

DESARROLLO DE UNA APLICACIÓN WEB PARA LA GESTIÓN DE ACTIVIDADES EN CANCHAS DE FRONTÓN



TRABAJO FIN DE MÁSTER
CURSO 2021-2022

AUTOR
ALICIA ESTHER JOY RAMIREZ

DIRECTOR
ANTONIO SARASA CABEZUELO

MÁSTER EN INGENIERÍA INFORMÁTICA
FACULTAD DE INFORMÁTICA
UNIVERSIDAD COMPLUTENSE DE MADRID

DESARROLLO DE UNA APLICACIÓN WEB PARA LA GESTIÓN DE ACTIVIDADES EN CANCHAS DE FRONTÓN

TRABAJO DE FIN DE MÁSTER EN INGENIERÍA INFORMÁTICA
DEPARTAMENTO DE INFORMATICA

AUTOR
ALICIA ESTHER JOY RAMIREZ

DIRECTOR
ANTONIO SARASA CABEZUELO

CONVOCATORIA: SEPTIEMBRE 2022
CALIFICACIÓN: 5.5

MÁSTER EN INGENIERÍA INFORMÁTICA
FACULTAD DE INFORMÁTICA
UNIVERSIDAD COMPLUTENSE DE MADRID

23 DE SEPTIEMBRE DE 2022

DEDICATORIA

*A Dios por su ayuda en todo tiempo, a mis
padres y hermana por su motivación*

AGRADECIMIENTOS

Quiero empezar dando las gracias a Dios en ayudarme a ser mejor persona, brindándome las fuerzas en momentos de tristeza logrando ser mi refugio y mi paz.

Gracias infinitas a mis padres por su educación y enseñarme a nunca rendirme en dificultades, por su amor y su paciencia en cada momento de mi vida, ellos siempre estuvieron para levantarme, a mi hermana y mi cuñado por su apoyo en lograr mis estudios acoguéndome en su hogar con mucho cariño.

A mis compañeros que me apoyaron desinteresadamente y brindarme su tiempo siempre que lo necesite. A los compañeros invisibles de los pasillos de la Universidad, fueron personas que no me conocían, pero brindaron su apoyo.

A mis profesores por su motivación a esforzarme y sacar lo mejor de mí en cada materia, a mis profesores que no lo ponían muy fácil ya que me hicieron más luchadora y sacar el máximo de mí para lograr llegar a mi meta.

Seguidamente, quiero agradecer a mi asesor Antonio Sarasa Cabezuelo por sus consejos y su ayuda en la realización de este proyecto, gracias por su paciencia, tiempo desinteresado y sus consejos en cada etapa de la realización de este proyecto.

Quiero agradecer a la Universidad Complutense de Madrid, el recibimiento que tuve en sus instalaciones y brindarme las mejores experiencias de mi vida al cursar mis estudios durante estos casi tres años. Tienen las mejores instalaciones y aunque sufrimos una pandemia casi dos años, ellos nunca descansaron y se esforzaron para darme la mejor educación.

RESUMEN

DESARROLLO DE UNA APLICACIÓN WEB PARA LA GESTIÓN DE ACTIVIDADES EN CANCHAS DE FRONTÓN

En este trabajo se presenta el diseño y desarrollo de una aplicación web que tiene como objetivo ayudar a mejorar la gestión de actividades en canchas de frontón. En este sentido, se ha implementado una aplicación web que proporciona un buscador de canchas e inscripciones a torneos, un sistema de reserva de canchas, y un asistente virtual que ayuda interaccionar con la aplicación. Así mismo, la aplicación permite acceder a gráficas que muestran el desempeño de los usuarios que han participado en algún torneo.

El proyecto puede descargarse en el siguiente enlace: <https://github.com/Alice-ucm/TFM>

Palabras clave

Aplicación web, Php, MYSQL, Google Charts API, Chatbot

ABSTRACT

DEVELOPMENT OF A WEB APPLICATION FOR THE MANAGEMENT OF ACTIVITIES IN FRONTÓN COURTS

This paper presents the design and development of a web application that aims to help improve the management of activities in fronton courts. In this sense, a web application has been implemented that provides a field and tournament registration search engine, a field reservation system, and a virtual assistant that helps interact with the application. Likewise, the application allows access to graphs that show the performance of users who have participated in a tournament.

The project can be downloaded at the following link:

<https://github.com/Alice-ucm/TFM>

Keywords

Web application, Php, MYSQL, Google Charts API, Chatbot

ÍNDICE

Dedicatoria	IV
Agradecimientos	V
Resumen.....	VII
Abstract	IX
Índice.....	X
Índice de figuras.....	XIII
Índice de tablas.....	XVII
Capítulo 1 - Introducción.....	19
1.1 Motivación	19
1.2 Objetivos.....	20
1.3 Estructura de la memoria.....	20
1.4 Planificación.....	21
Capítulo 2 - Estado del arte.....	22
2.1 SportMember	22
2.2 SimplyBook	23
2.3 Cluber	24
2.4 Runkeeper	25
2.5 Tennis Score Trancker	26
2.6 Clupik	27
Capítulo 3 - Especificación de requisitos.....	28
3.1 Actores.....	28
3.2 Casos de uso.....	28
3.3 Especificación de casos de uso.....	31

Capítulo 4 - Tecnologías y Herramientas.....	40
4.1 PHP	40
4.2 MySQL	40
4.3 Xampp	40
4.4 Google Chart API	41
4.5 Node.Js	41
4.6 Express.js.....	41
4.7 Dialogflow	42
Capítulo 5 - Arquitectura y modelo de datos.....	43
5.1 Arquitectura	43
5.2 Modelo de Datos.....	44
5.2.1 Tabla users	44
5.2.2 Tabla cancha	45
5.2.3 Tabla reserva_cancha.....	46
5.2.4 Tabla club	46
5.2.5 Tabla torneos	46
5.2.6 Tabla inscripción.....	47
5.2.7 Tabla sport	47
5.2.8 Tabla roles.....	48
5.2.9 Tabla desempeño.....	48
Capítulo 6 - Implementación	49
6.1 Aplicación web	49
6.1.1 Página Inicio	49
6.1.2 Registro e inicio de sesión.....	50
6.1.3 Menú principal	52

6.1.4 Perfil de usuario	53
6.1.5 Desempeño	55
6.1.6 Torneos	57
6.1.7 Canchas y reservas.....	61
6.2 Chatbot	66
6.2.1 Creación de agente.....	66
6.2.2 Primera conversación.....	67
6.2.3 Selección noticias	68
6.2.4 Selección frontón	69
6.2.5 Selección cancha.....	70
6.2.6 Resultados chatbot.....	70
Capítulo 7 - Conclusiones y trabajo futuro.....	74
7.1 Conclusiones	74
7.2 Trabajo futuro	75
Chapter - 7. Introduction	77
7.1. Motivation.....	77
7.2. Objectives.....	77
Chapter - 8 Conclusions and future work.....	78
Bibliografía.....	80
Apéndices	83

ÍNDICE DE FIGURAS

Figura 1-1. Plan de trabajo.....	21
Figura 2-1. Pantalla aplicación de actividades.....	22
Figura 2-2. Pantalla aplicación SimplyBook	24
Figura 2-3. Pantalla Aplicación Cluber	25
Figura 2-4. Pantalla Aplicación Runkeenper.....	26
Figura 2-5. Pantalla Aplicación Tennis Score.....	27
Figura 2-6. Clupik	27
Figura 3-1. Caso de uso - Deportista	29
Figura 3-2. Caso de uso – Club.....	29
Figura 3-3. Caso de uso – Administrador	30
Figura 3-4. Caso de uso – Público	30
Figura 3-5. Caso de uso 1 - Registrar en el sistema	31
Figura 3-6. Caso de uso 2 - Iniciar sesión	31
Figura 3-7. Caso de uso 3 - Buscar cancha.....	32
Figura 3-8. Caso de uso 4 - Reservar Cancha.....	32
Figura 3-9. Caso de uso 5 - Ver Torneo	33
Figura 3-10. Caso de uso 6 - Inscribirse Torneo.....	33
Figura 3-11. Caso de uso 7 - Agregar Desempeño	34
Figura 3-12. Caso de uso 9 - Editar Perfil	34
Figura 3-13. Caso de uso 10 - Dar baja.....	35
Figura 3-14. Caso de uso 11 - Cerrar sesión.....	35
Figura 3-15. Caso de uso 12 - Agregar Cancha	36
Figura 3-16. Caso de uso 13 - Eliminar cancha	36

Figura 3-17. Caso de uso 14 - Editar cancha	37
Figura 3-18. Caso de uso 15 - Agregar torneo	37
Figura 3-19. Caso de uso 19 - Eliminar torneo	38
Figura 3-20. Caso de uso 17 - Editar torneo.....	38
Figura 3-21. Caso de uso 18 - Gestionar Cancha.....	39
Figura 3-22. Caso de uso 19 - Gestionar torneos	39
Figura 3-23. Caso de uso 20 - Gestionar usuarios.....	39
Figura 4-1. Ejemplo flujo Dialogflow	42
Figura 5-1. Arquitectura del proyecto	43
Figura 5-2. Tablas de la base de datos.....	44
Figura 6-1. Página de inicio	49
Figura 6-2 Código página inicio.....	50
Figura 6-3. Pantalla – Registro deportista	50
Figura 6-4. Registrar en el sistema	51
Figura 6-5 Pantalla - Inicio de sesión	51
Figura 6-6. Código – Inicio de sesión.....	52
Figura 6-7. Menú principal deportista	52
Figura 6-8. Código - Menú principal	53
Figura 6-9. Perfil deportista.....	53
Figura 6-10. Código - Perfil deportista.....	54
Figura 6-11. Perfil club.....	54
Figura 6-12. Código - Dar de baja	54
Figura 6-13. Desempeño deportista.....	55
Figura 6-14. Agregar Desempeño.....	55
Figura 6-15 Gráfico desempeño	56

Figura 6-16. Código - Google Chart	56
Figura 6-17. Torneos deportistas	57
Figura 6-18 Código - Ver torneo.....	57
Figura 6-19. Ver torneo	58
Figura 6-20 Código - Ver torneo.....	58
Figura 6-21. Inscripción torneo.....	59
Figura 6-22. Código - Inscripción torneo	59
Figura 6-23. Agregar Torneo club.....	60
Figura 6-24. Código - Agregar torneo club	60
Figura 6-25. Buscador cancha – Deportista	61
Figura 6-26 Código - Tabla cancha.....	61
Figura 6-27. Lista Reservas	62
Figura 6-28 Código - Lista reserva	62
Figura 6-29. Ver Cancha	63
Figura 6-30 Código - Ver cancha	63
Figura 6-31. Reserva cancha	64
Figura 6-32. Reserva cancha	64
Figura 6-33. Agregar Cancha – Club.....	65
Figura 6-34 Código - Agregar cancha	65
Figura 6-35. Creando agente.....	66
Figura 6-36. Lista de Intents.....	67
Figura 6-37. Primera conversación.....	68
Figura 6-38. Selección de noticias.....	68
Figura 6-39. Selección frontón	69
Figura 6-40. Selección cancha.....	70

Figura 6-41. Asistente virtual – chartbot	71
Figura 6-42. Chatbot.....	71
Figura 6-43. Chatbot – Noticias	72
Figura 6-44. Chatbot - Canchas.....	73

ÍNDICE DE TABLAS

Tabla 6-1. Frases de entrenamiento.....	67
Tabla 6-2. Parámetro identificador	67

Capítulo 1 - Introducción

En este capítulo se va a describir la problemática que motiva la realización de este proyecto. Así mismo, se describirán los objetivos planteados, la estructura de la memoria y la planificación seguida para realizar el proyecto.

1.1 Motivación

La gestión de las canchas deportivas de frontón es complicada dado que en general no existen aplicaciones específicas orientadas al usuario final, donde éste pueda buscar una cancha, gestionar la reserva de una, o realizar otras operaciones tales como apuntarse a torneos, estar informado de eventos o actividades deportivas. En este sentido, esta situación supone una limitación para los usuarios aficionados al frontón. De la misma forma, constituye una limitación para los centros deportivos y las instituciones que organizan torneos al no disponer de un canal de comunicación con los aficionados y un mecanismo de gestión de la información.

Por estas razones, sería interesante disponer de una herramienta que integrará este tipo de servicios y facilitará las gestiones y acceso a las instalaciones y a los eventos deportivos del ámbito del frontón. Una aplicación que resolviera estas limitaciones debería tener al menos dos tipos de usuarios (deportista, club) y ofrecer funciones tales como agregar el desempeño, gestión de resultados, gestión de inscripciones a torneos, reserva de canchas y la gestión del perfil de usuario.

En esta memoria se describe el desarrollo de una aplicación que implementa un sistema para la gestión de los servicios que ofrecen los centros deportivos a los aficionados al frontón. Concretamente, la aplicación facilita funciones para la reserva de canchas, inscripción a torneos, gestión de actividades, gestión de resultados y la búsqueda de información.

1.2 Objetivos

El principal objetivo de este proyecto es diseñar e implementar una aplicación web que permita gestionar las actividades que se realizan en las canchas de frontón. Este objetivo se particulariza en los siguientes objetivos más específicos:

- Proporcionar un buscador para el público general y para los usuarios inscritos.
- Ofrecer al usuario la opción de poder registrarse.
- Proporcionar al usuario un listado de torneos.
- Ofrecer la posibilidad de hacer reservas de canchas.
- Disponer de un registro de los resultados de su participación en los torneos deportivos.
- Mostrar gráficamente el desempeño del deportista.
- Disponer de un asistente virtual que ayude al usuario en su búsqueda de información.

1.3 Estructura de la memoria.

La memoria de este proyecto se estructura de la siguiente manera:

- Capítulo 1. Introducción: En este capítulo se describe la motivación y objetivos del proyecto, así como la planificación del desarrollo y la estructura de la memoria.
- Capítulo 2. Estado del arte: En este capítulo se describen algunas aplicaciones similares a la herramienta desarrollada.
- Capítulo 3. Especificación de requisitos: En este capítulo se describen los requisitos funcionales.
- Capítulo 4. Tecnologías y Herramientas: En este capítulo se describen las principales tecnologías que se utilizaron en la implementación.
- Capítulo 5. Arquitectura y modelo de datos: En este capítulo se describe la arquitectura de la aplicación y el diseño de la base de datos.

- Capítulo 6. Implementación: En este capítulo se describe la implementación de la aplicación.
- Capítulo 7. Conclusiones y trabajo futuro: En este capítulo se describen las principales conclusiones obtenidas al finalizar el proyecto y un conjunto de líneas de trabajo futuro.

1.4 Planificación

Para llevar a cabo este proyecto se realizó una planificación basada en un conjunto de tareas agrupadas en 5 fases realizadas secuencialmente:

- Análisis de requisitos: En esta fase se definieron los requisitos funcionales, los actores principales, los casos de uso y el diagrama de la aplicación.
- Diseño: En esta fase se diseñó la arquitectura del sistema y la funcionalidad necesaria de acuerdo con los requisitos que fueron obtenidos en la fase anterior.
- Implementación: En esta fase se implementó la funcionalidad de la aplicación.
- Pruebas: En esta fase se realizaron pruebas funcionales para detectar errores en la aplicación, los cuales fueron corregidos.
- Realización de memoria: En esta fase se elaboró la memoria del proyecto.

En la figura 1-1. se muestra la distribución temporal de la ejecución de las fases del proyecto llevadas a cabo.

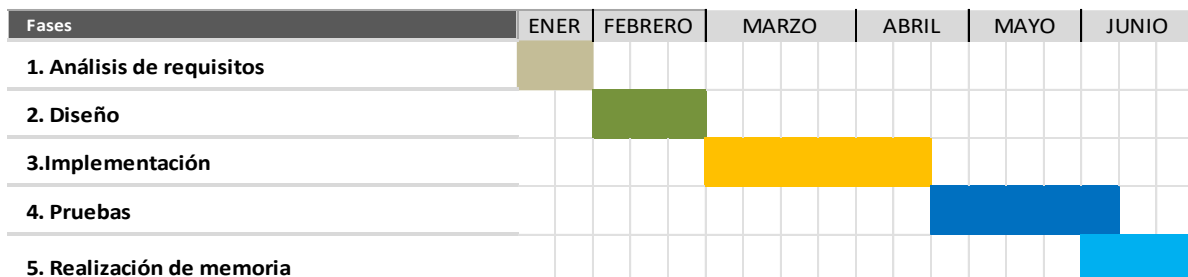


Figura 1-1. Plan de trabajo

Capítulo 2 - Estado del arte

En este capítulo se estudia y analiza el estado del arte en relación con el proyecto.

2.1 SportMember

Es una herramienta para facilitar el trabajo diario en clubes o centros deportivos que puede adaptarse a cualquier tipo de deporte: Rugby, baloncesto, fútbol, gimnasios entre otros. En particular, ayuda a organizar actividades, a mantener una base de datos actualizada de los deportistas, a gestionar los pagos, a gestionar las reservas sus instalaciones y a elaborar estadísticas de sus deportistas. [1]

En la figura 2-1 se puede apreciar una captura de la interfaz principal de la aplicación.



Figura 2-1. Pantalla aplicación de actividades

Además, esta plataforma cuenta con distintas herramientas integradas para facilitar la coordinación y comunicación con los usuarios de la aplicación.

Las principales funcionalidades de la aplicación son:

- Calendario inteligente.
- Piloto automático.
- Chat en vivo.
- Foros y enlaces.
- Galería de fotos.
- Suscripciones automáticas.
- Facturaciones.
- Gestión de deportistas.
- Noticias de equipos.

La plataforma cuenta con dos versiones de pago con distintas funcionalidades.

2.2 SimplyBook

Es un sistema de reservas orientado hacia empresas que quieren digitalizar la programación de sus servicios. Así mismo, cuenta con distintas funcionalidades e integraciones. [2]

En particular permite:

- Calendario sincronizado.
- Reservas automáticas.
- Integración con redes sociales tales como Facebook y Twitter.
- Conectar con cuenta Google.
- Integración con WordPress.
- Integración vía API REST.

En la figura 2-2 se puede apreciar una captura de la interfaz principal de la aplicación.

