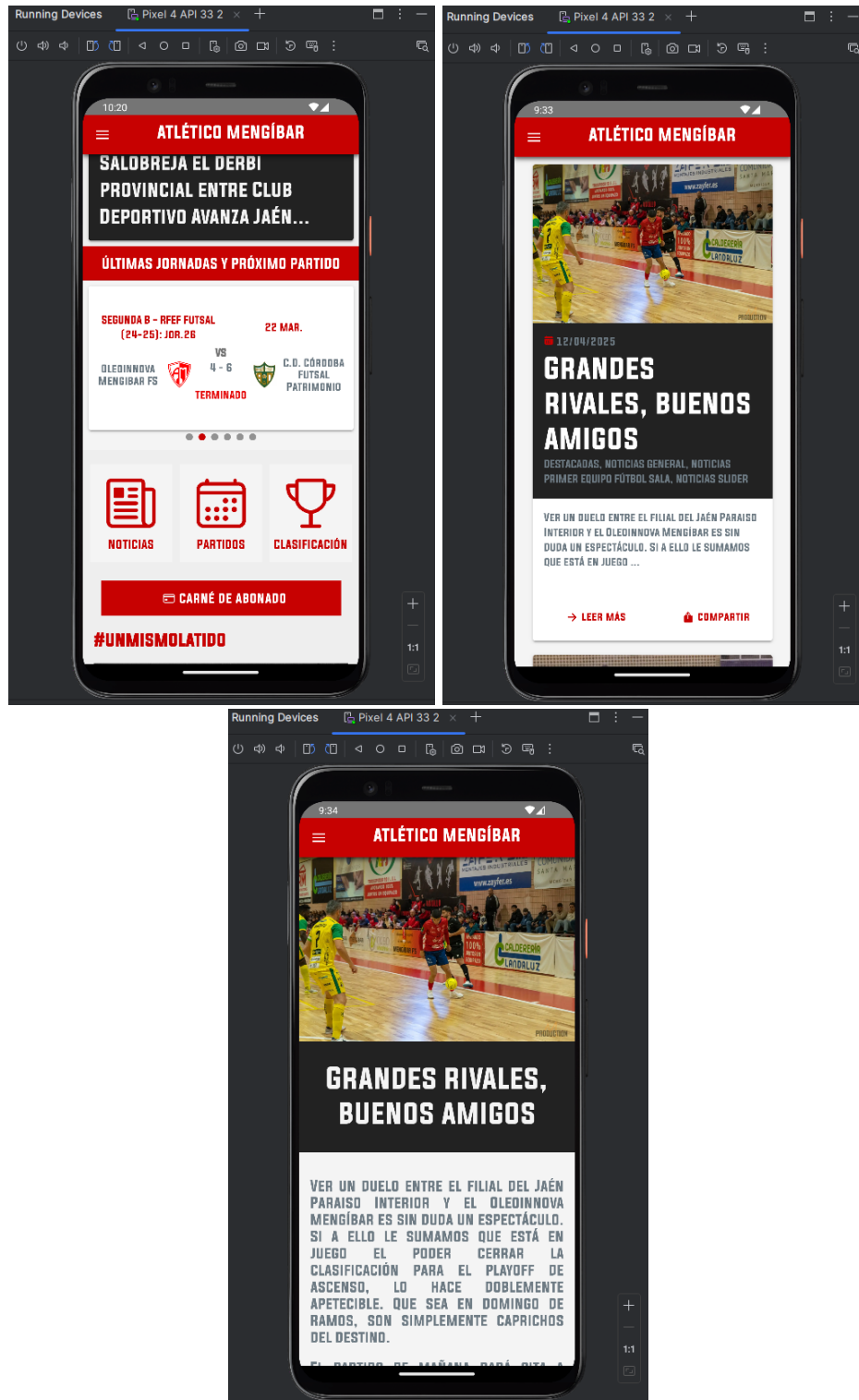
## B.4 Ver Taquilla

Para revisar los números de la taquilla de un evento concreto el usuario deberá abrir la aplicación e iniciar sesión con su cuenta de administrador previamente, una vez iniciada la sesión bastará con abrir el menú lateral, pulsar en el botón “Taquilla”, seleccionar un evento pasado o activo, y pulsar “Consultar” para mostrar el registro de la taquilla.



## B.5 Ver o compartir noticias

Para poder ver y compartir noticias, el usuario deberá abrir la aplicación, abrir el menú lateral y pulsar el botón “Noticias”, o pulsar el apartado de noticias en la página de inicio, una vez mostradas bastará con pulsar el botón de “ver detalles” o “compartir” según prefiramos.



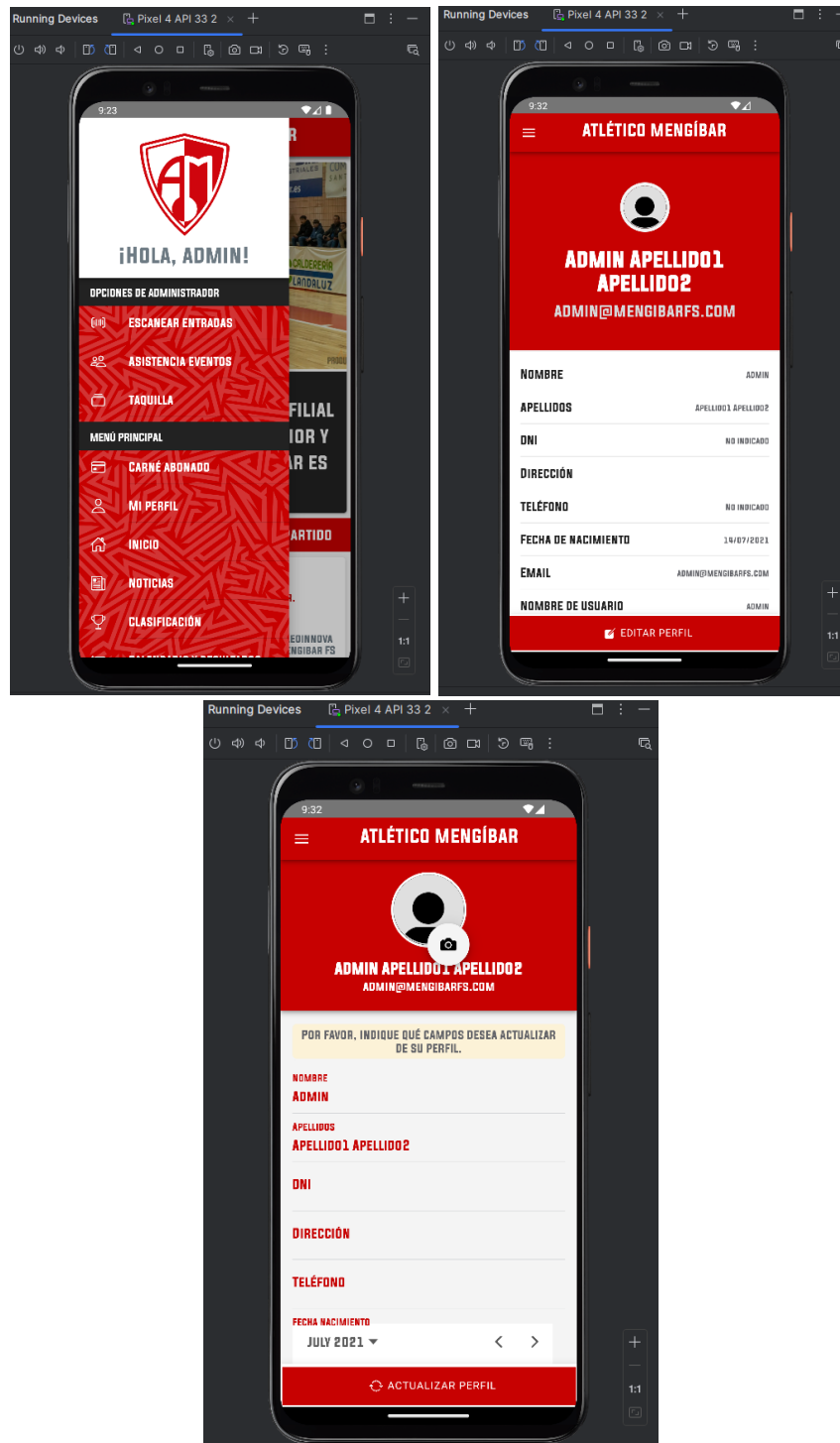
## B.6 Ver carné de abonado

Para poder ver su carné de abonado o tickets, el usuario deberá abrir la aplicación, iniciar sesión con su cuenta de usuario, abrir el menú lateral y pulsar el botón “carné abonado” y se mostrarán los diferentes carnés asociados al usuario.



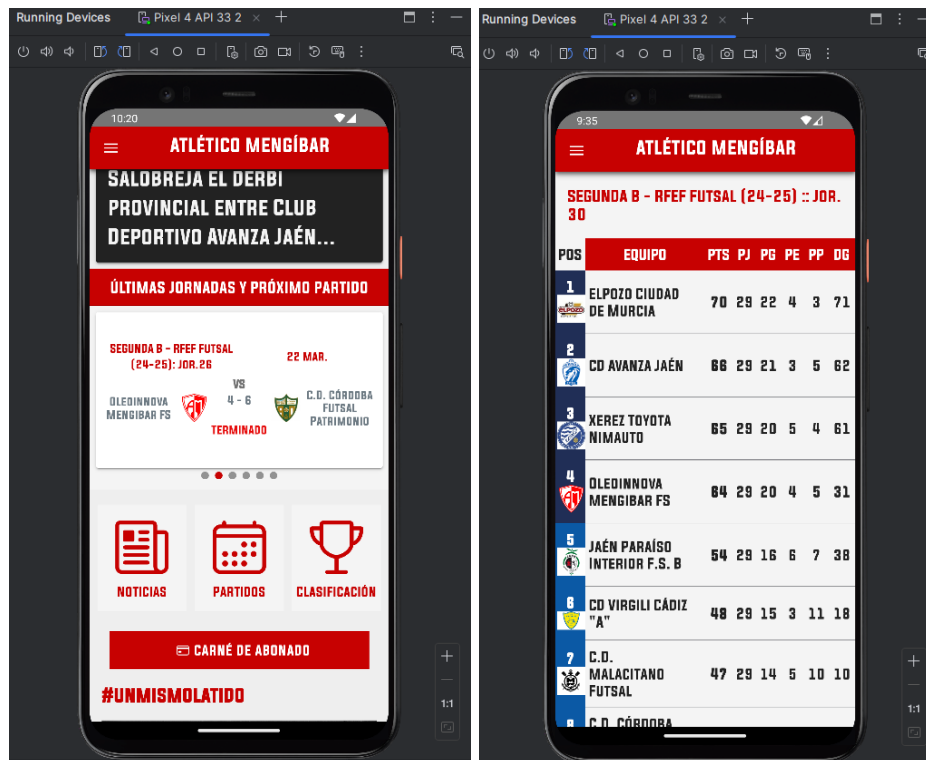
## B.7 Ver o editar perfil de usuario

Para poder revisar el perfil de usuario, el usuario deberá abrir la aplicación, iniciar sesión con su cuenta de usuario, abrir el menú lateral y pulsar el botón “Mi perfil”, así se mostrarán los datos del perfil del usuario, si quisiera editar dichos datos, bastaría con pulsar el botón de “editar perfil” en la vista de los datos, y modificar los datos que quiera cambiar.



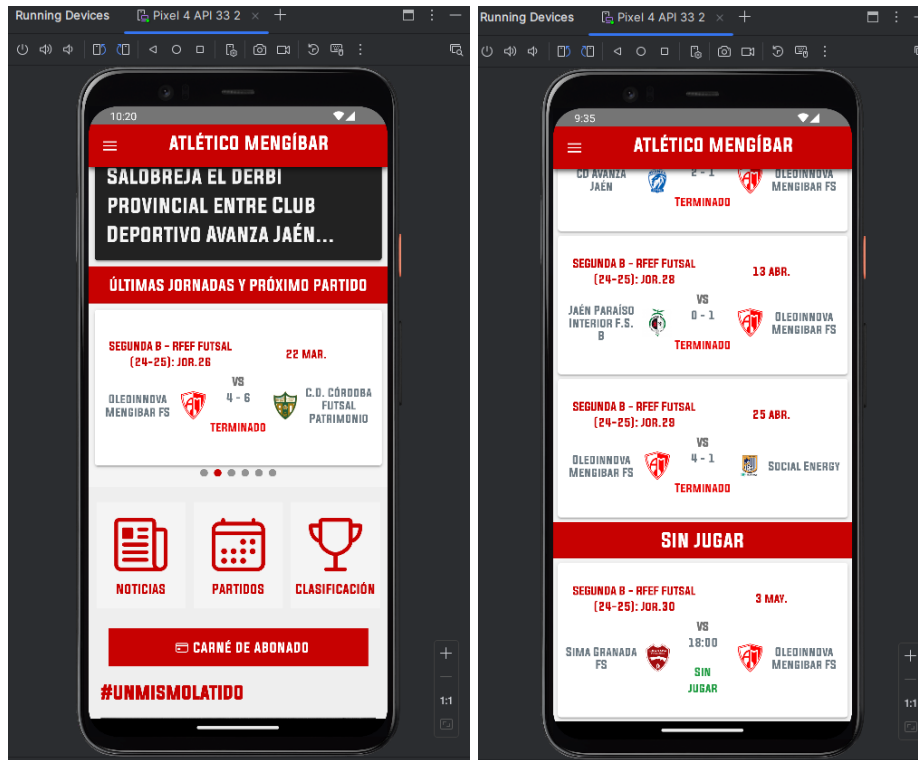
## B.8 Ver clasificación

Para poder ver la clasificación, el usuario deberá abrir la aplicación, abrir el menú lateral y pulsar el botón “Clasificación”, o bastaría con pulsar en el apartado “Clasificación” desde la página de inicio.