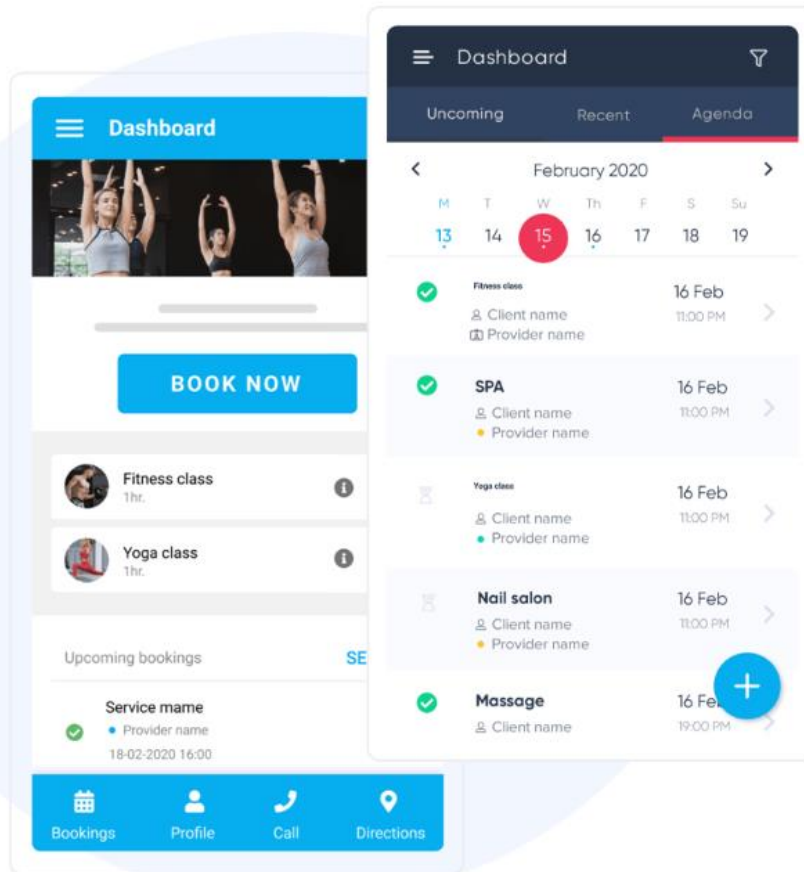


Figura 2-2. Pantalla aplicación SimplyBook

2.3 Cluber

Es una aplicación de control de accesos para instalaciones deportivas. Así mismo, ayuda a gestionar y administrar clubes. Dispone de una versión móvil y otra web. [3] Para poder usarla, los socios y deportistas deben darse de alta. Las principales características de la aplicación son:

- Gestión del cobro de las entradas.
- Sistema de mensajes entre miembros.
- Sistema de notificaciones.
- Gestión de horarios y resultados.
- Organización de eventos.

En la figura 2-3 se puede apreciar una captura de las interfaces principales de la aplicación.

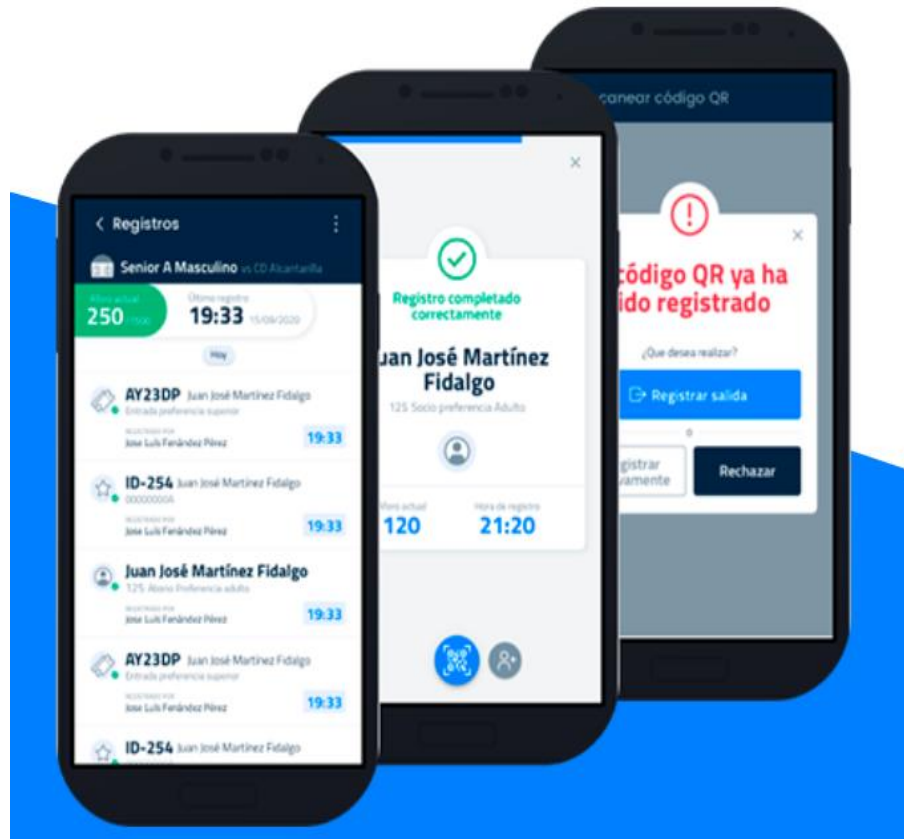


Figura 2-3. Pantalla Aplicación Cluber

2.4 Runkeeper

Es una aplicación diseñada para corredores que cuenta con rutinas específicas para cada tipo de deportista, propuesta de desafíos mensuales y una guía de actividades para deportistas de alto nivel. Así mismo, permite planificar objetivos de entrenamiento, gestionar el progreso del deportista y compartir los resultados de los entrenamientos con otros miembros. [4]

En la figura 2-4 se puede apreciar una captura de la interfaz principal de la aplicación.

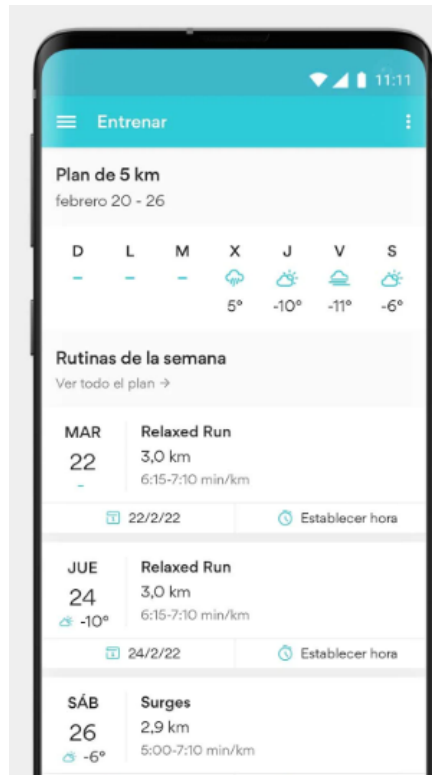


Figura 2-4. Pantalla Aplicación Runkeeper

2.5 Tennis Score Tracker

Es una herramienta que permite gestionar la puntuación de los partidos de tenis. Además, ofrece diferentes tipos de estadísticas y hasta tres modos de puntuación. [5]

Las principales funcionalidades de esta aplicación son:

- Puntuación de torneos.
- Geolocalización de partidos.
- Gestión de equipos y eventos.
- Agregar Canchas.
- Seguimiento y desempeño.

En la figura 2-5 se puede apreciar una captura de la interfaz principal de la aplicación.



Figura 2-5. Pantalla Aplicación Tennis Score

2.6 Clupik

Es una Aplicación que permite administrar. Entre sus funcionalidades ofrece herramientas de comunicación entre los usuarios de la aplicación, la gestión de campeonatos y entrenamientos. [6]

En la figura 2-6 se puede apreciar una captura de la interfaz principal de la aplicación.

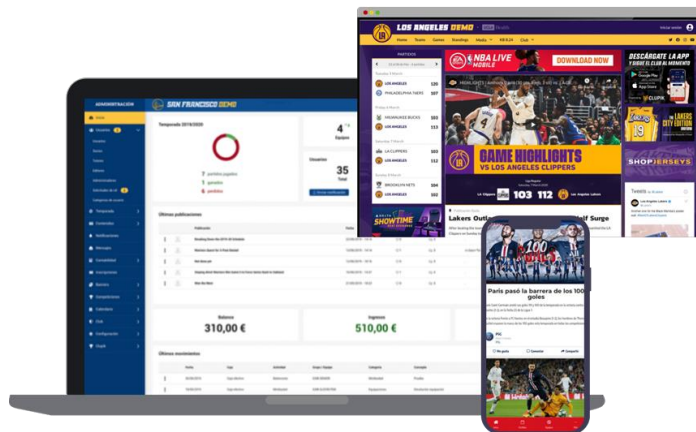


Figura 2-6. Clupik

Capítulo 3 - Especificación de requisitos

En este capítulo se va a describir los actores de la aplicación, así como los casos de uso asociados a cada uno de ellos.

3.1 Actores

A continuación, se describen todos los actores que interactúan en el sistema:

- **Actor deportista:** es aquel que hace uso de la aplicación y puede generar reservas, gestionar una cuenta o inscribirse en un torneo.
- **Actor club:** es aquel que se encarga de dar de alta canchas y torneos.
- **Actor administrador:** es aquel que se encarga de administrar y gestionar la aplicación.
- **Actor público:** es aquel que puede hacer búsquedas de canchas y ver torneos.

3.2 Casos de uso

En este apartado se muestran los actores junto con los casos de uso que se han definido para cada uno de ellos. Así, en la figura 3-1 se muestra el diagrama de casos de uso relativo al actor deportista, en la figura 3-2 se muestra el diagrama de casos de uso relativo al actor club y en la figura 3-3 se muestra el diagrama de casos de uso del administrador. Así mismo, en la figura 3-4 se muestra el diagrama de casos de uso del actor público.

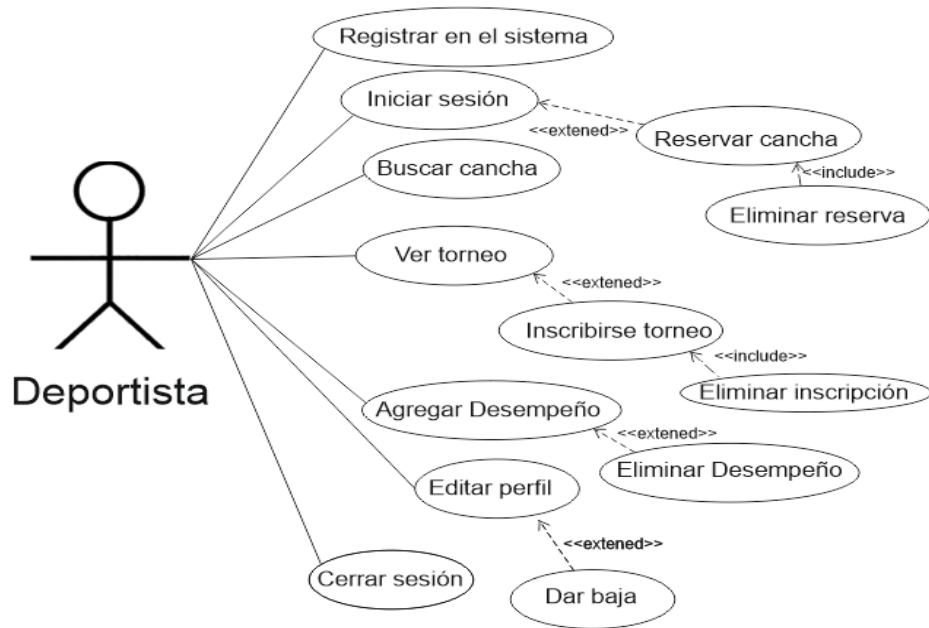


Figura 3-1. Caso de uso - Deportista

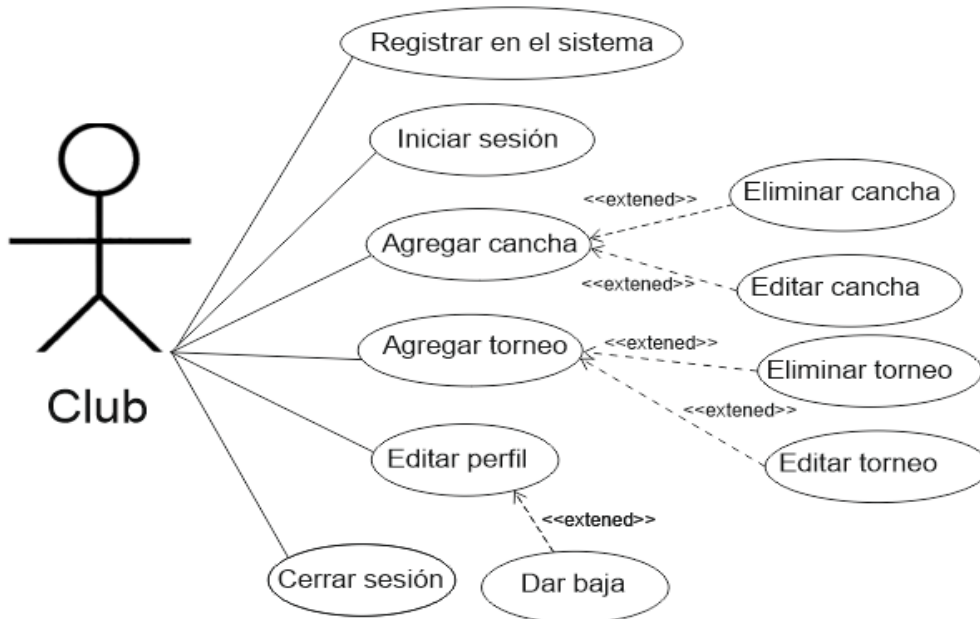


Figura 3-2. Caso de uso - Club

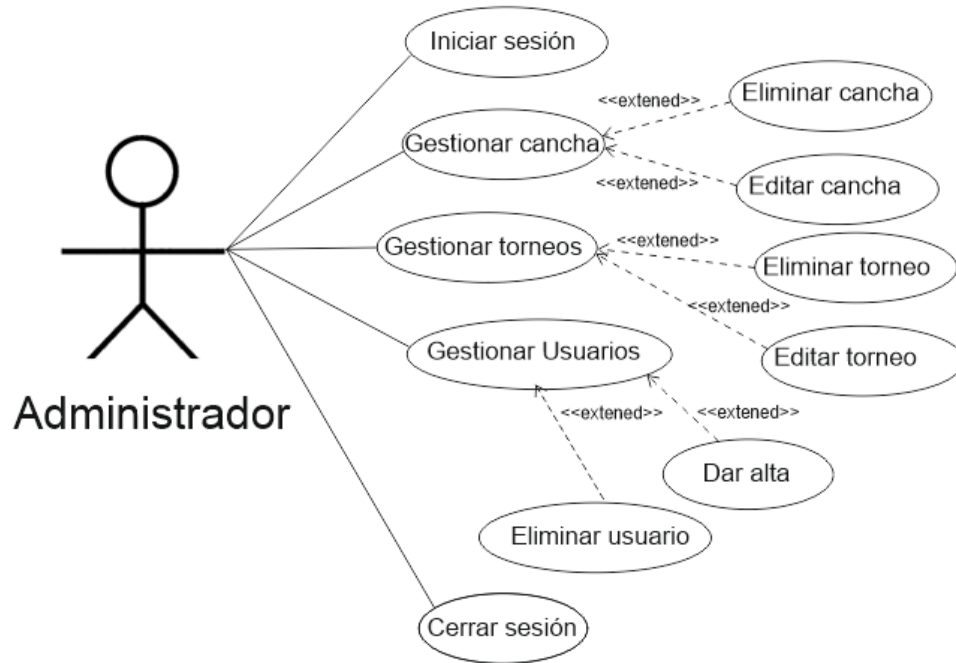


Figura 3-3. Caso de uso – Administrador

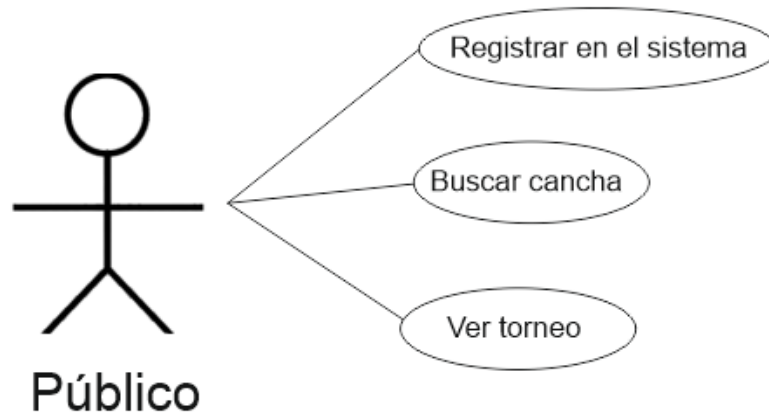


Figura 3-4. Caso de uso – Público

3.3 Especificación de casos de uso

En este apartado se describe cada uno de los casos de uso. Así mismo, las siglas AC1 corresponden al actor deportistas, el AC2 corresponde al club, el AC3 corresponde al administrador y el AC4 corresponde al actor público.

CU-001	Registrar en el sistema	
Versión	1.0	
Autor	Alicia Joy Ramírez	
Dependencia	Alta	
Descripción	El usuario necesita registrarse para hacer uso de la aplicación	
Entradas	Datos personales	
Precondiciones	El usuario espera que lo den de alta	
Salidas	Mensaje "Los datos fueron guardados exitosamente, dentro de 24 horas se lo dará de alta"	
Postcondición Si éxito	El usuario puede interactuar con la interfaz de ingreso de datos	
Postcondición Si fallo	Mensaje "Datos incorrectos o cuenta inactiva"	
Actores	AC1, AC2, AC4	
Secuencia normal	Paso	Acción
	1	Ingresa sus datos
	2	El sistema válido que los datos tienen el formato correcto y que no contenga campos vacíos.
	3	El Sistema muestra un mensaje que se le comunicara su alta dentro de 24 horas.
Secuencia fallo	Paso	Acción
	1	Si los datos de entrada son incorrectos o están vacíos, se le mostrará un mensaje "Verifique sus Datos".
	2	Si no se puede hacer el registro se le mostrará un mensaje "Vuelva a intentar su registro".
Importancia	Vital	
Estabilidad	Alta	

Figura 3-5. Caso de uso 1 - Registrar en el sistema

CU-002	Iniciar sesión	
Versión	1.0	
Autor	Alicia Joy Ramírez	
Dependencia	Alta	
Descripción	El usuario necesita ingresar a la aplicación y necesita sus credenciales.	
Entradas	Credenciales	
Precondiciones	El usuario espera que lo den de alta	
Salidas	El usuario ingreso a menú principal	
Postcondición Si éxito	El usuario puede interactuar con la interfaz menú	
Postcondición Si fallo	Mensaje "Datos incorrectos o cuenta inactiva"	
Actores	AC1, AC2, AC3	
Secuencia normal	Paso	Acción
	1	Ingresa sus credenciales
	2	El sistema válido que los datos
	3	El sistema le direccionará al menú principal
Secuencia fallo	Paso	Acción
	1	Si los datos de entrada son incorrectos o están vacíos se le mostrará un mensaje "Verifique sus Datos".
	2	Si no se puede ingresar se le mostrará un mensaje "Vuelva a intentarlo".
Importancia	Vital	
Estabilidad	Alta	

Figura 3-6. Caso de uso 2 - Iniciar sesión

CU-003	Buscar cancha	
Versión	1.0	
Autor	Alicia Joy Ramírez	
Dependencia	Ninguna	
Descripción	El usuario realiza una búsqueda de una cancha	
Entradas	Provincia o Ciudad	
Precondiciones	El usuario tiene que estar registrado	
Salidas	Lista de las canchas buscadas	
Postcondición Si éxito	Se muestra la lista con las canchas	
Postcondición Si fallo	Se muestran campos vacíos	
Actores	AC1, AC4	
Secuencia normal	Paso	Acción
	1	Ingresar hace una búsqueda
	2	Ingresar por Provincia o ciudad
	3	El Sistema muestra una lista con los resultados
Secuencia fallo	Paso	Acción
	1	Si se ingresaron mal los datos, el sistema no mostrara resultados
	2	No se mostrará la lista
Importancia	Vital	
Estabilidad	Alta	

Figura 3-7. Caso de uso 3 - Buscar cancha

CU-004	Reservar Cancha	
Versión	1.0	
Autor	Alicia Joy Ramírez	
Dependencia	Ninguna	
Descripción	El usuario realiza una reserva de una cancha	
Entradas	Canchas	
Precondiciones	El usuario tiene que estar registrado	
Salidas	Lista con sus reservas	
Postcondición Si éxito	Se le muestra su reserva	
Postcondición Si fallo	Se muestra mensaje error	
Actores	AC1	
Secuencia normal	Paso	Acción
	1	Se le muestra al usuario un listado con las horas disponibles
	2	El usuario selecciona su hora y hace su reserva
	3	El sistema guarda su registro y le muestra su reserva
Secuencia fallo	Paso	Acción
	1	Si no se encuentra un horario se le mostrará con rojo
	2	No se mostrará la lista de horas
Importancia	Vital	
Estabilidad	Alta	

Figura 3-8. Caso de uso 4 - Reservar Cancha

CU-005	Ver Torneo	
Versión	1.0	
Autor	Alicia Joy Ramírez	
Dependencia	Ninguna	
Descripción	El usuario quiere ver los torneos disponibles	
Entradas	Torneos	
Precondiciones	El usuario tiene que estar en la sección Torneos	
Salidas	Lista Torneos	
Postcondición Si éxito	Se le muestra los torneos	
Postcondición Si fallo	Se muestra mensaje error	
Actores	AC1, AC2, AC3, AC4	
Secuencia normal	Paso	Acción
	1	Se le mostrará un listado con los torneos
	2	El usuario selecciona un torneo
	3	El sistema le muestra el detalle del torneo
Secuencia fallo	Paso	Acción
	1	Sin datos
Importancia	Vital	
Estabilidad	Alta	

Figura 3-9. Caso de uso 5 - Ver Torneo

CU-006	Inscribirse Torneo	
Versión	1.0	
Autor	Alicia Joy Ramírez	
Dependencia	Ninguna	
Descripción	El usuario quiere inscribirse en un torneo	
Entradas	Torneos	
Precondiciones	El usuario tiene que estar registrado	
Salidas	Lista torneos inscritos	
Postcondición Si éxito	Inscribirse en el torneo	
Postcondición Si fallo	Se muestra mensaje error	
Actores	AC1	
Secuencia normal	Paso	Acción
	1	El usuario selecciona el torneo que quiere registrarse
	2	El sistema muestra un registro
	3	El usuario se registra y se le muestra un listado con sus inscripciones
Secuencia fallo	Paso	Acción
	1	Mensaje que ya se encuentra registrado
Importancia	Vital	

Figura 3-10. Caso de uso 6 - Inscribirse Torneo

CU-007	Agregar Desempeño	
Versión	1.0	
Autor	Alicia Joy Ramírez	
Dependencia	Ninguna	
Descripción	El usuario quiere agregar sus puntos ganados en desempeño	
Entradas	Desempeño	
Precondiciones	El usuario tiene que estar registrado	
Salidas	Lista desempeño	
Postcondición Si éxito	Agregar desempeño	
Postcondición Si fallo	Se muestra mensaje error	
Actores	AC1	
Secuencia normal	Paso	Acción
	1	El usuario agrega sus datos en el formulario desempeño
	2	El sistema valido los datos e inserta los datos
	3	El sistema muestra un listado con los desempeños agregados, el usuario puede Eliminar su desempeño
Secuencia fallo	Paso	Acción
	1	Datos incorrectos
Importancia	Vital	
Estabilidad	Alta	

Figura 3-11. Caso de uso 7 - Agregar Desempeño

CU-009	Editar Perfil	
Versión	1.0	
Autor	Alicia Joy Ramírez	
Dependencia	Ninguna	
Descripción	Se edita los datos personales del usuario	
Entradas	Datos personales	
Precondiciones	El usuario tiene que estar registrado	
Salidas	Datos actualizados	
Postcondición Si éxito	Se actualizó sus datos	
Postcondición Si fallo	Se muestra mensaje error	
Actores	AC1, AC2	
Secuencia normal	Paso	Acción
	1	El usuario actualiza e ingresa sus datos
	2	El sistema válido los datos e inserta los datos
	3	El sistema le muestra sus datos actualizados
Secuencia fallo	Paso	Acción
	1	Datos incorrectos
Importancia	Vital	
Estabilidad	Alta	

Figura 3-12. Caso de uso 9 - Editar Perfil

CU-010	Dar baja	
Versión	1.0	
Autor	Alicia Joy Ramírez	
Dependencia	Ninguna	
Descripción	El usuario quiere darse de baja del sistema	
Entradas	Datos	
Precondiciones	El usuario tiene que estar registrado	
Salidas	Usuario de baja en el sistema	
Postcondición Si éxito	Se logró la baja	
Postcondición Si fallo	Se muestra mensaje error	
Actores	AC1, AC2	
Secuencia normal	Paso	Acción
	1	El usuario presiona el botón de baja
	2	El sistema le pregunta "Está seguro de que quiere darse de baja"
	3	El usuario confirma y el sistema lo elimina del sistema.
Secuencia fallo	Paso	Acción
	1	Datos incorrectos
Importancia	Vital	
Estabilidad	Alta	

Figura 3-13. Caso de uso 10 - Dar baja

CU-011	Cerrar sesión	
Versión	1.0	
Autor	Alicia Joy Ramírez	
Dependencia	Ninguna	
Descripción	El usuario quiere salir del sistema	
Entradas	Login	
Precondiciones	El usuario tiene que estar registrado	
Salidas	Usuario sale del sistema	
Postcondición Si éxito	Se logró salir del sistema	
Postcondición Si fallo	Se muestra mensaje error	
Actores	AC1, AC2, AC3	
Secuencia normal	Paso	Acción
	1	El usuario presiona el botón salir
	2	El sistema hace la desconexión del usuario
	3	El sistema le muestra la interfaz de login
Secuencia fallo	Paso	Acción
	1	Datos incorrectos
Importancia	Vital	
Estabilidad	Alta	

Figura 3-14. Caso de uso 11 - Cerrar sesión

CU-012	Agregar Cancha	
Versión	1.0	
Autor	Alicia Joy Ramírez	
Dependencia	Ninguna	
Descripción	El usuario va a agregar una nueva cancha	
Entradas	Datos Cancha	
Precondiciones	El usuario tiene que estar registrado	
Salidas	Cancha agregada al sistema	
Postcondición Si éxito	Nueva cancha	
Postcondición Si fallo	Se muestra mensaje error	
Actores	AC2	
Secuencia normal	Paso	Acción
	1	El usuario ingreso los datos de la nueva cancha
	2	El sistema válido los datos y los inserta
	3	El sistema le muestra su cancha
Secuencia fallo	Paso	Acción
	1	Datos incorrectos
Importancia	Vital	

Figura 3-15. Caso de uso 12 - Agregar Cancha

CU-013	Eliminar cancha	
Versión	1.0	
Autor	Alicia Joy Ramírez	
Dependencia	Ninguna	
Descripción	El usuario va a eliminar la cancha	
Entradas	Datos Cancha	
Precondiciones	El usuario tiene que estar registrado	
Salidas	Cancha eliminada	
Postcondición Si éxito	Eliminar cancha	
Postcondición Si fallo	Se muestra mensaje error	
Actores	AC2, AC3	
Secuencia normal	Paso	Acción
	1	El usuario selecciona la cancha a eliminar y presiona el botón
	2	El sistema válido y realizó la eliminación
	3	El sistema muestra el listado de las canchas con la cancha eliminada
Secuencia fallo	Paso	Acción
	1	Datos incorrectos
Importancia	Vital	
Estabilidad	Alta	

Figura 3-16. Caso de uso 13 - Eliminar cancha

CU-014	Editar cancha	
Versión	1.0	
Autor	Alicia Joy Ramírez	
Dependencia	Ninguna	
Descripción	El usuario quiere editar los datos de una cancha	
Entradas	Datos Cancha	
Precondiciones	El usuario tiene que estar registrado	
Salidas	Cancha actualizada	
Postcondición Si éxito	Datos han sido editados correctamente	
Postcondición Si fallo	Se muestra mensaje error	
Actores	AC2, AC3	
Secuencia normal	Paso	Acción
	1	El usuario selecciona la cancha y edita los datos
	2	El sistema válido los campos
	3	El sistema muestra la cancha actualizada
Secuencia fallo	Paso	Acción
	1	Datos incorrectos
Importancia	Vital	
Estabilidad	Alta	

Figura 3-17. Caso de uso 14 - Editar cancha

CU-015	Agregar Torneo	
Versión	1.0	
Autor	Alicia Joy Ramírez	
Dependencia	Ninguna	
Descripción	El usuario quiere agregar un torneo	
Entradas	Datos torneo	
Precondiciones	El usuario tiene que estar registrado	
Salidas	Nuevo torneo publicado	
Postcondición Si éxito	Torneo ha sido publicado	
Postcondición Si fallo	Se muestra mensaje error	
Actores	AC2	
Secuencia normal	Paso	Acción
	1	El usuario agrega los datos del torneo
	2	El sistema válido los datos
	3	El sistema muestra el torneo agregado
Secuencia fallo	Paso	Acción
	1	Si los datos son incorrectos o vacíos se le mostrará un mensaje
Importancia	Vital	
Estabilidad	Alta	

Figura 3-18. Caso de uso 15 - Agregar torneo

CU-016	Eliminar torneo	
Versión	1.0	
Autor	Alicia Joy Ramírez	
Dependencia	Ninguna	
Descripción	El usuario quiere eliminar un torneo	
Entradas	Datos Torneo	
Precondiciones	El usuario tiene que estar registrado	
Salidas	Datos han sido eliminados	
Postcondición Si éxito	Torneo ha sido eliminado	
Postcondición Si fallo	Se muestra mensaje error	
Actores	AC2, AC3	
Secuencia normal	Paso	Acción
	1	El usuario selecciona el torneo y presiona el botón
	2	El sistema válido la eliminación
	3	El sistema muestra el torneo eliminado de la lista
Secuencia fallo	Paso	Acción
	1	Si los datos son incorrectos o vacíos se le mostrará un mensaje
Importancia	Vital	
Estabilidad	Alta	

Figura 3-19. Caso de uso 19 - Eliminar torneo

CU-017	Editar torneo	
Versión	1.0	
Autor	Alicia Joy Ramírez	
Dependencia	Ninguna	
Descripción	El usuario quiere editar un torneo	
Entradas	Datos torneo	
Precondiciones	El usuario tiene que estar registrado	
Salidas	Datos han sido editados	
Postcondición Si éxito	Torneo ha sido actualizado	
Postcondición Si fallo	Se muestra mensaje error	
Actores	AC2, AC3	
Secuencia normal	Paso	Acción
	1	El usuario selecciona el torneo y edita los datos
	2	El sistema válido los datos
	3	El sistema muestra el torneo actualizado
Secuencia fallo	Paso	Acción
	1	Si los datos son incorrectos o vacíos se le mostrará un mensaje
Importancia	Vital	
Estabilidad	Alta	

Figura 3-20. Caso de uso 17 - Editar torneo

CU-018	Gestionar cancha	
Versión	1.0	
Autor	Alicia Joy Ramírez	
Dependencia	Ninguna	
Descripción	El usuario gestiona y administra las canchas	
Entradas	Datos canchas	
Precondiciones	Ninguna	
Salidas	Datos han sido editados, eliminados con éxito	
Postcondición Si éxito	Datos han sido gestionados con éxito	
Postcondición Si fallo	Se muestra mensaje error	
Actores	AC3	
Secuencia normal	Paso	Acción
	1	El usuario selecciona la cancha que quiere gestionar, editar o eliminar
	2	El sistema válido los datos
	3	El sistema muestra la cancha actualizada
Secuencia fallo	Paso	Acción
	1	Si los datos son incorrectos o vacíos se le mostrará un mensaje
Importancia	Vital	
Estabilidad	Alta	

Figura 3-21. Caso de uso 18 - Gestionar Cancha

CU-019	Gestionar torneos	
Versión	1.0	
Autor	Alicia Joy Ramírez	
Dependencia	Ninguna	
Descripción	El usuario gestiona y administra los torneos	
Entradas	Datos torneo	
Precondiciones	Ninguna	
Salidas	Los datos han sido editados o eliminados	
Postcondición Si éxito	Los datos fueron gestionados con éxito	
Postcondición Si fallo	Se muestra mensaje error	
Actores	AC3	
Secuencia normal	Paso	Acción
	1	El usuario selecciona torneo que quiere gestionar, editar, eliminar
	2	El sistema válido los datos
	3	El sistema muestra el torneo actualizado
Secuencia fallo	Paso	Acción
	1	Si los datos son incorrectos o vacíos se le mostrará un mensaje
Importancia	Vital	
Estabilidad	Alta	

Figura 3-22. Caso de uso 19 - Gestionar torneos

CU-020	Gestionar usuarios	
Versión	1.0	
Autor	Alicia Joy Ramírez	
Dependencia	Ninguna	
Descripción	El usuario gestiona y administra los usuarios	
Entradas	Datos torneo	
Precondiciones	Ninguna	
Salidas	Los usuarios fueron editados, eliminados o dados de alta	
Postcondición Si éxito	Los datos fueron gestionados con éxito	
Postcondición Si fallo	Se muestra mensaje error	
Actores	AC3	
Secuencia normal	Paso	Acción
	1	El usuario selecciona el usuario que quiere gestionar, editar, eliminar o dar de alta
	2	El sistema válido los datos
	3	El sistema muestra el usuario actualizado, o dado de alta
Secuencia fallo	Paso	Acción
	1	Si los datos son incorrectos o vacíos se le mostrará un mensaje
Importancia	Vital	
Estabilidad	Alta	

Figura 3-23. Caso de uso 20 - Gestionar usuarios

Capítulo 4 - Tecnologías y Herramientas

En este capítulo se describirán las tecnologías que se utilizaron en la implementación de la aplicación.

4.1 PHP

PHP, es un lenguaje de programación de código abierto que se utiliza para implementar funcionalidades en la parte del servidor.

Sus principales características:

- Compatible con distintos motores de bases de datos.
- Permite generar contenido web de manera dinámica.
- Utiliza módulos llamados extensiones.
- Incluye aplicaciones y frameworks.
- No se rige por una metodología en la programación.

4.2 MySQL

Es un sistema de gestión de bases de datos de código abierto que se utiliza para realizar persistencia de la información de un sistema informático.

Se caracteriza por su alta disponibilidad a latencias, robustez a peticiones y su alto rendimiento. Así mismo, tiene una interfaz intuitiva. [8]

4.3 Xampp

Es un servidor web multiplataforma de formato gratuito y de código abierto. Así mismo, permite a los programadores escribir y probar sus códigos en un servidor web local. Por otro lado, esta aplicación puede cargar varios gestores de contenidos tales como: Joomla, Wordpress, Prestashop y Drupal. [9]

Sus principales características:

- Es Multiplataforma, implica la funcionalidad en Windows y Linux.

- Cuenta con un Panel de control con Apache
- También incluye phpMyAdmin, OpenSSL, MediaWiki, y otros módulos adicionales.
- Permite interactuar con otras bases de datos tales como: MariaDB y SQLite, entre otras.

4.4 Google Chart API

Es una API de Google que permite visualizar datos de manera gráfica que se encuentran en una base de datos. Cuenta con una gran variedad de tipos de gráficos de gráficos lineales y mapas de árboles jerárquicos. Así mismo, ofrece la posibilidad de embeber los gráficos generados en otras aplicaciones siendo compatible con la mayoría de los navegadores (no es necesario un software específico para visualizarlos). [10]

4.5 Node.Js

Es un entorno de tiempo de ejecución de código abierto que utiliza Javascript por el lado del servidor. Así mismo, se utiliza para crear aplicaciones web basado en motor V8 de Google Chrome y se programa de manera asíncrona. Por otro lado, su principal funcionalidad se emplea en servidores controlados mediante eventos. [11]

Sus principales características:

- Altamente escalable y ligero.
- Programación asíncrona para evitar bloqueos de rendimiento
- Se ejecuta en varias plataformas y aplicaciones web.

4.6 Express.js

Es un framework de código abierto para Node.js. Cuenta con herramientas y plugins para simplificar los procesos de desarrollo. [12]

Entre sus principales características se tiene:

- Utiliza JavaScript como lenguaje de programación en la parte frontend y backend.
- Facilita la integración con otras tecnologías tales como (MongoDB, AngularJS).
- Proporciona funcionalidades de enrutamiento, gestión de sesiones y cookies.

4.7 Dialogflow

Es una plataforma que permite implementar un sistema de chatbot donde los usuarios interactúan con la aplicación utilizando lenguajes naturales escritos. Puede ser integrado tanto en aplicaciones web como móviles.

La plataforma permite entradas de texto, imágenes, audio. Así mismo, el sistema proporciona un conjunto de agentes que actúan como bots a la espera de que los usuarios empiecen la interacción.

En la figura 4-1 se puede apreciar el flujo en Dialogflow, donde se ven las peticiones y las respuestas del bot. [13]

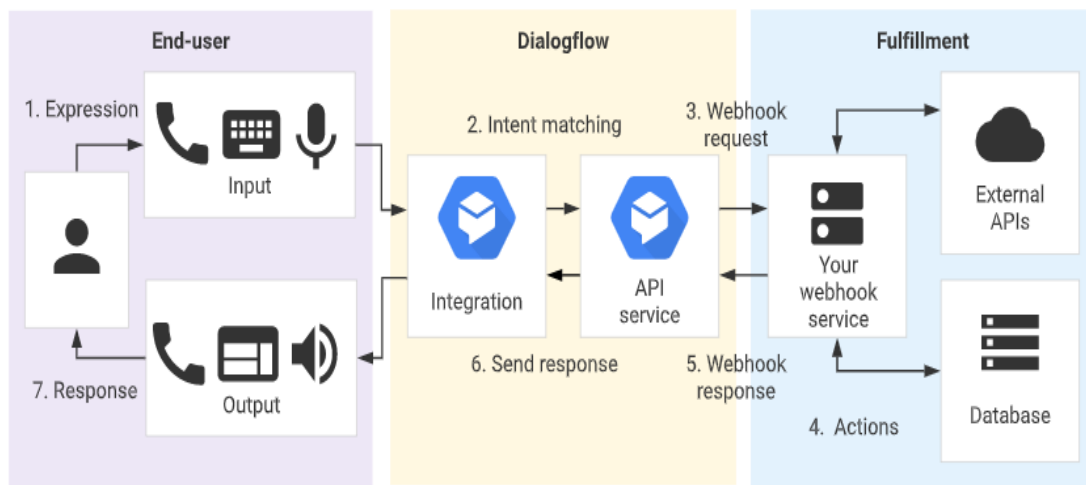


Figura 4-1. Ejemplo flujo Dialogflow

Capítulo 5 - Arquitectura y modelo de datos

En este capítulo se describe la arquitectura de la aplicación desarrollada y el modelo de datos desarrollado.