## B.9 Ver calendario y resultados

Para poder ver el calendario, el usuario deberá abrir la aplicación, abrir el menú lateral y pulsar el botón “Calendario y resultados”, o bastaría con pulsar en el apartado “Partidos” desde la página de inicio. Además, también se muestran los 5 últimos partidos en la página de inicio.





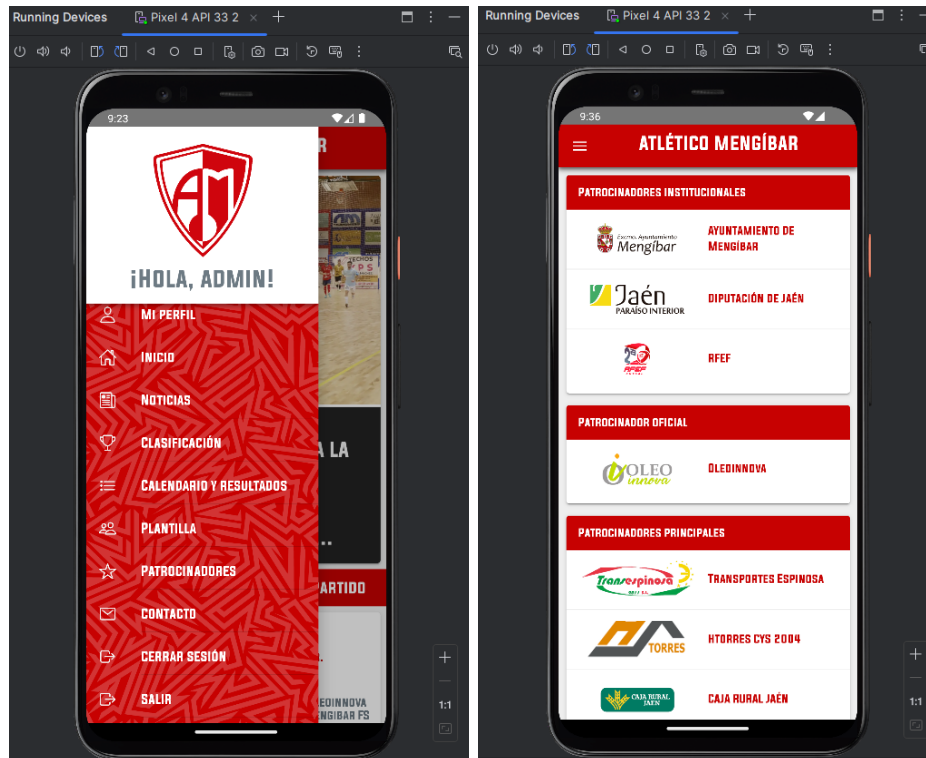
## B.10 Ver plantilla

Para poder ver la plantilla activa del club, el usuario deberá abrir la aplicación, abrir el menú lateral y pulsar el botón “Plantilla” para mostrar el listado de la plantilla actual del club.



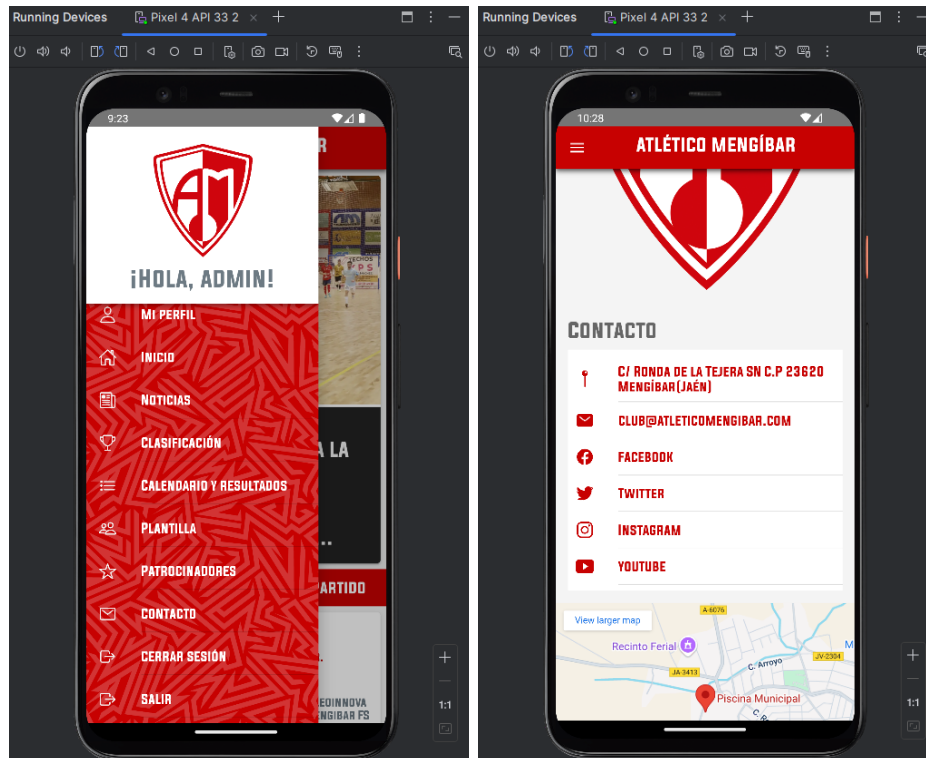
## B.11 Ver patrocinadores

Para poder ver el listado de patrocinadores, el usuario deberá abrir la aplicación, abrir el menú lateral y pulsar el botón “Patrocinadores” para mostrar el listado de patrocinadores del club.



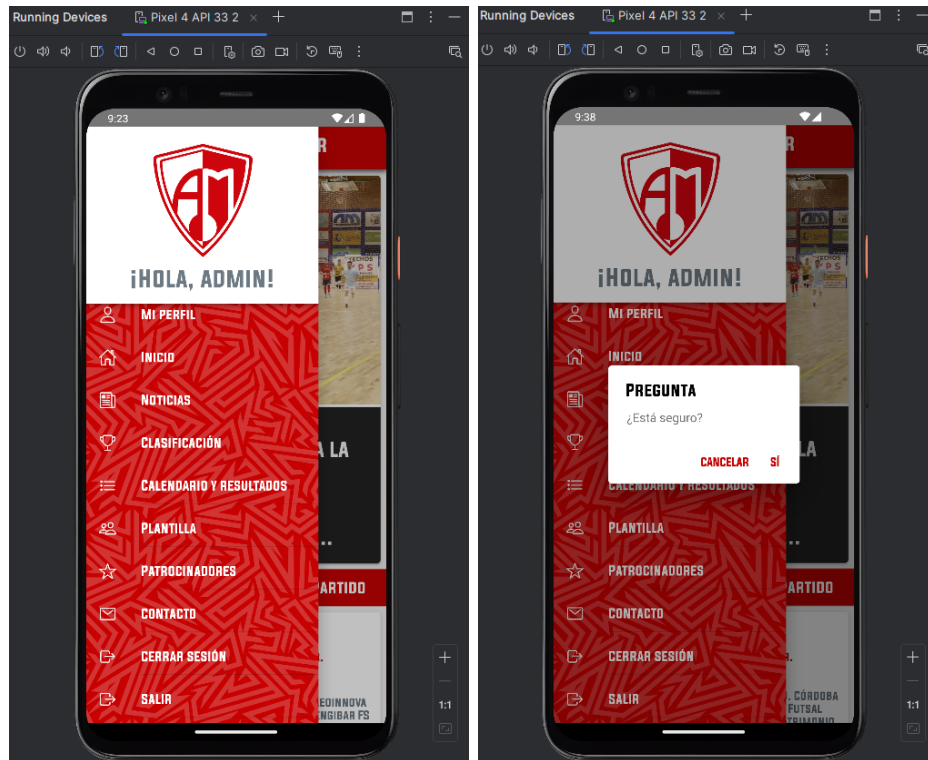
## B.12 Ver información de contacto

Para poder ver la información de contacto, el usuario deberá abrir la aplicación, abrir el menú lateral y pulsar el botón “Contacto” para mostrar el listado de la información de contacto.