5.1 Arquitectura

Para implementar la aplicación se ha utilizado una arquitectura cliente-servidor, la cual se caracteriza por tener tres principales componentes tales como, el servidor que se encarga de recibir y enviar datos, las peticiones que hace el cliente y la base de datos donde se almacena toda la información. [14]

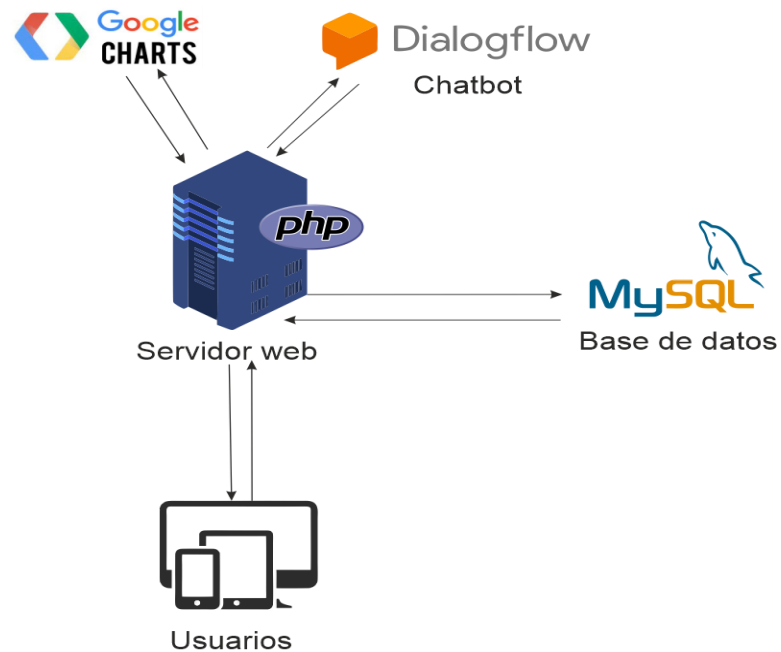


Figura 5-1. Arquitectura del proyecto

En la figura 5-1, se puede observar la interacción de los componentes de la arquitectura desarrollada.

- a) Los usuarios mediante un navegador acceden a la funcionalidad de la aplicación. Así, realizan peticiones que son recibidas por el servidor web.
- b) El servidor puede requerir información que se encuentra almacenada en la base de datos MySQL.

- c) La aplicación hace uso de dos servicios externos tales como, Google Charts y Dialogflow para las interacciones.

5.2 Modelo de Datos

En esta sección, se va a describir las tablas de la base de datos relacional que se ha desarrollado en MySQL para realizar la persistencia de la información de la aplicación.

En la figura 5-2 se muestra el conjunto de tablas que se han diseñado, así como las relaciones que existen entre las mismas.

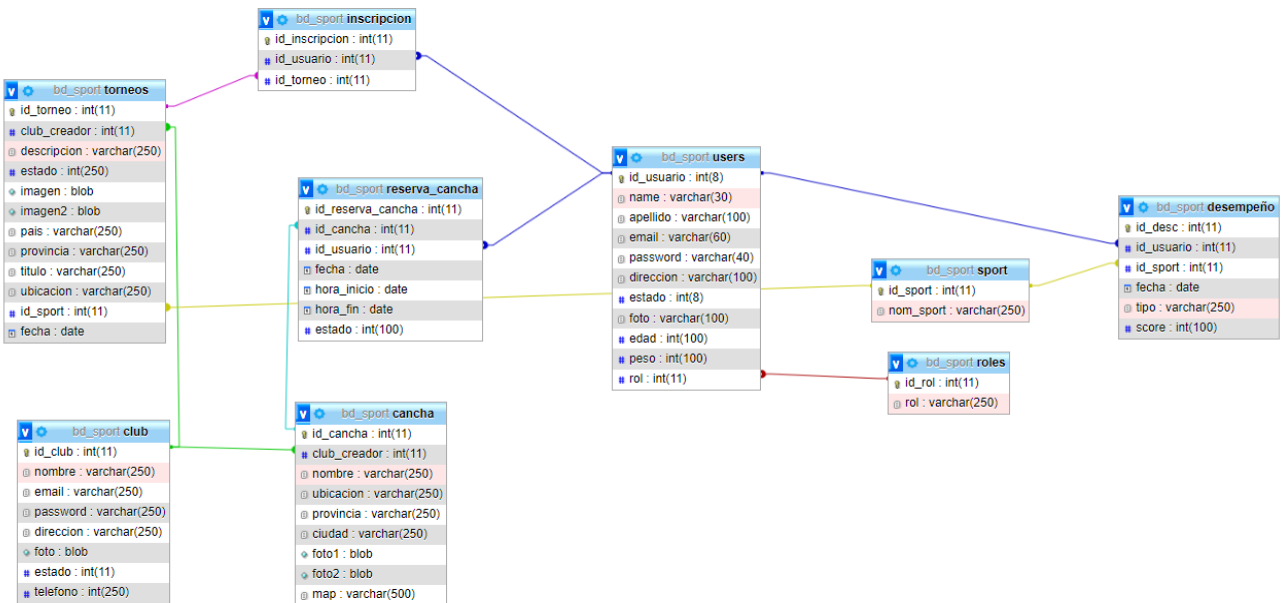


Figura 5-2. Tablas de la base de datos

A continuación, se describe cada una de ellas.

5.2.1 Tabla users

- **Descripción:** La tabla users se utiliza para almacenar todos los usuarios que se registraron en la aplicación.
- **Columnas:**
 - *id_usuario*: identifica únicamente al usuario dentro del sistema.
 - *name*: nombre del usuario.

- *apellido*: apellido del usuario.
- *email*: correo electrónico del usuario.
- *password*: contraseña para ingresar al sistema.
- *dirección*: dirección física del usuario.
- *estado*: con este campo podemos saber si está activo o de baja el usuario.
- *foto*: foto del usuario registrado.
- *edad*: edad del usuario.
- *peso*: peso que tiene el deportista.
- *rol*: tipo de usuario.

5.2.2 Tabla cancha

- **Descripción:** La tabla cancha se utiliza para guardar los datos de la cancha que fue creada por un club.
- **Columnas:**
 - *id_cancha*: identificador único de cada cancha.
 - *club_creador*: identificador único de club creador.
 - *nombre*: nombre de la cancha.
 - *ubicacion*: ubicación física en el cual se encuentra la cancha geográficamente.
 - *provincia*: la provincia donde se encuentra la cancha.
 - *ciudad*: la ciudad donde se encuentra la cancha.
 - *foto1*: foto de logo del club.
 - *foto2*: es una foto de la cancha para dar mejor visibilidad al usuario.
 - *map*: ubicación para Google map.

5.2.3 Tabla *reserva_cancha*

- **Descripción:** La tabla *reserva_cancha* se utiliza para guardar los datos relacionados con la reserva de cada cancha.
- **Columnas:**
 - *id_reserva_cancha*: identificador único de reserva.
 - *id_cancha*: identificador único de cancha.
 - *id_usuario*: el identificador de cada usuario para guardar la reserva.
 - *fecha*: fecha de la reserva.
 - *hora_inicio*: la hora en la que inicia su juego.
 - *estado*: indica el estado que se encuentra la reserva.

5.2.4 Tabla *club*

- **Descripción:** La tabla *club* se utiliza para guardar los datos del club.
- **Columnas:**
 - *id_club*: identificador único de cada club.
 - *name*: nombre del club.
 - *email*: correo electrónico por el cual podemos notificar alguna incidencia.
 - *password*: contraseña para poder ingresar al sistema.
 - *dirección*: dirección física del club.
 - *estado*: con este campo podemos saber si está activo o de baja.
 - *foto*: foto del club.
 - *teléfono*: el número telefónico del club.

5.2.5 Tabla *torneos*

- **Descripción:** La tabla *torneos* se utiliza para guardar los datos del torneo.

- **Columnas:**
 - *id_torneo*: identificador único de cada torneo.
 - *club_creador*: identificador del club.
 - *descripción*: una descripción con el detalle del torneo.
 - *estado*: si es público o privado.
 - *imagen*: imagen publicitaria.
 - *imagen2*: imagen opcional puede ser del local o deportistas.
 - *país*: es el país donde se realiza el torneo.
 - *provincia*: donde se realiza el torneo.
 - *título*: es el título del torneo.
 - *ubicación*: es la ubicación donde se realiza el torneo.
 - *id_sport*: es el tipo de deporte, frontón, frontenis, pelota vasca.
 - *fecha*: la fecha de la realización del torneo.

5.2.6 Tabla inscripción

- **Descripción:** La tabla inscripción se utiliza para guardar los datos de inscripción a cada torneo por parte del deportista.
- **Columnas:**
 - *id_inscripcion*: identificador único de cada inscripción.
 - *id_usuario*: identificador de usuario que se registró.
 - *id_torneo*: el identificador del torneo.

5.2.7 Tabla sport

- **Descripción:** La tabla sport se utiliza para guardar tipos de deportes frontón, frontenis, pelota vasca.
- **Columnas:**

- *id_sport*: identificador de cada deporte.
- *nom_sport*: nombre de cada deporte.

5.2.8 Tabla roles

- **Descripción:** La tabla roles se utiliza para guardar tipos de usuarios.
- **Columnas:**
 - *id_rol*: identificador único del rol.
 - *nom_sport*: nombre del rol, usuario o administrador.

5.2.9 Tabla desempeño

- **Descripción:** La tabla desempeño se utiliza para guardar los desempeños de cada deportista.
- **Columnas:**
 - *id_desc*: identificador único de cada desempeño.
 - *id_usuario*: identificador del usuario.
 - *id_sport*: identificador de cada deporte.
 - *fecha*: es la fecha del juego.
 - *tipo*: puede ser individual o dobles.
 - *score*: los puntos que hizo en el juego.

Capítulo 6 - Implementación

En este capítulo se describirá la implementación de la aplicación, la cual se ha dividido en dos apartados, la aplicación web y el diseño del chatbot.

6.1 Aplicación web

En este apartado se describirá la aplicación web.

6.1.1 Página Inicio

En la figura 6-1 se muestra la página de inicio. Para su diseño se utilizó la plantilla base de W3layouts [15] en HTML, CSS y Bootstrap.

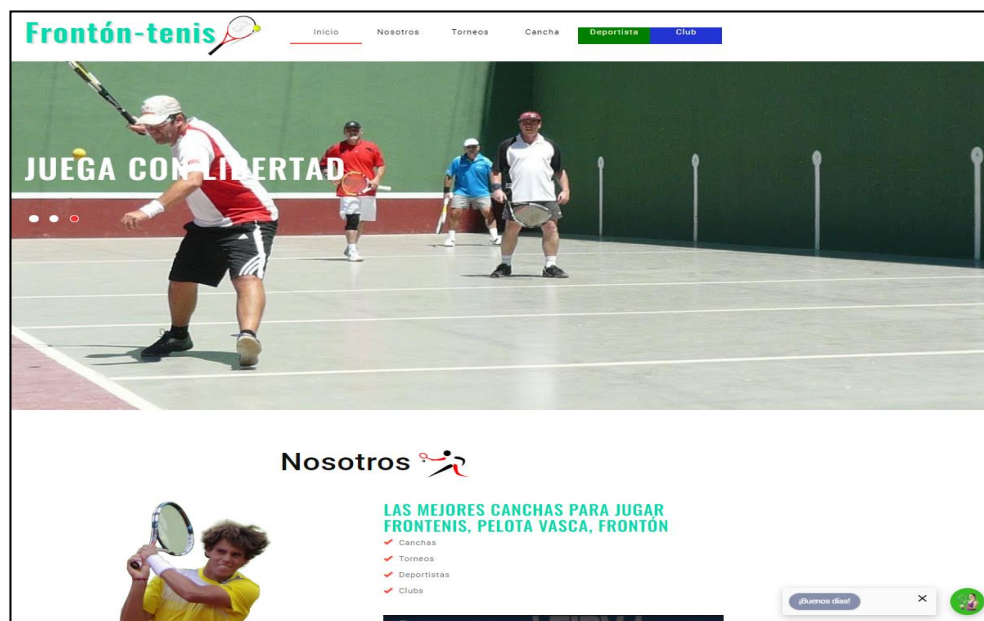


Figura 6-1. Página de inicio

En la figura 6-2 se muestra el código que implementa la página de inicio en el cual se ha utilizado un menú principal, logo, slider y distintos módulos de información.

```

<div class="logo">
  <h1>
    <a href="index.html">
      Frontón-tenis
      
    </a>
  </h1>
</div>
</div>

<div class="collapse navbar-collapse nav-wil" id="bs-example-navbar-collapse-1">
  <nav class="link-effect-9" id="link-effect-9">
    <ul class="nav navbar-nav">
      <li class="active">
        <a class="hvr-overline-from-center scroll" href="index.html">Inicio</a>
      </li>
      <li>
        <a href="#about" class="hvr-overline-from-center scroll">Nosotros</a>
      </li>
      <li>
        <a href="torneos.php">Torneos</a>
      </li>
      <li>
        <a href="cancha.php">Cancha</a>
      </li>
      <li><a href="view/login.php">Deportista</a></li>
      <li><a href="view/login-club.php">Club</a></li>
    </ul>
  </nav>
</div>

```

Figura 6-2 Código página inicio

6.1.2 Registro e inicio de sesión

En este apartado se muestra la implementación del registro del deportista y la función de inicio de sesión.

En la figura 6-3 se puede apreciar una captura de la interfaz inicio de sesión.



Figura 6-3. Pantalla – Registro deportista

La figura 6-4 se muestra la implementación del código para registrar a los usuarios.

```
if (!preg_match("/^[a-zA-Z ]+$/", $name)) {
    $error = true;
    $name_error = "El nombre debe contener solo caracteres del alfabeto y espacio.";
}
if(!filter_var($email,FILTER_VALIDATE_EMAIL)) {
    $error = true;
    $email_error = "Ingresa un correo electrónico válido.";
}
if(strlen($password) < 6) {
    $error = true;
    $password_error = "La contraseña debe tener un mínimo de 6 caracteres.";
}
if($password != $cpassword) {
    $error = true;
    $cpassword_error = "Las contraseñas no coinciden";
}
if(!$terminosycond) {
}
}
if (!$error) {
    if(mysql_query($con, "INSERT INTO users(name,email,password) VALUES(' . $name . ', ' . $email . ', ' . md5($password) . ')") {
        //Mensaje registro con exito
        $successmsg = '
        <div class="alert alert-success alert-dismissible fade in">
        <a href="#" class="close" data-dismiss="alert" aria-label="close">&times;</a>
        <strong>EXITO.!!</strong> ¡Registrado exitosamente! Se le habilitara en menos de 24 horas!!
        </div>
        ';
    } else {

```

Figura 6-4. Registrar en el sistema

En la figura 6-5 se muestra la vista de inicio de sesión por parte del deportista. Así mismo, se puede apreciar el formulario de login con email y contraseña.



Figura 6-5 Pantalla - Inicio de sesión

En la figura 6-6 se observa el código para iniciar sesión. En este trozo de código, en particular se lleva a cabo la validación de las credenciales.

```
<?php
session_start();

if(isset($_SESSION['usr_id'])!="") {
    header("Location: perfil.php");
}
include_once '../controller/dbconnect.php';

if (isset($_POST['login'])) {

    $email = mysqli_real_escape_string($con, $_POST['email']);
    $password = mysqli_real_escape_string($con, $_POST['password']);
    $result = mysqli_query($con, "SELECT * FROM users WHERE email = '" . $email . "' and password = '" . md5($password) . "'");

    if ($row = mysqli_fetch_array($result)) {

        if($row['estado']==1){
            $_SESSION['usr_id'] = $row['id_usuario'];
            $_SESSION['usr_name'] = $row['name'];
            $_SESSION['usr_apellido'] = $row['apellido'];

            header("Location: perfil.php");
        }else
            //si la cuenta todavia no esta dado de alta por el usuario recibe un mensaje.
            $errormsg = "Esta cuenta esta desactivada";
        } else {
            $errormsg = "Revisa los datos!!!";
        }
    }
}
?>
```

Figura 6-6. Código – Inicio de sesión

6.1.3 Menú principal

En la figura 6-7 se muestra el menú de opciones cuando el usuario ha ingresado con éxito. Por otro lado, se aprecian las cuatro funciones: perfil, torneos, desempeño y canchas.



Figura 6-7. Menú principal deportista

En la figura 6-8 se muestra parte del código que implementa la vista del menú de opciones.

```
<!-- Pagina principal para deportista -->
<div class="about" id="about">
  <div class="container">
    <h3 class="title"><?php echo $_SESSION['usr_name']; ?>
    <?php if (isset($_SESSION['usr_id'])) { ?>
      <?php } else { ?>
        <a href="login.php">Login</a>
        <a href="register.php">Registro</a>
      <?php }
    ?>
  </div>
</h3>
<div class="about-top">
  <div class="col-md-4 about-img-agile">
    <div style="margin-left: 15px;" class="row">
      <h3>Menú</h3><br>
      <a href="perfilp.php"></a><br>
      <a href="torneos.php"></a><br>
      <a href="desempeño.php"></a><br>
      <a href="canchas.php"></a><br>
    </div>
  </div>
</div>
```

Figura 6-8. Código - Menú principal

6.1.4 Perfil de usuario

En la figura 6-9 se muestra la interfaz del perfil del deportista con sus datos personales. En particular, desde esta interfaz el deportista puede editar sus datos.

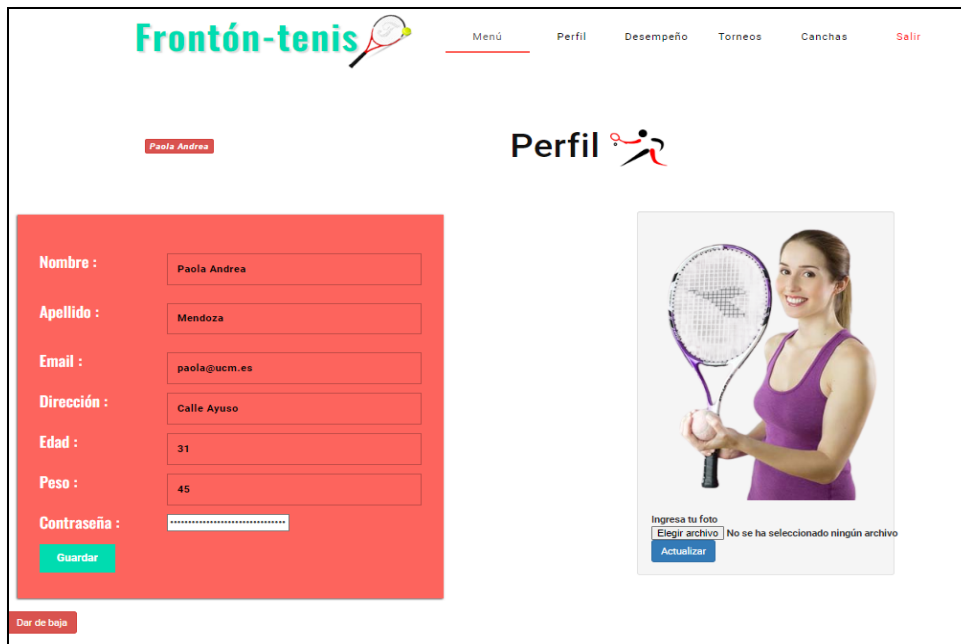


Figura 6-9. Perfil deportista

En la figura 6-10 se muestra una captura de la implementación de la vista de perfil con todos los campos.

```

<?php
include './controller/dbconnect.php';
session_start();
$id1 = $_SESSION['usr_id'];
$consulta = mysql_query($con, "SELECT * FROM users WHERE id_usuario = '$id1'");
$valores = mysql_fetch_array($consulta);

$name = $valores['name'];
$email = $valores['email'];
$apellido = $valores['apellido'];
$direccion = $valores['direccion'];
$password = md5($valores['password']);
$foto = $valores['foto'];
$edad = $valores['edad'];
$peso = $valores['peso'];
//Actualiza los datos en la base de datos.
if(isset($_POST['update'])){
    $id_usuario = $_POST['id_usuario'];
    $query = UPDATE users SET name = $_POST[name], apellido = $_POST[apellido], email = $_POST[email],
    direccion = $_POST[direccion], peso = $_POST[peso], edad = $_POST[edad], password = md5($_POST[password])
    where id_usuario = $_POST[id_usuario]";
    $query_run = mysql_query($con,$query);
    if($query_run){
        echo '<script type="text/javascript"> alert("Datos Guardados Exitosamente");</script>';
    }
    else{
        echo '<script type="text/javascript"> alert("Datos no Actualizados");</script>';
    }
}
?>

```

Figura 6-10. Código - Perfil deportista

En la figura 6-11 se muestra la interfaz del perfil de club, con funcionalidades iguales a la interfaz de perfil del deportista.