### B.13 Cerrar sesión

Para cerrar sesión el usuario deberá tener la sesión iniciada previamente (de lo contrario no aparece el botón) y bastará con abrir el menú lateral y pulsar el botón “Cerrar sesión”, después pedirá confirmación y si confirmamos se cerrará la sesión en la aplicación.



## C. Documentación del proceso de subida de la Aplicación móvil a la tienda

En el siguiente anexo vamos a documentar el proceso seguido para la publicación de la aplicación en la *Google Play Store* utilizando *Google Play Console*. Describiremos las etapas más clave, desde la firma del AAB, la configuración en Play Console, la fase de pruebas y la publicación final. Por decisión de la empresa MGS INGENIERÍA INFORMÁTICA, la aplicación de momento no será subida a la Apple Store.

En primer lugar, para subir una aplicación a la *Google Play Store* debemos tener una cuenta de desarrollador. En este caso la empresa ya tenía una cuenta y lo único necesario ha sido invitar al correo de la cuenta con la que vamos a manejar la subida. Cabe destacar que este proceso no puede hacerse con una cuenta como la de la universidad ([itg00023@red.ujaen.es](mailto:itg00023@red.ujaen.es)) ya que al ser de una institución tiene restricciones y Google no lo permite, así que tuve que hacerlo con mi cuenta personal de Gmail. En este caso ha sido utilizada la cuenta de la empresa MGS INGENIERÍA INFORMÁTICA S.L.

Una vez tengamos los permisos y entremos en la cuenta de desarrollador, nos aparecerá una vista de las aplicaciones a las que podemos acceder (en función de los permisos dados), y a partir de ahí, podremos entrar en el perfil de la aplicación.

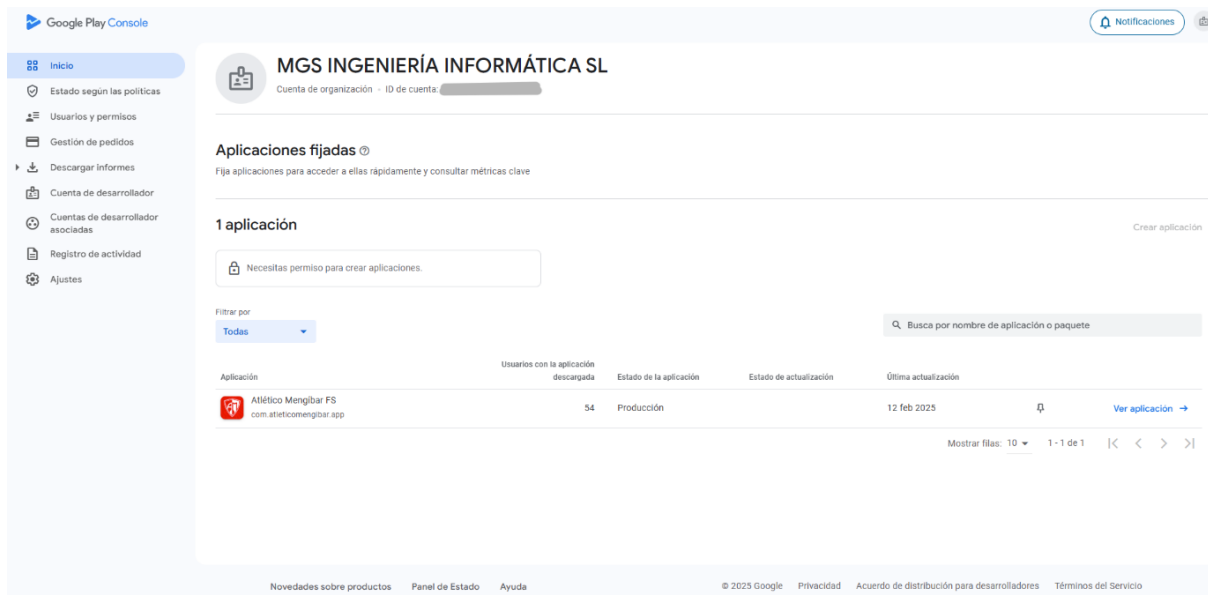


Figura Google Play Console 98

Para poder continuar con el proceso de subida de la aplicación, más adelante será necesario subir el *App Bundle* de la aplicación, el cual, primero hay que generar

desde Android Studio firmado por la clave de la propia aplicación. El procedimiento para generar el *App Bundle* firmado es el siguiente:

- Primero, necesitamos el archivo *keystore* de la aplicación, el cual solo se puede generar una vez y es necesario guardarlo bien porque sin él no se puede actualizar la misma aplicación, ya que, con una firma diferente, *Google Play Console* lo detectaría como una aplicación diferente y se subiría como tal. En este caso, nos bastó con pedirle el archivo *keystore* y las credenciales a la empresa.
- Después necesitamos que el proyecto compile correctamente y sin errores de dependencias en Android Studio, tras lo que le daremos al botón de “Generar App Bundle / APK firmada”.

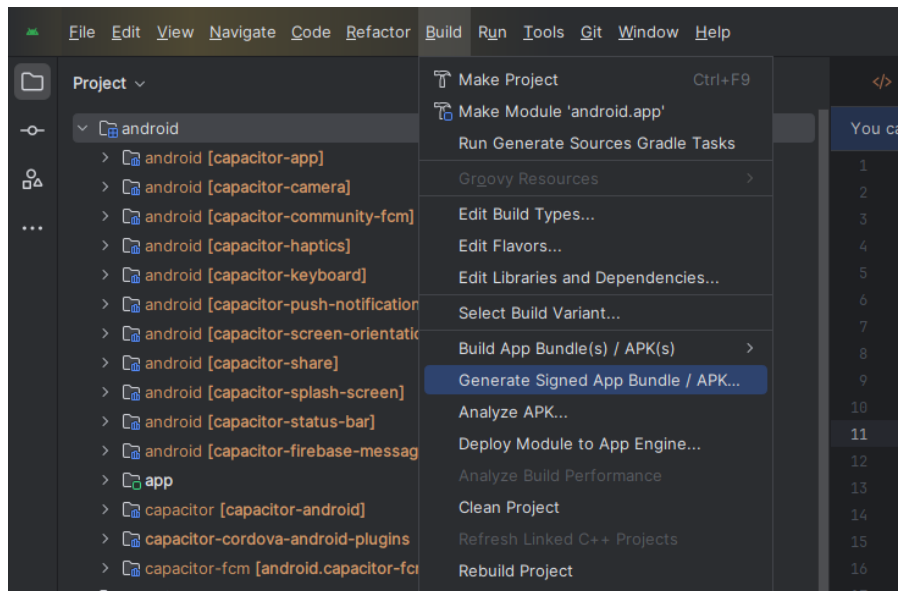


Figura de Android Studio 99

- Se nos abrirá una ventana que nos pedirá el archivo *keystore* y las credenciales del mismo.

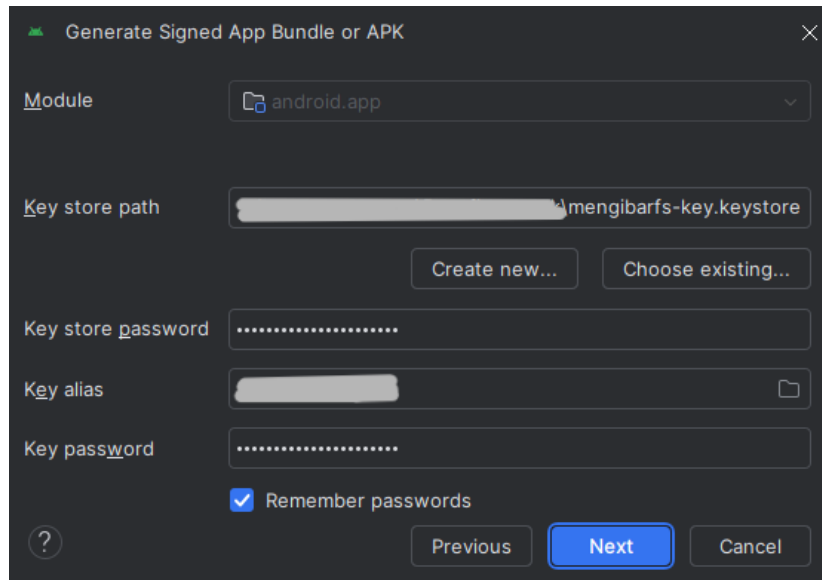


Figura de Android Studio 100

- Una vez introducidas las credenciales, se generará el archivo *App Bundle* (.aab) que será el que contendrá la aplicación, similar a un archivo *APK*.

Una vez dentro del perfil de la aplicación, lo primero que haremos es crear una nueva prueba interna, que nos permitirá subir y hacer cambios en la aplicación sin afectar a la fase de producción, lo que nos facilitará la corrección de errores antes del lanzamiento. Dentro de la prueba interna nos pedirá que creamos una nueva versión de la aplicación, para lo cual debemos hacer 2 cosas:

1. Subir el nuevo *App Bundle* (.aab) ya firmado de la aplicación.

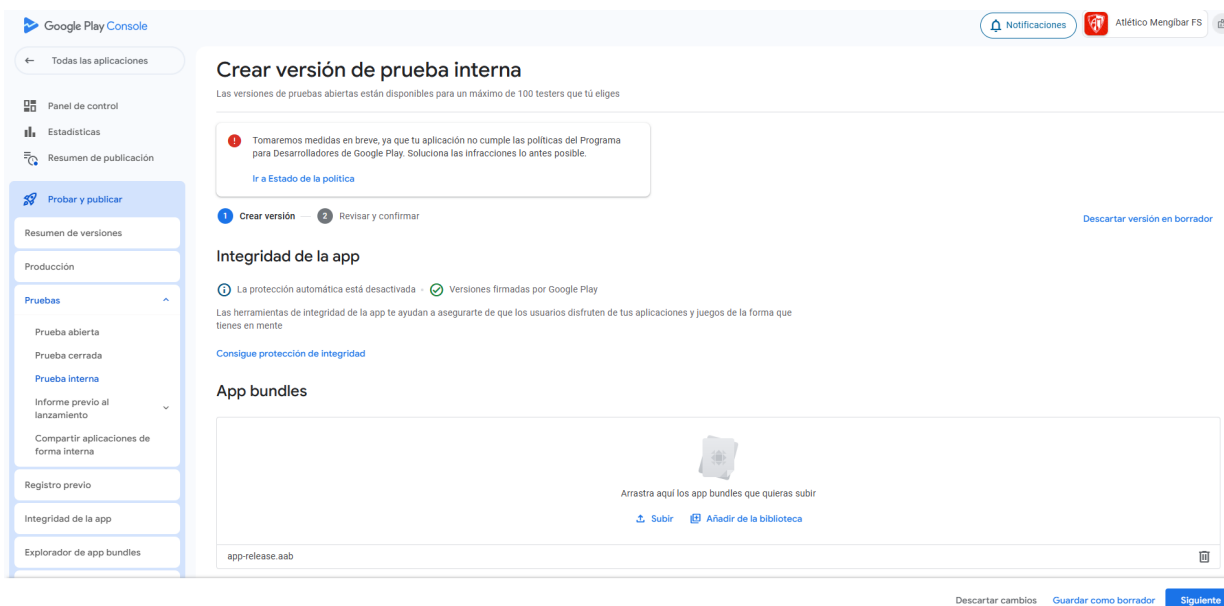
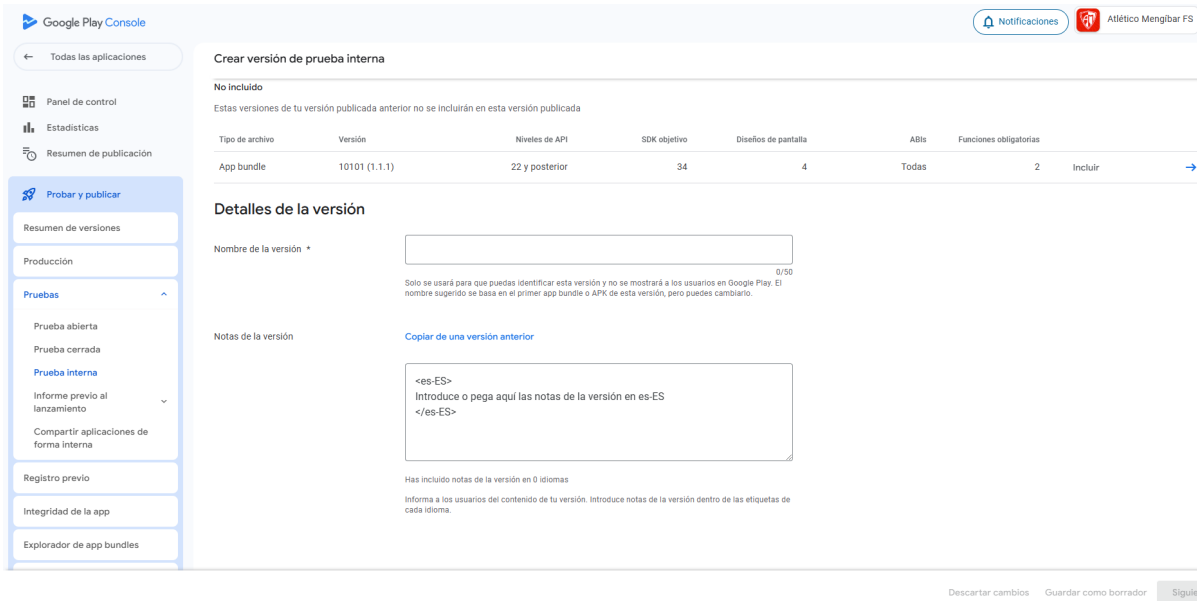