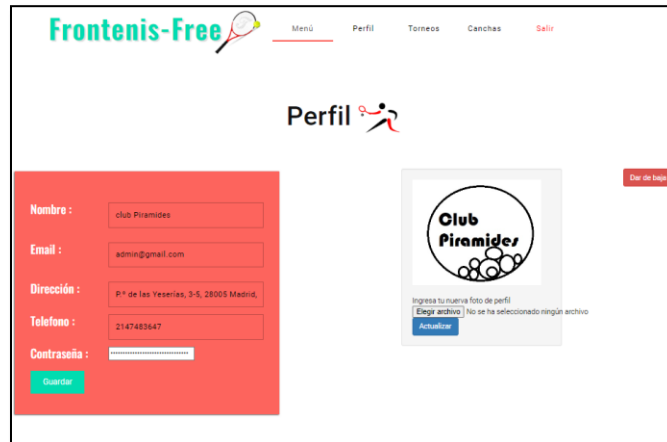


Figura 6-11. Perfil club

En la figura 6-12 se muestra la implementación de la vista de perfil club.

```

<div class="about" id="about">
<div class="container">
<h3 class="title">Perfil
<?php if (isset($_SESSION['usr_id'])) { ?>
<?php } else { ?>
<a href="login.php">Login</a>
<a href="register.php">Registro</a>
<?php ?>
</div></li>
</h3>
<form action="" method="POST" enctype="multipart/form-data">
<div class="about-top">
<div class="col-md-5 col-xs-5 contact-grid-agile">
<div class="contact-right2">
<div class="call">
<input type="hidden" name="id_club" value="<?php echo $id1; ?>"><br>
<div class="col-xs-4 contact-grdr-w3l">
<h3>Nombre :</h3>
</div>
<div class="col-xs-8 contact-grdr-w3l">
<input type="text" name="nombre" placeholder="Nombre.." value="<?php echo $nombre;">
</div><div class="clearfix"></div>

```

Figura 6-12. Código - Dar de baja

6.1.5 Desempeño

En la siguiente figura 6-13 se muestra la vista referida al desempeño de un deportista. Desde esta interfaz el deportista puede ingresar sus puntos de cada partido para llevar un control histórico.

El usuario tiene una tabla con un listado de todos sus registros en el cual puede hacer una búsqueda de sus juegos y puede eliminar o editar los datos.



Figura 6-13. Desempeño deportista

En la figura 6-14 se muestra el código que implementa la página de desempeño en el cual se ha utilizado un formulario y una gráfica en Google Chart.

```
<?php
include("../controller/dbconnect.php");

$sport = $_POST['id_sport'];
$fecha = $_POST['fecha'];
$usuario = $_POST['id_usuario'];
$score = $_POST['score'];
$tipo = $_POST['tipo'];

$query="INSERT INTO desempeño(id_sport,fecha,score,tipo,id_usuario)
VALUES('$sport','$fecha','$score','$tipo','$usuario')";
$resultado= $con ->query($query);

if($resultado){
    header("Location:../view/desempeño.php");
}
else{
    echo "No se agrego";
}
?>
```

Figura 6-14. Agregar Desempeño

En la figura 6-15 se muestra un ejemplo de gráfica de desempeño mediante Google Chart.

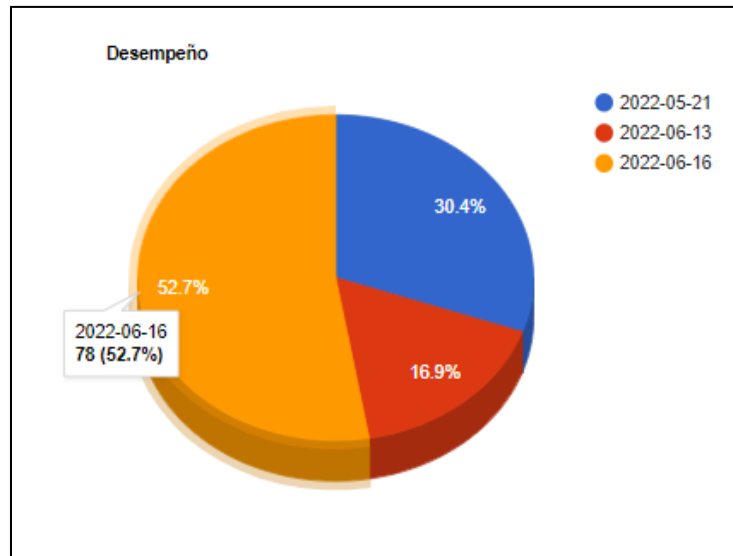


Figura 6-15 Gráfico desempeño

En la figura 6-16 se muestra parte de la implementación de la gráfica que se genera utilizando Google chart. [16]

```
<!--/JS-google -chart -->
<script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
<script type="text/javascript" src="../js/chart.js"></script>
<script type="text/javascript">
  google.charts.load("current", {packages:["corechart"]});
  google.charts.setOnLoadCallback(drawChart);
  function drawChart() {
    var data = google.visualization.arrayToDataTable([
      ['fecha', 'score'],
      <?php
        while($fila=$res->fetch_assoc()){
          echo "[".$fila["fecha"].",".$fila["score"].",";
        }
      ?>
    ]);

    var options = {
      title: 'Desempeño',
      is3D: true,
    };

    var chart = new google.visualization.PieChart(document.getElementById('piechart_3d'));
    chart.draw(data, options);
  }
</script>
```

Figura 6-16. Código - Google Chart

6.1.6 Torneos

En la figura 6-17 se muestra la vista del torneo. Desde esta vista, el deportista puede visualizar los torneos dados de alta por un club.

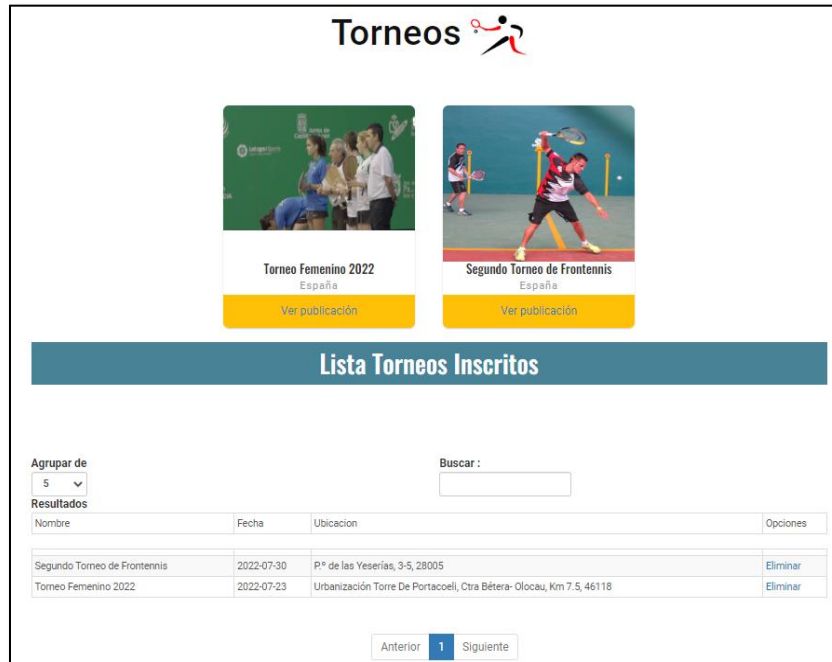


Figura 6-17. Torneos deportistas

En la figura 6-18 se muestra el código que implementa la vista de torneos en el cual se ha utilizado un contenedor para mostrar los torneos y solo se muestran los que están activos.

```
<div class="container2">
  <?php
  include("controller/dbconnect.php");
  $query ="SELECT * FROM torneos WHERE estado=1";
  $resultado =$con->query($query);
  while($row =$resultado->fetch_assoc()){
    ?>
    <div class="vistaPistas">
      <?php echo $row['titulo'];?></h4>
      <p><?php echo $row['pais'];?></p>
      <div class="news-events-agile">
      <a href="ver-torneo.php?id=<?php echo $row['id_torneo'];?>">Ver publicación</a>
      </div> </div>
      <?php
    }
  ?>
</div>
```

Figura 6-18 Código - Ver torneo

En la figura 6-19 se puede apreciar la página referida a un torneo. En particular cuenta con un botón para la inscripción al torneo.



Figura 6-19. Ver torneo

En la figura 6-20 se muestra parte del código que implementa la página de un torneo.

```
<div class="col-md-6 agileits_updates_grid_right">
  <h3>Detalles del Torneo</h3>
  <h2>Fecha: <?php echo $row['fecha'];></h2><br>
  <div class="panel-group" id="accordion" role="tablist" aria-multiselectable="true">
    <div class="panel panel-default">
      <div class="panel-heading" role="tab" id="headingOne">
        <h4 class="panel-title asd">
          <a class="pa_italic" role="button" data-toggle="collapse" data-parent="#accordion" href="#collapseOne" aria-expanded="true"
            aria-controls="collapseOne">
            <span class="glyphicon glyphicon-plus" aria-hidden="true"></span>
            <i class="glyphicon glyphicon-minus" aria-hidden="true"></i> Descripción
          </a>
        </h4>
      </div>
      <div id="collapseOne" class="panel-collapse collapse in" role="tabpanel" aria-labelledby="headingOne">
        <div class="panel-body panel_text">
          <?php echo $row['descripcion'];>
        </div>
      </div>
    </div>
    <div class="panel panel-default">
      <div class="panel-heading" role="tab" id="headingThree">
        <h4 class="panel-title asd">
          <a class="pa_italic collapsed" role="button" data-toggle="collapse" data-parent="#accordion" href="#collapseThree" aria-expanded="
            aria-controls="collapseThree">
            <span class="glyphicon glyphicon-plus" aria-hidden="true"></span>
            <i class="glyphicon glyphicon-minus" aria-hidden="true"></i> Provincia
          </a>
        </h4>
      </div>
      <div id="collapseThree" class="panel-collapse collapse" role="tabpanel" aria-labelledby="headingThree">
        <div class="panel-body panel_text">
          <?php echo $row['provincia'];>
        </div>
      </div>
    </div>
  </div>
</div>
```

Figura 6-20 Código - Ver torneo

En la figura 6-21 se puede apreciar la ventana para confirmar la inscripción a un torneo.



Figura 6-21. Inscripción torneo

En la figura 6-22 se muestra el código que implementa la inscripción a un torneo.

```
<?php
include '../controller/dbconnect.php';
$id3 = $_POST['id_torneo'];
$usuario = $_POST['id_usuario'];
$query="INSERT INTO inscripcion(id_torneo,id_usuario) VALUES('$id3','$usuario')" ;
// Verifica si el usuario ya esta registrado en el torneo
$verificar=mysqli_query($con,"SELECT * FROM inscripcion WHERE id_usuario='$usuario' AND id_torneo='$id3'");

if(mysqli_num_rows($verificar)>0){
    echo '<script>
        alert("Disculpe,ya esta registrado en este torneo");
        window.location="../view/torneos.php";
    </script>'
    ;
    exit();
}
$ejecutar=mysqli_query($con,$query);
if($ejecutar){
    echo '<script>
        alert("Exito");
        window.location="../view/torneos.php";
    </script>'
    ;
}
else{
    echo '<script>
        alert("Duplicado");
        window.location="../view/torneos.php";
    </script>'
    ;
}
?>
```

Figura 6-22. Código - Inscripción torneo

En la figura 6-23 se muestra la vista de agregar torneo. En particular, en la parte inferior aparecen listados los torneos dados de alta desde donde el administrador puede manipularlos.

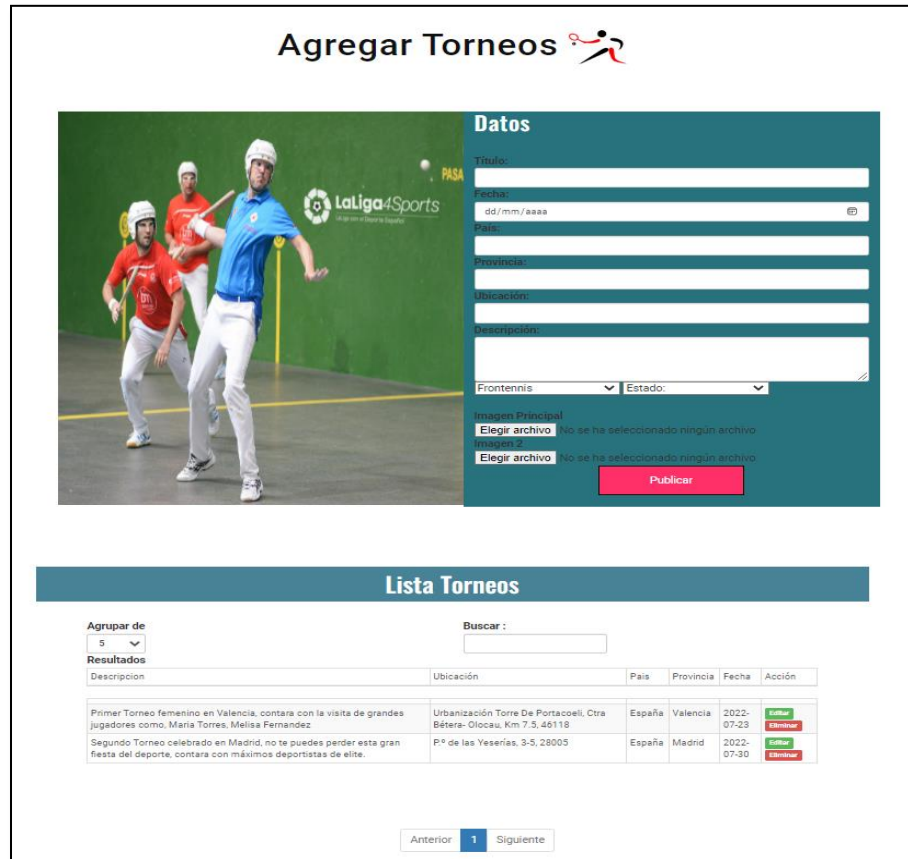


Figura 6-23. Agregar Torneo club

En la figura 6-24 se muestra el código que implementa agregar torneo. Así mismo se puede apreciar los campos del registro.

```

<?php
include("../controller/dbconnect.php");

$club = $_POST['club_creador'];
$descripcion = $_POST['descripcion'];
$estado = $_POST['estado'];
$imagen = addslashes(file_get_contents($_FILES['imagen']['tmp_name']));
$imagen2 = addslashes(file_get_contents($_FILES['imagen2']['tmp_name']));
$pais = $_POST['pais'];
$provincia = $_POST['provincia'];
$titulo = $_POST['titulo'];
$ubicacion = $_POST['ubicacion'];
$sport = $_POST['id_sport'];
$fecha = $_POST['fecha'];

$query="INSERT INTO torneos(club_creador,descripcion,estado,imagen,imagen2, pais, provincia,titulo,ubicacion,id_sport,fecha)
VALUES('$club','$descripcion','$estado','$imagen','$imagen2','$pais','$provincia','$titulo','$ubicacion','$sport','$fecha')";
$resultado= $con ->query($query);

if($resultado){
    header("Location:../view/torneos-club.php");
}
else{
    echo "No se agrego";
}
?>

```

Figura 6-24. Código - Agregar torneo club

6.1.7 Canchas y reservas

En la figura 6-25 se muestra un buscador de canchas de frontón. Así mismo, en la parte inferior existe un botón que permite mostrar las reservas de un usuario.



Figura 6-25. Buscador cancha – Deportista

En la figura 6-26 se muestra parte del código que implementa la vista canchas donde se aprecia la tabla con los datos.

```
<table id="tabla" class="table table-striped table-bordered" style="width:100%; ">
  <thead>
    <tr>
      <td>Nombre</td>
      <td>Ubicación</td>
      <td>Provincia</td>
      <td>Ciudad</td>
      <td colspan="1">Acción</td>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td><?php echo $mostrar['nombre'];></td>
      <td><?php echo $mostrar['ubicacion'];></td>
      <td><?php echo $mostrar['provincia'];></td>
      <td><?php echo $mostrar['ciudad'];></td>
      <td><a class="label label-success" href="ver-cancha.php?id=<?php echo $mostrar['id_cancha'];>"
        class="btn_update" >Ver</a></td>
    </tr>
  </tbody>
</table>
```

Figura 6-26 Código - Tabla cancha

En la figura 6-27 se muestra la ventana para gestionar las reservas. En cada reserva existe la posibilidad de eliminar la reserva y ver el estado de la misma.

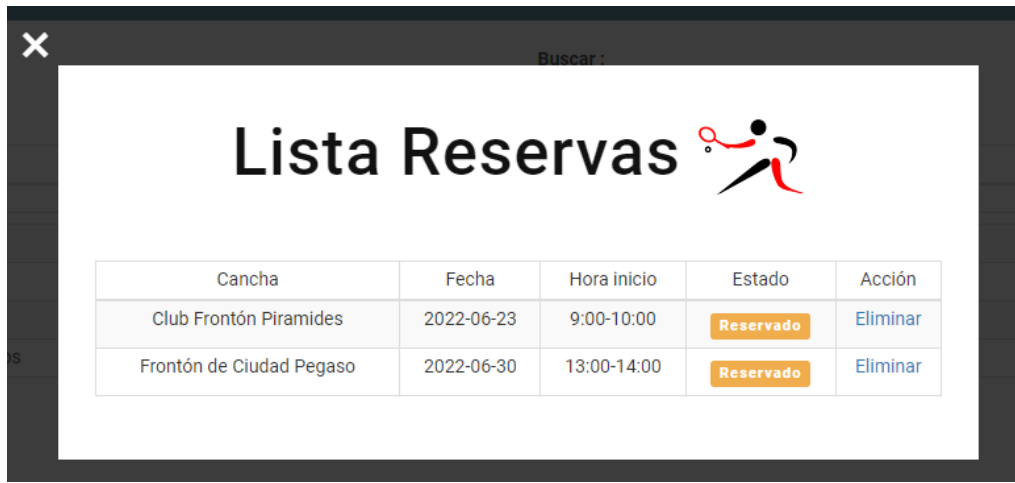


Figura 6-27. Lista Reservas

En la figura 6-28 se muestra la implementación de la ventana lista reserva en la cual se puede apreciar el filtrado de los datos.

```

<tbody>
  <?php
  $query="SELECT c.nombre, r.fecha, r.hora_inicio, r.estado, c.id_cancha, r.id_reserva_cancha
  FROM cancha c
  INNER JOIN reserva_cancha r ON c.id_cancha=r.id_cancha
  WHERE id_usuario='$id'";
  $resultado=mysqli_query($con,$query);
  while ($rows=mysqli_fetch_array($resultado)){
  ?>
  <tr>
    <td><?php echo $rows['nombre'];></td>
    <td><?php echo $rows['fecha'];></td>
    <td><?php echo $rows['hora_inicio'];></td>
    <td>
      <?php
      if($rows['estado']==0){
        echo '<p><a class="label label-success">Libre
        </a></p>';
      }else{
        echo '<p><a class="label label-warning">Reservado
        </a></p>';
      }
      ?>
    </td>
    <td><a href="../model/eliminar-rv.php?id_reserva_cancha=<?php echo $rows['id_reserva_cancha'];>">Eliminar</a></td>
  </tr>
  <?php
  }
  ?>
</tbody>

```

Figura 6-28 Código - Lista reserva

En la figura 6-29 se muestra la vista del detalle de cancha. En particular, se puede apreciar un mapa embebido de Google maps.