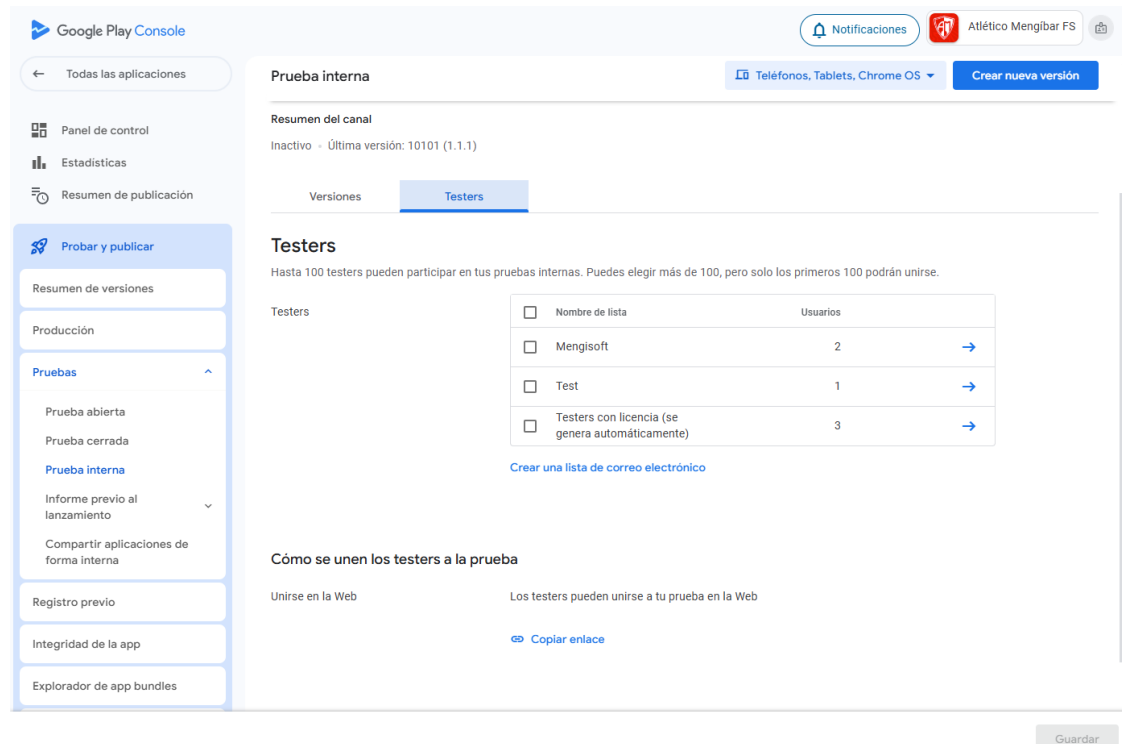
Figura de Google Play Console 101

## 2. Darle un nombre y unas notas o descripción a la nueva versión.



**Figura de Google Play Console 102**

El siguiente paso una vez subida la nueva versión de prueba interna, es asignar a los testers que van a probar esta prueba. Al ser una prueba interna, pueden participar hasta un máximo de 100 testers para probar la nueva versión de nuestra aplicación.



**Figura de Google Play Console 103**



A la hora de añadir los testers nos pedirá un correo electrónico, al que se le darán los permisos para descargar la aplicación automáticamente de la Google Play Store, con un aviso de que es una versión de prueba. Podemos introducir los correos electrónicos a mano, o si son muchos se podría subir un archivo .CSV con todos los correos. En este caso, solo hemos introducido un correo de prueba.

Google Play Console

Notificaciones Atlético Mengibar FS

Todas las aplicaciones Prueba interna Telefonos, Tablets, Chrome OS Crear nueva versión

### Editar lista de correo electrónico

\* Campos obligatorios

Nombre de lista \* Test 4/200

Añadir direcciones de correo electrónico user@example.com, user@example.com

Separa las direcciones de correo con una coma y pulsa Intro para añadirías a tu lista

[Subir archivo CSV](#)

Direcciones de correo electrónico añadidas \*

|                      |  |
|----------------------|--|
| jatoga01@gmail.com   |  |
| jtg00023@red.ujen.es |  |

Eliminar lista de correo electrónico Descartar **Guardar cambios**

**Figura de Google Play Console 104**

Una vez creada y configurada la versión, nos muestra una serie de detalles acerca de ella, como el tamaño de descarga que tendrá, el tiempo que llevará dicha descarga, o incluso una lista del catálogo completo de los dispositivos que son compatibles con ella.

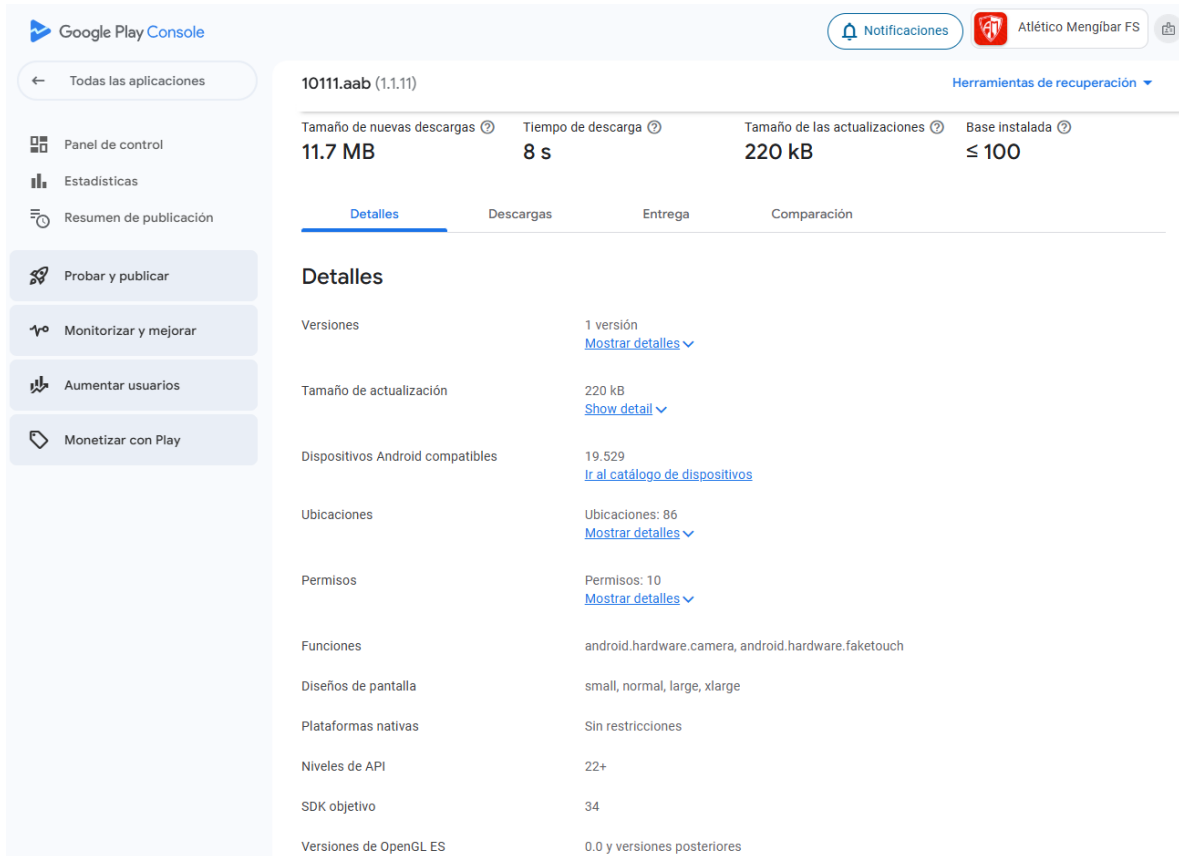


Figura de Google Play Console 105

Posteriormente a la publicación de la versión, Google nos detalla un informe previo al lanzamiento final, en el que nos muestra si hay problemas de estabilidad, rendimiento o accesibilidad. En nuestro caso, ya había problemas de estabilidad antes de la subida de la nueva versión, en concreto problemas con las versiones de Android, ya que la versión anterior estaba creada con una versión de Ionic muy antigua y obsoleta. Estos problemas provocaban que la aplicación no se pudiese descargar en dispositivos con las últimas versiones de Android (Android 13 y Android 14), y apareciese un error en la ficha de Play Store. Pero con la subida de la nueva versión y la revisión de las políticas, estos errores se solucionaron.

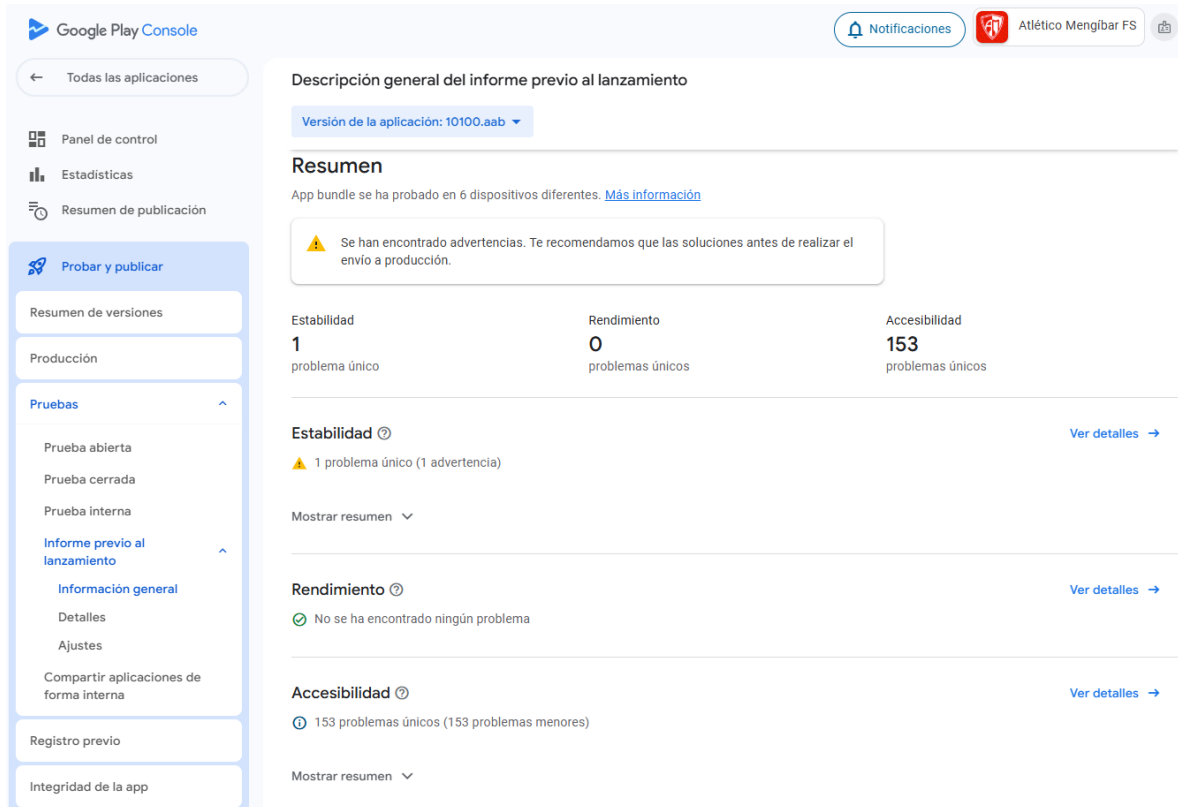
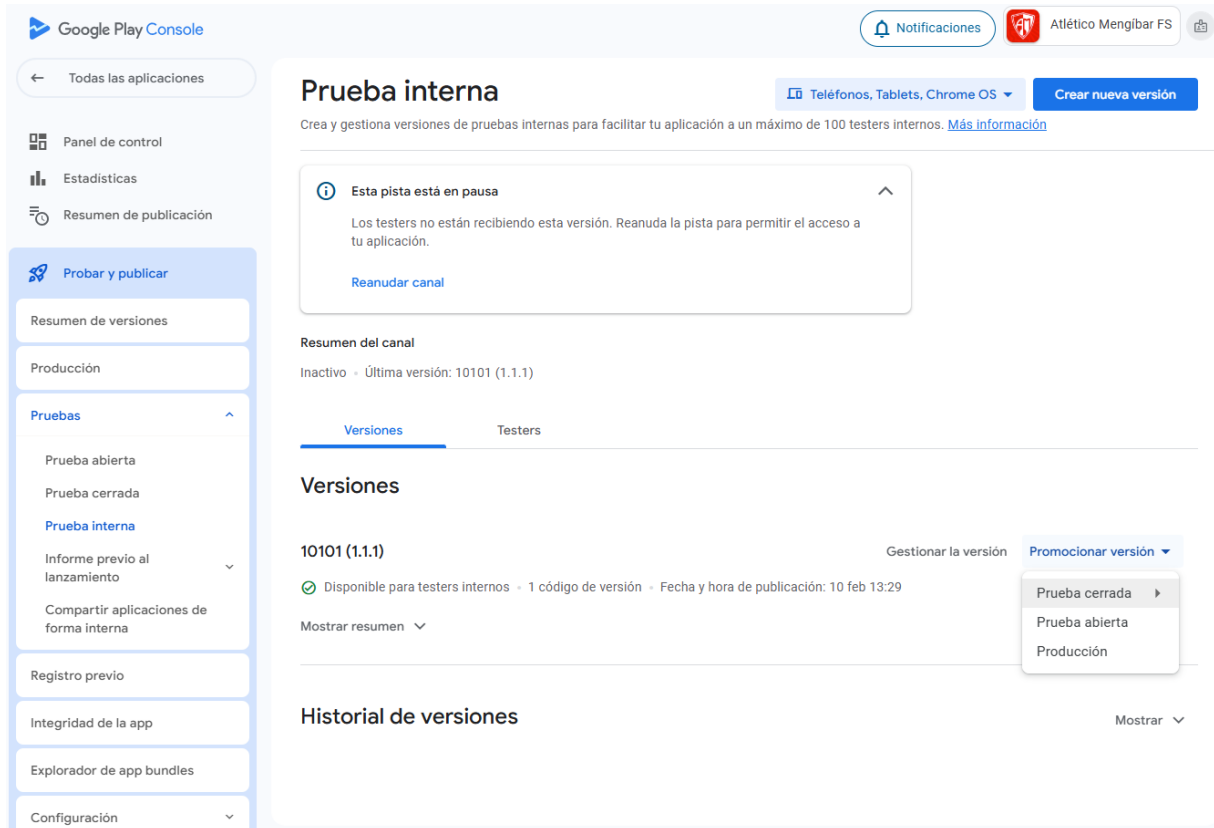