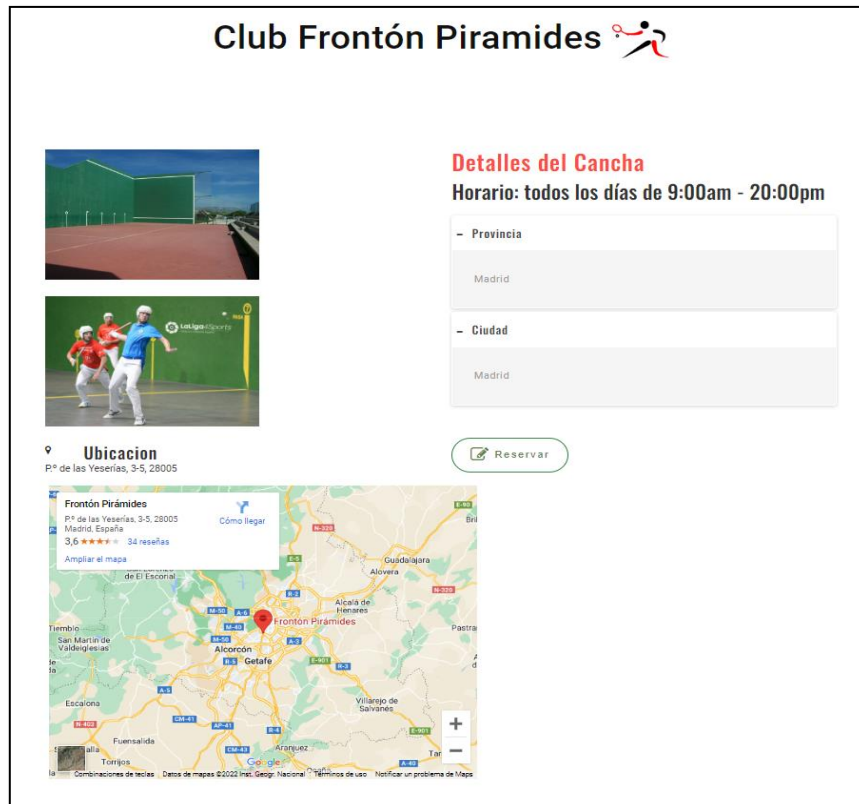


Figura 6-29. Ver Cancha

En la figura 6-30 se muestra la implementación de la vista detalle ver cancha donde se aprecia la consulta a la base de datos.

```

<!-- Datos Cancha-->
<div class="news-section" id="event">
  <div class="container">
    <?php
      $id7=$REQUEST['id'];
      $query ="SELECT * FROM cancha WHERE id_cancha='$id7'";
      $resultado=$con->query($query);
      $row=$resultado->fetch_assoc();
    >
    <h3 class="title"><?php echo $row['nombre'];>
      </h3>
    </div>
  </div>
<!-- Detalle Cancha -->
<div class="services" id="services">
  <div class="container">
    <div class="col-md-6 left-services-agile">
      <div class="left-services-agile1">
        <div class="clearfix"></div>
        
      </div>
      <div class="left-services-agile1">
        <div class="clearfix"></div>
        
      </div>
      <div class="left-services-agile1">
        <i class="fa fa-map-marker" aria-hidden="true"></i>
        <h4>Ubicacion</h4>
        <div class="clearfix"></div>
        <?php echo $row['ubicacion'];>
      </div>
    </div>
  </div>

```

Figura 6-30 Código - Ver cancha

En la figura 6-31 se muestra la página de registro de reserva. En particular, en la parte superior de color verde se encuentra el formulario para realizar reservas introduciendo la fecha y la hora.

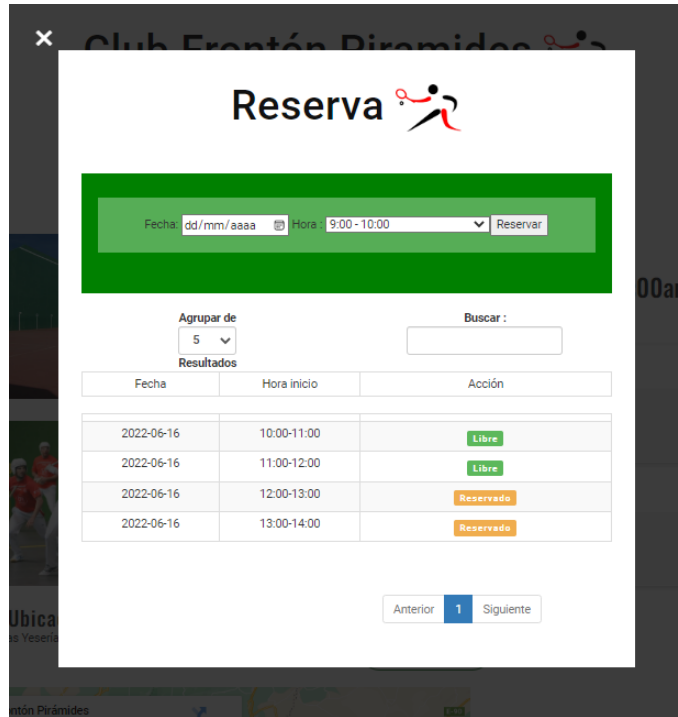


Figura 6-31. Reserva cancha

En la figura 6-32 se muestra la implementación del formulario reserva cancha, donde se puede apreciar los campos y las validaciones.

```
<?php
include '../controller/dbconnect.php';
$cancha = $_POST['id_cancha'];
$usuario = $_POST['id_usuario'];
$fecha = $_POST['fecha'];
$hora_inicio = $_POST['hora_inicio'];
$estado = $_POST['estado'];
$query="INSERT INTO reserva_cancha(id_cancha,id_usuario,fecha,hora_inicio,estado)
VALUES('$cancha','$usuario','$fecha','$hora_inicio','$estado')";

$verificar=mysqli_query($con,"SELECT * FROM reserva_cancha WHERE fecha='$fecha' AND hora_inicio='$hora_inicio'");
// Verifica esta disponible esta reserva
if(mysqli_num_rows($verificar)>0){
    echo '<script>
        alert("Disculpe,ya esta reservado");
        window.location="../view/canchas.php";
    </script>';
    ;
    exit();
}
$ejecutar=mysqli_query($con,$query);
if($ejecutar){
    echo '<script>
        alert("Ya esta su reserva");
        window.location="../view/canchas.php";
    </script>';
    ;
}
else{
    echo '<script>
        alert("No esta disponible");
        window.location="../view/canchas.php";
    </script>';
    ;
}
?>
```

Figura 6-32. Reserva cancha

En la figura 6-33 se muestra el formulario para agregar una cancha por parte de cada club. En la parte inferior se aprecia un listado de todas las canchas registradas para su administración.

Nombre	Ubicación	Provincia	Ciudad	Opciones
Club Frontón Orcasitas	C. del Camino Viejo de Villaverde, 28, 28041	Madrid	Madrid	Editar Eliminar
Club Frontón Pirámides	Pº de las Yserías, 3-5, 28005	Madrid	Madrid	Editar Eliminar
Pirámides4	calle ayuso 23	Córdoba	MADRID	Editar Eliminar

Figura 6-33. Agregar Cancha – Club

En la figura 6-34 se muestra la implementación del código agregar cancha en el cual se puede apreciar los datos del formulario.

```
<?php
include("../controller/dbconnect.php");

$club = $_POST['club_creador'];
$descripcion = $_POST['descripcion'];
$estado = $_POST['estado'];
$imagen = addslashes(file_get_contents($_FILES['imagen']['tmp_name']));
$imagen2 = addslashes(file_get_contents($_FILES['imagen2']['tmp_name']));
$pais = $_POST['pais'];
$provincia = $_POST['provincia'];
$titulo = $_POST['titulo'];
$ubicacion = $_POST['ubicacion'];
$sport = $_POST['id_sport'];
$fecha = $_POST['fecha'];

$query="INSERT INTO torneos(club_creador,descripcion,estado,imagen,imagen2, pais, provincia,titulo,ubicacion,id_sport,fecha)
VALUES('$club','$descripcion','$estado','$imagen','$imagen2','$pais','$provincia','$titulo','$ubicacion','$sport','$fecha')";
$resultado= $con ->query($query);

if($resultado){
    header("Location:../view/torneos-club.php");
}
else{
    echo "No se agrego";
}
}>
```

Figura 6-34 Código - Agregar cancha

6.2 Chatbot

En esta sección se describe el desarrollo del asistente virtual utilizando Dialogflow.

6.2.1 Creación de agente

En la figura 6-34 muestra la creación de un bot llamado agente y sus principales características.

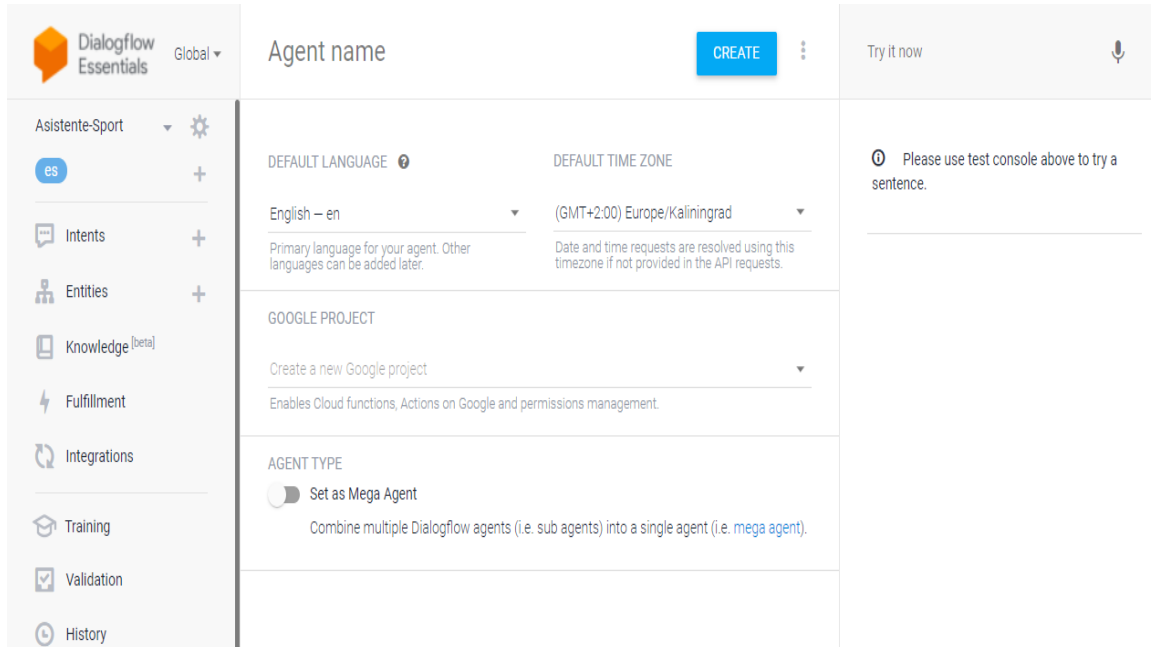


Figura 6-35. Creando agente

En la parte izquierda se muestra el menú de opciones para configurar:

- Intents: palabras ingresadas por el cliente mediante preguntas.
- Entities: son estructuras jerárquicas para entidades.
- Integración: integra la aplicación con otras mediante API Rest.
- Training: frases entrenadas para probar la aplicación.

El listado que se observa en la parte central hace relación al comportamiento que se espera en la conversación con el usuario.

En la figura 6-36 se muestra la lista de intents dentro de Dialogflow.

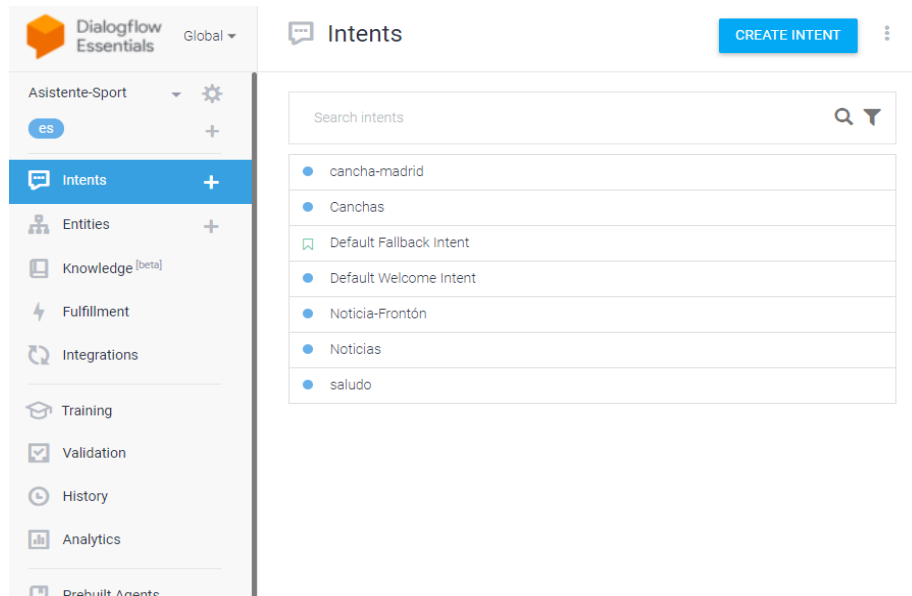


Figura 6-36. Lista de Intents.

6.2.2 Primera conversación

La primera conversación con el usuario comienza en la definición de frases de entrenamiento.

En la tabla 6-1 se muestran las frases introducidas para la interacción referida al saludo.

Intent Name	Training Phrases
Saludo	Hi
	Hola
	Que tal
	Buenos días
	Buenas tardes

Tabla 6-1. Frases de entrenamiento

En la tabla 6-2 se muestran los parámetros de identificador para mejorar las búsquedas.

Action Name	Required	Parametername	Entity	Value	Prompts
Saludo	no	geo-city	@sys.g eo-city	\$geo-city	-

Tabla 6-2. Parámetro identificador

En la figura 6-37 se muestra una captura de implementación de las frases de entrenamiento referidas al saludo.

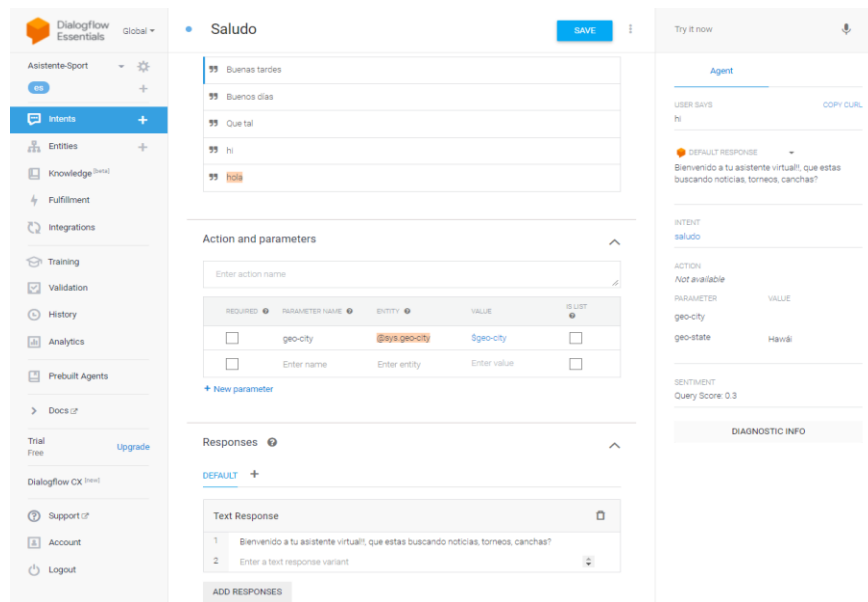


Figura 6-37. Primera conversación

6.2.3 Selección noticias

En la figura 6-38 se puede apreciar las frases de entrenamiento para indicar que se quiere una noticia. Por otro lado, en el apartado de respuestas se agrega uno de los tipos de campos custom payload que se utiliza para ingresar frases de entrenamiento mediante código.

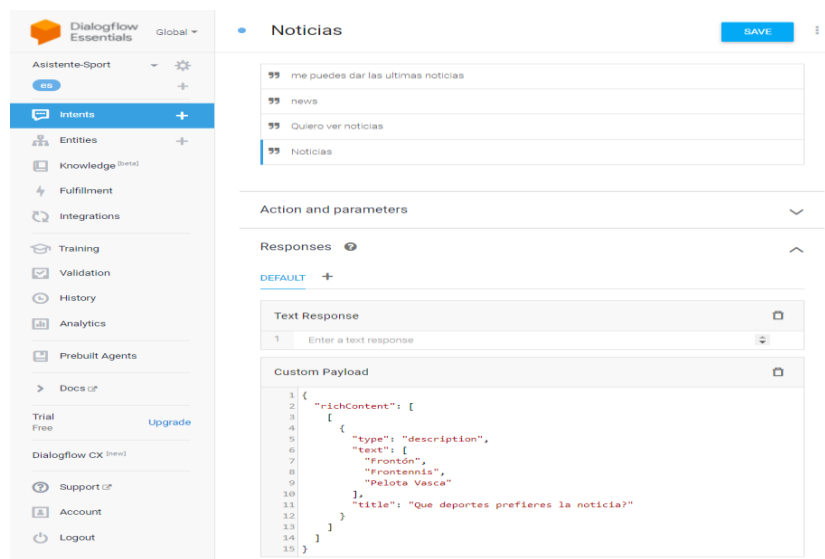


Figura 6-38. Selección de noticias.

6.2.4 Selección frontón

Cuando el usuario selecciona un frontón, se le muestra la noticia, una imagen y el link para que el usuario pueda ver la noticia y se cierra con una segunda pregunta por si necesita más ayuda.

En la figura 6-39 se muestra la captura del Itent noticias-frontón y sus respectivos campos personalizados.

The screenshot shows the Dialogflow Essentials interface for configuring the 'Noticia-Frontón' intent. The left sidebar contains navigation options: Asistente-Sport, Intents (selected), Entities, Knowledge (beta), Fulfillment, Integrations, Training, Validation, History, Analytics, Prebuilt Agents, Docs, Trial Free, Upgrade, Dialogflow CX (new), Support, Account, and Logout. The main area shows the intent configuration for 'Noticia-Frontón' with a 'SAVE' button. The 'Responses' section is expanded to show a 'Text Response' and a 'Custom Payload'.

```
1 {
2   "richContent": [
3     [
4       {
5         "title": "Fiebre pelotazale en San Fermín 2022",
6         "image": {
7           "src": {
8             "rawUrl":
9             "https://imagenes.diariodenavarra.es/files/image_645_359/uploads/2022/06/09/62a1ae257ee0f.jpeg"
10          }
11        },
12        "type": "info",
13        "actionLink": "https://www.diariodenavarra.es/noticias/deportes/pelota/2022/06/09/fiebre-pelotazale-san-fermin-530737-1053.html"
14      }
15    ],
16    [
17      {
18        "text": [
19          "(1)Noticia",
20          "(2)Torneos",
21          "(3)Canchas"
22        ],
23        "type": "description",
24        "title": "Puedo ayudarte en algo más?"
25      }
26    ]
27  ]
28 }
```

Figura 6-39. Selección frontón

6.2.5 Selección cancha

En la figura 6-40 se muestra la implementación en Dialogflow de selección de una cancha de Madrid.

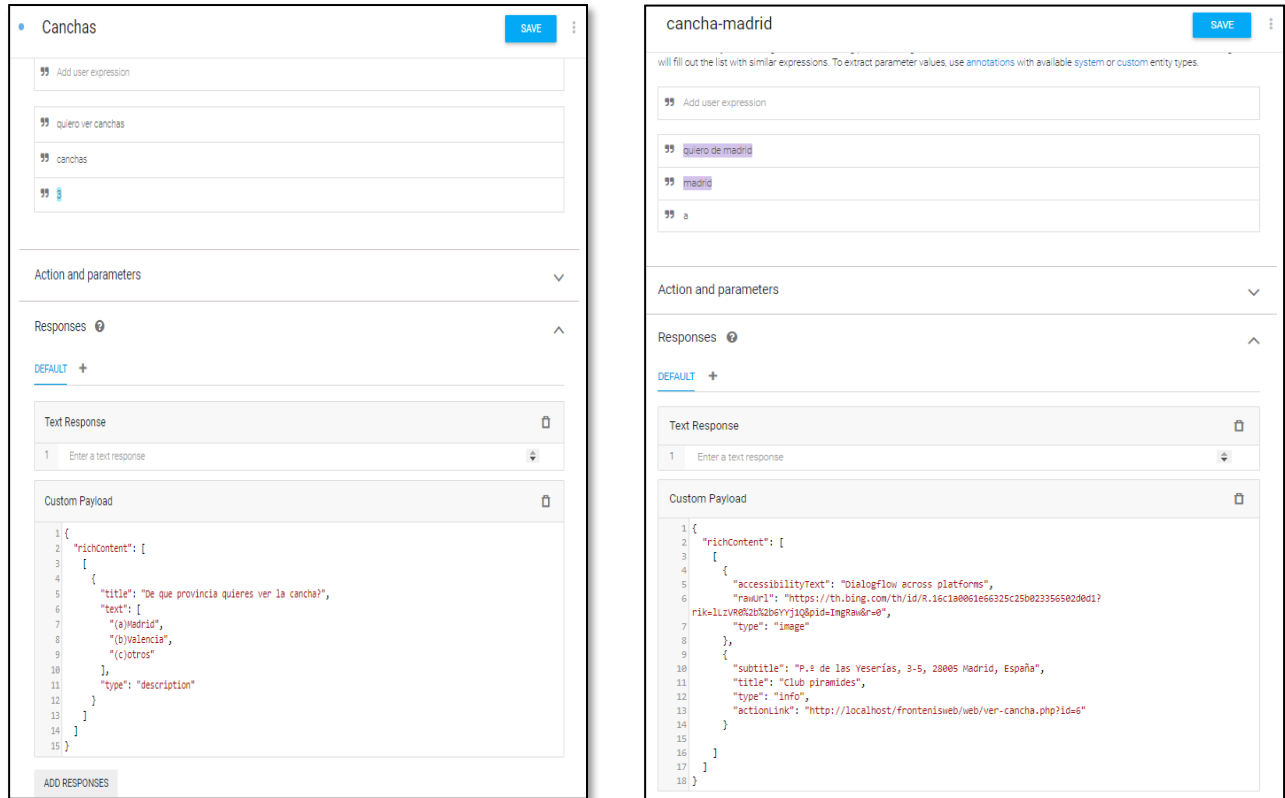


Figura 6-40. Selección cancha

6.2.6 Resultados chatbot

En la parte inferior se muestra el chatbot a la espera que el usuario interactúe y empiece a preguntar.

En la figura 6-41 se muestra la captura de la vista principal con el chatbot en la parte inferior.



Figura 6-41. Asistente virtual – chatbot

En la figura 6-42 se aprecia el código que implementa parte del chatbot.

```

<!-- Chatbot-->
<script src="https://www.gstatic.com/dialogflow-console/fast/messenger/bootstrap.js?v=1"></script>
<df-messenger
  intent="WELCOME"
  chat-title="Asistente deportivo"
  agent-id="2d9d5956-d2b1-4b7e-a890-f341374c90ed"
  language-code="es"
  chat-icon="images/icono-chatbot.png"
>
</df-messenger>
<style>
<style>
df-messenger {
  --df-messenger-bot-message: #878fac;
  --df-messenger-button-titlebar-color: #39bb25;
  --df-messenger-chat-background-color: #fafafa;
  --df-messenger-font-color: white;
  --df-messenger-send-icon: #878fac;
  --df-messenger-user-message: #479b3d;
}
</style>
<!-- //Chatbot-->

```

Figura 6-42. Chatbot

Las primeras conversaciones del chatbot son de bienvenida. A continuación, el asistente hace una pregunta al usuario para que especifique su búsqueda. A continuación, el asistente muestra un listado de noticias. A continuación, el usuario le indica que quiere una noticia sobre frontón, seguidamente el bot le envía su noticia.

Como se aprecia en la figura 6-43 el asistente virtual responde con datos almacenados en Dialogflow.

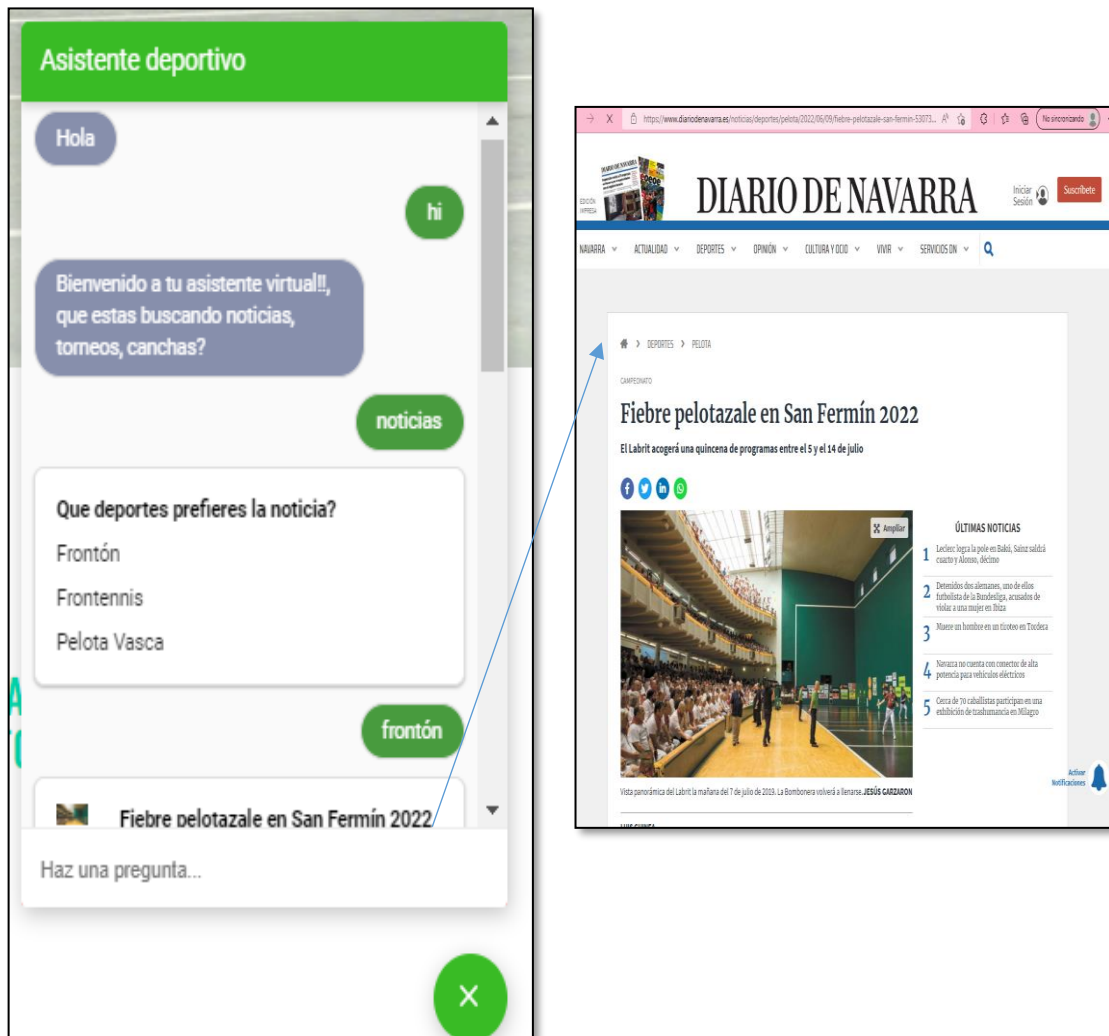


Figura 6-43. Chatbot – Noticias

En la figura 6-44 se muestra un ejemplo de selección de la cancha por provincia. En este caso, el bot hace una búsqueda y clasificación de la respuesta del usuario para mostrar la cancha.

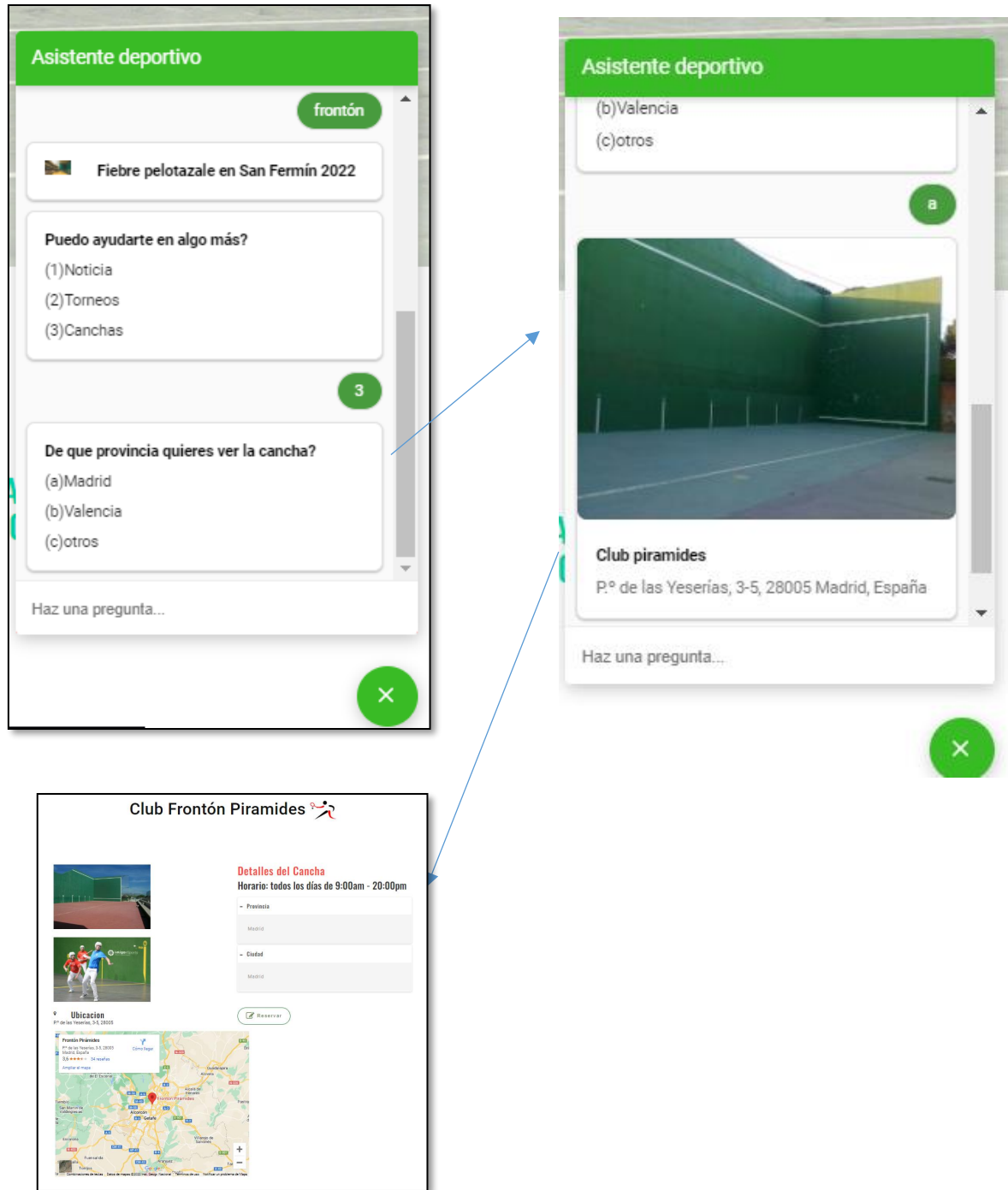


Figura 6-44. Chatbot - Canchas

Capítulo 7 - Conclusiones y trabajo futuro

En este capítulo se presentarán las principales conclusiones obtenidas al finalizar el proyecto. Así mismo, también se describirán algunas ideas de trabajo futuro.

7.1 Conclusiones

En este trabajo se ha desarrollado una aplicación que permite gestionar actividades relacionadas al frontón, a su vez dicha aplicación ayuda a administrar reservas de canchas, también permite agregar un registro de desempeño, Así mismo, los usuarios pueden inscribirse a torneos. Los objetivos planteados al inicio se cumplieron seguidos de un periodo de investigación e implementación, con lo cual se obtuvo una aplicación web disponible para cuatro tipos de actores. Así mismo, esta aplicación cuenta con un buscador de canchas para facilitar a los usuarios en su búsqueda y reserva. Por otro lado, se implementó la funcionalidad de agregar desempeño para tener un registro de cada juego y poder visualizarlo mediante una gráfica creada en Google Chart. En este proyecto se implementó un asistente virtual para ayudar a los usuarios a sentirse acompañados en su búsqueda de información. Por lo tanto, tengan noticias en el ámbito deportivo. Por otro lado, este sistema de chatbot resta tiempo a los administradores web en responder consultas en tiempo real.

Sin embargo, la aplicación presenta algunas limitaciones tales como:

- La ubicación geográfica se ingresa de manera manual en trabajo futuros se describe para su estudio.
- La base de datos inicialmente tiene canchas básicas, se podría hacer un estudio a profundidad encontrando más canchas.

7.2 Trabajo futuro

La funcionalidad de la aplicación puede extenderse de acuerdo con las siguientes líneas de trabajo:

- **Añadir un filtrado en el gráfico de desempeño:** Actualmente, la aplicación genera un gráfico de desempeño con toda la información disponible. En este sentido, se propone como trabajo futuro incluir un mecanismo de configuración que permite generar la gráfica de acuerdo con aspectos tales como puntos, meses, o años, pero se podría hacer un filtrado para mostrar los resultados en base de puntos, meses o años.
- **Inserción automática de la ubicación geográfica de la cancha:** Actualmente, cuando el administrador agrega una cancha tiene que ingresar la ubicación de forma manual. En este sentido, se propone utilizar la API de Google Map para agregar la ubicación de forma automática.
- **Extender las funciones del asistente virtual:** En la aplicación se integró un asistente virtual con funcionalidades básicas. Se propone como trabajo futuro ampliar el dominio de conocimiento del asistente de forma que sea capaz de:
 - Buscar noticias de una fecha específica.
 - Buscar canchas que estén libres en una hora y fecha.
 - Buscar canchas más cerca.
- **Incluir un asistente virtual de voz:** Con el objetivo de hacer accesible la aplicación, se propone agregar un asistente mediante voz para las personas ciegas o con alguna discapacidad, Dado que Dialogflow ya tiene integrado el asistente de voz, sólo sería necesario incluirlo como un nuevo intents.

Chapter - 7. Introduction

This chapter will describe the problem that motivates the realization of this project. Likewise, the proposed objectives, the structure of the memory and the planning followed to carry out the project will be described.

7.1. Motivation

The management of fronton sports courts is complicated given that in general there are no specific applications aimed at the end user, where they can search for a court, manage the reservation of one, or carry out other operations such as signing up for tournaments, being informed of events or sports activities. In this sense, this situation supposes a limitation for users who are fond of the fronton. In the same way, it constitutes a limitation for sports centers and institutions that organize tournaments as they do not have a communication channel with fans and an information management mechanism.

For these reasons, it would be interesting to have a tool that integrates this type of service and facilitates procedures and access to sports facilities and events in the area of the fronton. An application that addresses these limitations should have at least two types of users (athlete, club) and offer functions such as performance aggregation, results management, tournament registration management, court reservation, and user profile management.

This report describes the development of an application that implements a system for managing the services offered by sports centers to fronton fans. Specifically, the application provides functions for court reservation, tournament registration, activity management, results management and information search.

7.2. Objectives

The main objective of this project is to design and implement a web application that allows managing the activities carried out in the fronton courts. This objective is specified in the following more specific objectives:

- Provide a search engine for the general public and registered users.

- Offer the user the option to register.
- Provide the user with a list of tournaments.
- Offer the possibility of making court reservations.
- Have a record of the results of their participation in sports tournaments.
- Graphically show the athlete's performance.
- Have a virtual assistant to help the user in their search for information.

Chapter - 8 Conclusions and future work

This chapter will present the main conclusions obtained at the end of the project. Likewise, some ideas for future projects will also be described.

8.1 Conclusions

In this work, an application has been developed that allows managing activities related to the fronton, in turn said application helps to manage court reservations, it also allows adding a performance record, likewise, users can register for tournaments. The objectives set at the beginning were met, followed by a period of investigation and implementation, which resulted in a web application available for four types of actors. Likewise, this application has a court search engine to facilitate users in their search and reservation. On the other hand, the functionality of adding performance was implemented to have a record of each game and be able to visualize it through a graph created in Google Chart. In this project, a virtual assistant was implemented to help users feel accompanied in their search for information. Therefore, have news in the sports field. On the other hand, this chatbot system takes time away from web administrators in answering queries in real time.

However, the application has some limitations such as:

- The geographical location is entered manually in future work is described for study.

- The database initially has basic courts, an in-depth study could be done to find more courts.

8.2 Future work

The functionality of the application can be extended according to the following lines of work:

- Add a filter in the performance graph: Currently, the application generates a performance graph with all the information available. In this sense, it is proposed as future work to include a configuration mechanism that allows generating the graph according to aspects such as points, months, or years, but a filter could be made to show the results based on points, months, or years. .

- Automatic insertion of the geographical location of the field: Currently, when the administrator adds a field, he has to enter the location manually. In this sense, it is proposed to use the Google Map API to add the location automatically.

- Extend the functions of the virtual assistant: A virtual assistant with basic functionalities was integrated into the application. It is proposed as future work to expand the knowledge domain of the assistant so that it is capable of:

- Search for news from a specific date.

- Search courts that are free at a time and date.

- Find closer courts.