Figura de Google Play Console 106



Figura de Google Play Console 107

Una vez hayamos realizado todas las pruebas pertinentes con la nueva versión, y tengamos el visto bueno del cliente, procederemos a promocionarla a la fase de producción, también podría promocionarse a la fase de prueba cerrada o prueba abierta, que son otro tipo de pruebas con diferentes características, pero en esta ocasión no será necesario.



**Figura de Google Play Console 108**

Y una vez tengamos la nueva versión en la fase de producción ya solo quedaría esperar a que *Google Play Console* le dé el visto bueno, y se actualice la ficha de la aplicación en Play Store.

A partir de aquí, podremos monitorear las estadísticas de la nueva actualización como cuanta gente tiene la nueva versión o la versión antigua, o las reseñas y valoraciones nuevas.

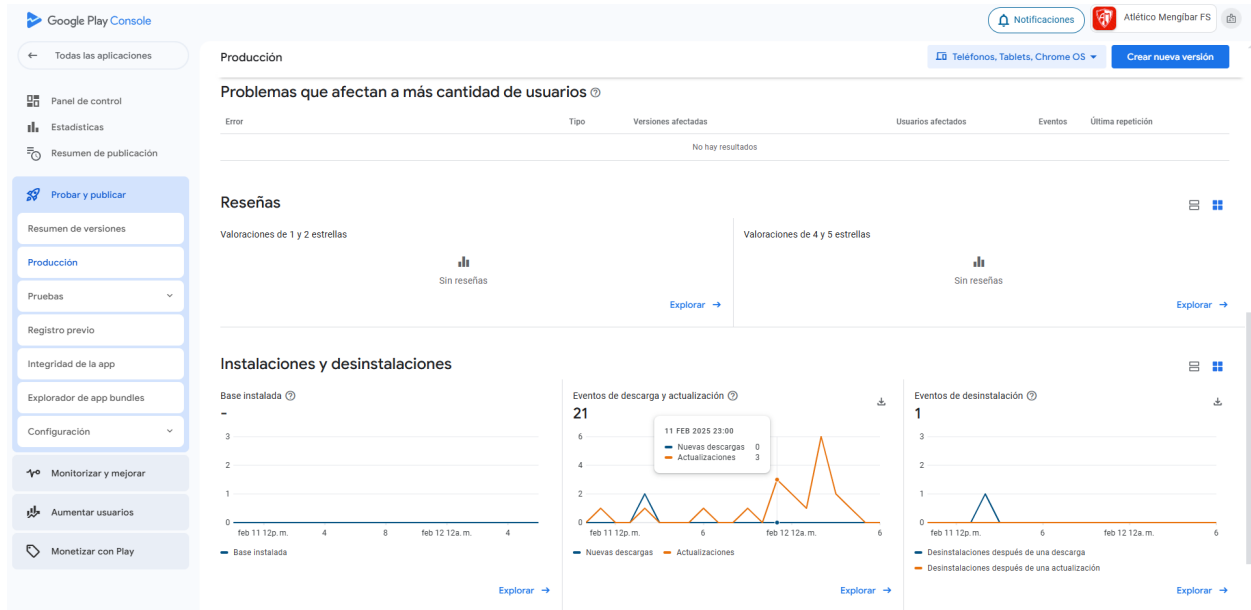


Figura de Google Play Console 109

## Bibliografía

- [1] R.S Pressman. *Ingeniería del Software, un enfoque práctico*. Mc Graw Hill 7ª Edición, 2010
- [2] Bennett, Simon. *Análisis y diseño orientado a objetos de sistemas: usando UML*. Edición: 3ª ed. Editorial: Madrid: Mac Graw-Hill Interamericana, 2007
- [3] Ian Sommerville. *Software engineering*. Edición: 10th ed. Editorial: Addison Wesley
- [4] F.Feito, Andrés Molina y J. Ruiz de Miras. *Análisis y Gestión de Datos*. Universidad de Jaén 1996.
- [5] MÉTRICA 3. *Ingeniería del Software para administraciones públicas. Ministerio para la transformación digital*.  
[https://administracionelectronica.gob.es/pae/Home/pae/Documentacion/pae/Metodolog/pae/Metrica\\_v3.html](https://administracionelectronica.gob.es/pae/Home/pae/Documentacion/pae/Metodolog/pae/Metrica_v3.html)
- [6] *Guía de diseño de material de Google*. <https://m3.material.io/>
- [7] *Guía de interfaz humana de Apple*. <https://developer.apple.com/design/human-interface-guidelines>
- [8] Symfony. <https://symfony.com/>
- [9] Ionic. <https://ionicframework.com/>
- [10] PhpStorm. <https://www.jetbrains.com/phpstorm/>
- [11] AndroidStudio. <https://developer.android.com/studio>
- [12] XCode. <https://developer.apple.com/xcode/>
- [13] BitBucket. <https://bitbucket.org/product/>
- [14] Yves Rocamora, David Guillén Jiménez. *Desarrollar aplicaciones web con PHP y Symfony*. ISBN 2409046002. Ediciones Eni. 2024
- [15] Josh Goldberg. *Learning TypeScript*. ISBN 1098113140. O'Reilly Media. 2020.
- [16] Steve Fenton. *Pro TypeScript: Application-Scale JavaScript Development*. ISBN 1484206485. Apress. 2016.
- [17] Vinicius M Grippa, Sergey Kuzmichev. *Learning MySQL: Get a Handle on Your Data*. O'Reilly Media. 2021
- [18] *Nginx Server Documentación oficial*. <https://nginx.org/>
- [19] Sonarqube. Herramienta para análisis de código estático  
<https://www.sonarsource.com/products/sonarqube/>

[20] <https://miro.com/es/storyboard/que-es-storyboard/>

[21] <https://miro.com/es/wireframe/que-es-wireframe/>

[22] Leff, Avraham; James T. Rayfield (September 2001). *Web-Application Development Using the Model/View/Controller Design Pattern*. IEEE Enterprise Distributed Object Computing Conference. pp. 118-127.