- Include a virtual voice assistant: In order to make the application accessible, it is proposed to add a voice assistant for blind people or people with disabilities. Since Dialogflow already has the voice assistant integrated, it would only be necessary to include it as a new try

BIBLIOGRAFÍA

- [1] SportMember, «SportMember,» [En línea]. Available: <https://www.sportmember.es/>.
- [2] SimplyBook, «SimplyBook,» 2021. [En línea]. Available: <https://simplybook.me/es/>.
- [3] Cluber, «Cluber,» 2022. [En línea]. Available: <https://www.cluber.es/>.
- [4] Runkeeper, «Runkeeper,» [En línea]. Available: <https://play.google.com/store/apps/details?id=com.fitnesskeeper.runkeeper.pro&hl=es&gl=US>.
- [5] T. S. Trancker. [En línea]. Available: <http://tennisscoretracker.com/>.
- [6] Clupik. [En línea]. Available: <https://clupik.com/>.
- [7] PHP.net, «PHP.net,» [En línea]. Available: <https://www.php.net/manual/es/intro-what-is.php>.
- [8] MySQL, «MySQL,» [En línea]. Available: <https://www.mysql.com/about/>.
- [9] Educba, «Educba,» 2022. [En línea]. Available: <https://www.educba.com/what-is-xampp/>.
- [10] GoogleCharts, «GoogleCharts,» [En línea]. Available: <https://developers.google.com/chart/interactive/docs>.
- [11] Educba, «Educba,» [En línea]. Available: <https://www.educba.com/what-is-node-js/>.
- [12] C. Academy, «Code Academy,» [En línea]. Available: <https://www.codecademy.com/article/what-is-express-js>.
- [13] GoogleCloud, «GoogleCloud,» [En línea]. Available: <https://cloud.google.com/dialogflow/docs>.

- [14] L. D. R. G. M. H. S. L. P. Q. L. CASTEDO, «UNED,» [En línea]. Available: <http://e-spacio.uned.es/fez/eserv/taee:congreso-2008-1116/SP118.pdf>.
- [15] W3layouts, «W3layouts,» 2020. [En línea]. Available: <https://w3layouts.com/template/tennis-hub-sports-category-flat-bootstrap-responsive-web-template/>.
- [16] G. Chart, «Google,» [En línea]. Available: https://developers.google.com/chart/interactive/docs/quick_start.
- [17] L. A. Bucki, Word 2013 Bible, John Wiley & Sons, 2013.
- [18] CFI, «Cursos de Formación en Informática,» [En línea]. Available: <http://cursosinformatica.ucm.es>. [Último acceso: 01 06 2019].
- [19] N. P. Europeo, «Europeo, Noticias Parlamento,» 09 2021. [En línea]. Available: <https://www.europarl.europa.eu/news/es/headlines/society/20200827STO85804/que-es-la-inteligencia-artificial-y-como-se-usa>.
- [20] E. M. P. B. A. D. AEISAD, «Munideporte,» Octubre 2010. [En línea]. Available: https://www.munideporte.com/imagenes/documentacion/ficheros/20101105092457congreso_aeisad.pdf. [Último acceso: 2022].
- [21] J. Martinez, «University of Houston,» Febrero 2018. [En línea]. Available: https://uh.edu/ir/reports/special-reports/conference-presentations/tair-2018_googlecharts_martinez.pdf.
- [22] M. C. M. I. A. O. C. M. L. Francisco Escolano, Inteligencia Artificial modelos, Técnicas y Áreas de aplicación, Madrid, España: Thomson Ediciones Spain, 2003.
- [23] ORACLE, «ORACLE,» 2022. [En línea]. Available: <https://www.oracle.com/chatbots/what-is-a-chatbot/>.
- [24] ApacheFriends, «ApacheFriends,» [En línea]. Available: <https://www.apachefriends.org/about.html>.

- [25] A. Schiaffarino, «Infranetworking,» 2019. [En línea]. Available: <https://blog.infranetworking.com/modelo-cliente-servidor/>.
- [26] A. Schiaffarino, «Infranetworking,» 2019. [En línea]. Available: <https://blog.infranetworking.com/modelo-cliente-servidor/>.
- [27] DiyMarkes, «DiyMarkes,» 2014. [En línea]. Available: <http://diymakers.es/raspberry-pi-como-servidor-web/>.
- [28] Google, «Dialogflow,» [En línea]. Available: <https://dialogflow.cloud.google.com/>.
- [29] Google, «GoogleCloud,» [En línea]. Available: <https://cloud.google.com/dialogflow/pricing>.
- [30] Ionos, «Ionos,» 2019. [En línea]. Available: <https://www.ionos.es/digitalguide/servidores/herramientas/instala-tu-servidor-local-xampp-en-unos-pocos-pasos/>.
- [31] V. code. [En línea]. Available: <https://code.visualstudio.com/download>.
- [32] Node.js, «Node.js,» [En línea]. Available: <https://nodejs.org/en/download/>.
- [33] Googlecloud, «Googlecloud,» [En línea]. Available: <https://console.cloud.google.com/>.
- [34] J. A. Quillas, «1Library,» 2016. [En línea]. Available: <https://1library.co/article/el-cambio-de-una-gesti%C3%B3n-deportiva-municipal.qvjvn4gq>.
- [35] F. F. M. L. M. L. Alberto Blanch, «Arsys,» 2017. [En línea]. Available: <https://www.arsys.es/blog/programacion/tomcat-servidores-cloud>.
- [36] Google, «Google,» [En línea]. Available: <https://support.google.com/a/answer/6123896?hl=es>.
- [37] Á. D. León, «Infranetworking,» [En línea]. Available: <https://blog.infranetworking.com/servidor-iis/>.

APÉNDICES

Apéndice A - Manual de usuario

En este apéndice se explica cómo utilizar la aplicación.

A.1. Inicio de sesión y Registro

Una vez que se ingresa en la aplicación, el usuario visualiza la página de inicio.

En la figura A.1 se muestra la captura de la página principal en la cual se observa el botón de login.

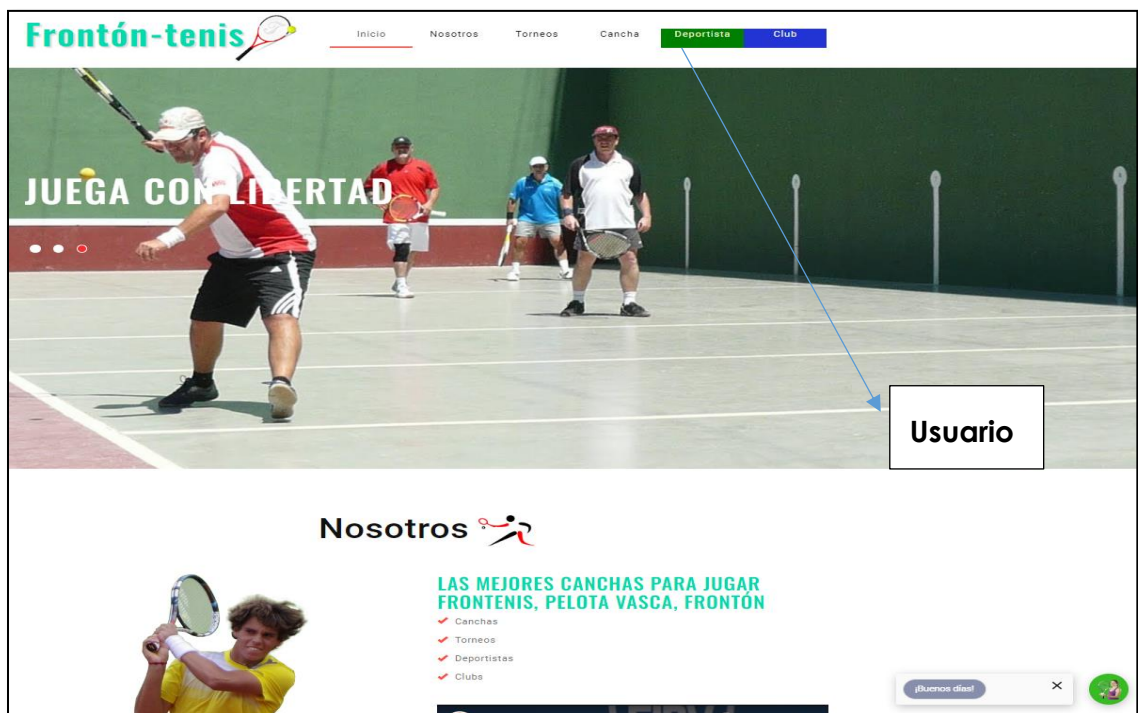


Figura A. 1. Seleccionar tipo usuario

Para hacer uso del sistema se necesita estar registrado, para lo cual desde la página de registro se deben rellenar todos los campos y presionar el botón. Si todos los campos son correctos se mostrará un mensaje.

En la figura A.2 se muestra la pantalla de registro de deportista.



Figura A. 2. Registro usuario

A continuación, puede ingresar a su cuenta introduciendo las credenciales.

En la figura A.3 se muestra la captura de inicio de sesión para ingresar sus credenciales y hacer uso de la aplicación. Está condicionada al alta por parte del administrador.



Figura A. 3. Inicio sesión

A.2. Menú principal

En esta vista se observa el menú principal del deportista que cuenta con cuatro opciones. En la figura A.4 se muestra la captura de menú principal del deportista. Así mismo, en esta vista el deportista cuenta con 4 opciones tales como: el perfil de usuario, la vista torneos para su inscripción, el registro de desempeño y la reserva de cancha.



Figura A. 4. Menú principal

A.3. Perfil

En la opción de perfil puede editar sus datos personales. En particular, se puede cambiar la foto de perfil. En la figura A.5 se muestra la captura de perfil de usuario.



Figura A. 5. Perfil de usuario

A.4. Desempeño

Desde esta vista se puede introducir el desempeño con el objetivo de tener un registro de los puntos.

En la figura A. 6 se puede ver los elementos principales de esta vista:

1. Formulario de desempeño, agregamos el tipo de deporte.
2. Gráfico con los resultados visto en porcentajes.
3. Buscador del desempeño, mostrados por fecha o filtramos por deporte.
4. Se puede eliminar puntos.



Figura A. 6. Desempeño deportista

A.5. Torneos

En la vista del torneo se puede apreciar los torneos que están activos. En el caso de seleccionar alguno se muestra el detalle de torneo:

1. Ver detalle torneo
2. Seleccionar inscribirse en el torneo
3. Lista de torneos inscritos

En la figura A.7 se muestra la captura de torneos. Por otro lado, se puede apreciar los torneos que están activos y un listado de inscripciones.

The figure illustrates the user interface for viewing and registering for a tournament on the Frontenis-Free website. It consists of three main panels connected by arrows:

- Panel 1 (Left):** Shows the 'Torneos' section with two active tournaments: 'Torneo Femenino 2022' and 'Segundo Torneo de Frontennis'. Below this is a 'Lista Torneos Inscritos' section with a search bar and a table of registered participants.
- Panel 2 (Middle):** Shows the 'Detalles del Torneo' for 'Torneo Femenino 2022', including the date (2022-07-23), description, and a registration button.
- Panel 3 (Bottom):** Shows the 'Inscripción' form where the user's name (Paola Andrea) is entered, and a confirmation prompt 'Confirma su inscripción?' is displayed.

Nombre	Fecha	Ubicacion	Opciones
Segundo Torneo de Frontennis	2022-07-30	Pº de las Yeseñas, 3-5, 28005	Eliminar
Torneo Femenino 2022	2022-07-23	Urbanización Torre De Portacoeli, Ctra Bètera- Olocau, Km 7.5, 46118	Eliminar

Figura A. 7. Torneos

A.6. Canchas y reserva

En la vista de cancha se proporciona un buscador por provincia, ciudad o nombre de la cancha.

En la figura A.8 se puede ver los pasos principales para realizar la búsqueda de una cancha:

1. Ver detalle de la cancha
2. Hacer reserva de cancha por fecha y hora.
3. Lista de reservas para gestionar cada reserva.

The diagram illustrates the user flow for finding a court and making a reservation. It consists of three main steps:

- Step 1: Search for Courts (Buscador de Canchas)**
The user is on the 'Frontenis-Free' website. The search results table is as follows:

Nombre	Ubicación	Provincia	Ciudad	Acción
Club Frontón Orcasitas	C. del Camino Viejo de Villaverde, 28, 28041	Madrid	Madrid	Ver
Club Frontón Piramides	Pº de las Yeseñas, 3-5, 28005	Madrid	Madrid	Ver
Frontón Avda. Pablo Iglesias	Av. de Pablo Iglesias, 62, 28039 Madrid	Madrid	Madrid	Ver
Frontón de Ciudad Pego	Av. Séptima, 12, 28022	Madrid	Madrid	Ver
Frontón Municipal de Higuerales	C. De Liria, 5, 46162 Higuerales, Valencia	Valencia	Higuerales	Ver
- Step 2: View Court Details (Club Frontón Piramides)**
The user clicks on the 'Ver' button for 'Club Frontón Piramides'. The details page shows:
 - Detalles del Cancha:** Horario: todos los días de 9:00am - 20:00pm
 - Ubicación:** Pº de las Yeseñas, 3-5, 28005
 - A map showing the location of the court.
 - A 'Reservar' button.
- Step 3: View Reservation List (Lista Reservas)**
The user clicks on the 'Reservar' button. The reservation list is as follows:

Cancha	Fecha	Hora inicio	Estado	Acción
6	2022-06-16	12:00-13:00	Reservada	Eliminar
6	2022-06-16	13:00-14:00	Reservada	Eliminar
3	2022-06-17	10:00-11:00	Reservada	Eliminar

Figura A. 8. Ver cancha y reserva

Apéndice B - Vista del administrador

A continuación, se muestra la vista del administrador del sistema.

C.1. Inicio sesión

El administrador ingresa mediante <http://localhost/Fronton-tenis/admin.php> usando las credenciales: usuario: pamela@gmail.com y contraseña: 123456

En la figura C.1 se muestra la captura de inicio de sesión por parte del administrador.

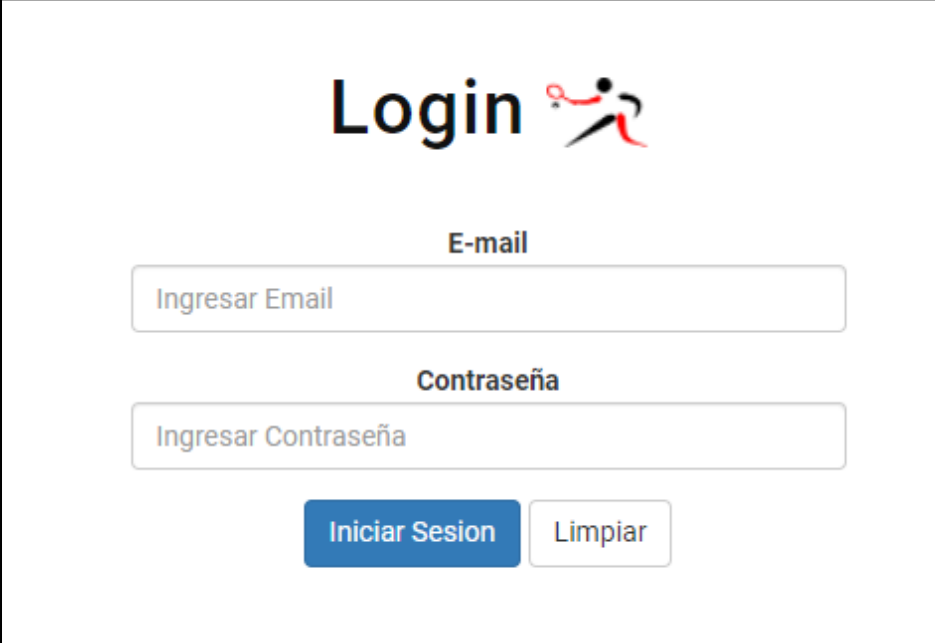


Figura C. 1. Iniciar sesión – Administrador

En la figura C.2. se muestra el menú principal que tiene el administrador y en la figura C.3 se visualiza la gestión de usuarios.

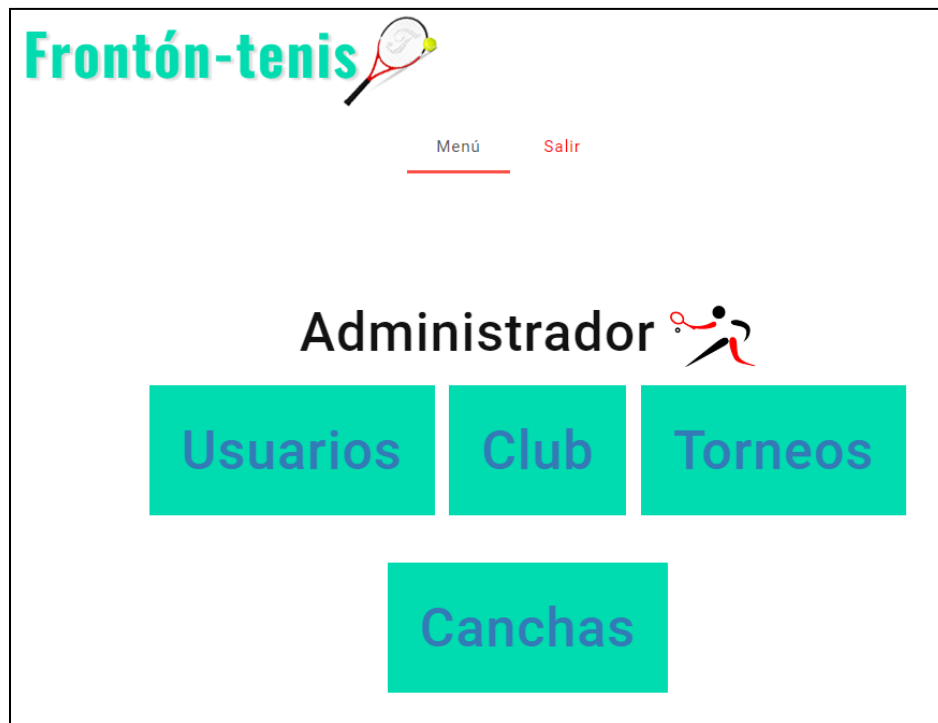


Figura C. 2. Menú - Administrador



Figura C. 3. Gestión usuarios

En la figura C.4 se muestra la captura de gestión de cada club. Así mismo, se observa las opciones de editar o eliminar club.

The screenshot shows the 'Club' management interface. At the top, there is a logo with the word 'Club' and a stylized figure. Below the logo, there is a search bar labeled 'Buscar:' and a dropdown menu for 'Agrupar de' set to '5'. The main content is a table with the following data:

ID ↑ ↓	Nombre	Email	Telefono	Estado ↑ ↓	Opciones
1	club Piramides	admin@gmail.com	2147483647	Activo	editar Eliminar
3	club Orcasitas	club2@gmail.com	652485256	Dar alta	editar Eliminar

At the bottom of the table, there is a pagination control with 'Anterior', '1', and 'Siguiente' buttons.

Figura C. 4. Gestión club

En la figura C.5 se puede apreciar la gestión de torneos. Así mismo, se observa las opciones tales como: editar y eliminar torneo.

The screenshot shows the 'Torneos' management interface. At the top, there is a logo with the word 'Torneos' and a stylized figure. Below the logo, there is a search bar labeled 'Buscar:' and a dropdown menu for 'Agrupar de' set to '5'. The main content is a table with the following data:

ID ↑ ↓	Nombre	Apellidos	Email	Estado ↑ ↓	Opciones
1	Segundo Torneo de Frontennis	España	Madrid	Activo	editar Eliminar
1	Torneo Femenino 2022	España	Valencia	Activo	editar Eliminar

At the bottom of the table, there is a pagination control with 'Anterior', '1', and 'Siguiente' buttons.

Figura C. 5. Gestión Torneos