

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: JUNIO 2022
CALIFICACIÓN:**

**MÁSTER EN INGENIERÍA INFORMÁTICA
FACULTAD DE INFORMÁTICA
UNIVERSIDAD COMPLUTENSE DE MADRID**

DÍA DE MES DE AÑO

DEDICATORIA

A Dios por ayudarme en todo tiempo, a
mis padres y hermana por motivarme y
amar me siempre.

AGRADECIMIENTOS

Quiero empezar dando las gracias a Dios por su ayuda incondicional y por ser mi fuerza en momentos difíciles siendo mi mayor referente en cada paso de mi vida. En este tiempo pude conocer a muchas personas que fueron parte fundamental en mi vida para lograr este proyecto.

Gracias infinitas a mis padres por enseñarme a esforzarme cada día más y a nunca rendirme, por su amor y su paciencia en cada momento de mi vida, ellos siempre estuvieron para levantarme, a mi hermana y mi cuñado gracias infinitas ya que me apoyaron para lograr mis estudios, me acogieron en su casa con mucho cariño.

A mis compañeros que me apoyaron desinteresadamente y por brindarme su tiempo cada momento que lo necesite. Y a los compañeros invisibles de los pasillos de la Universidad, fueron personas que, aunque no me conocían, cuando pedía ayuda siempre me apoyaron.

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

Seguidamente, quiero agradecer a mi asesor Antonio Sarasa Cabezuelo por sus consejos y por 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.

Por ultimo y no menos importante a mi Universidad Complutense de Madrid, por abrirme sus puertas y brindarme las mejores experiencias de mi vida al cursar mis estudios durante esos casi tres años. Tienen las mejores instalaciones y aunque sufrimos una pandemia casi dos años, ellos nunca descansaron y se esforzaron por darnos la mejor educación.

RESUMEN

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

Este Trabajo de Fin de Máster es realizado por Alicia Esther Joy Ramírez, alumna del Máster en ingeniería informática de la Universidad Complutense de Madrid.

El objetivo de este proyecto es desarrollar una aplicación web para gestionar actividades en canchas de frontón. Esta aplicación está orientada a mejorar la calidad en el juego de los deportistas de distintas disciplinas (Frontón, Pelota vasca, Frontennis etc.). Tiene como finalidad mantener a los deportistas informados sobre instalaciones deportivas y torneos, además podrán ver la disponibilidad de canchas para hacer su reserva y a su vez podrán inscribirse a torneos, también con esta aplicación podrán llevar un control de sus juegos y ver su rendimiento.

Esta aplicación está hecha en Php y Mysql, para poder publicarla en versión web y que a su vez sea responsive para versiones móviles y tablet. También hace uso de Google Charts API para mostrar resultados de desempeño, a su vez se integró un Chatbot diseñado como asistente virtual para asesorar en la búsqueda a los deportistas.

Esta aplicación es de código libre y de fácil acceso, puede descargarla en el siguiente enlace. [\(Enlace\)](#)

Palabras clave

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

ABSTRACT

Title

This Master's Thesis is carried out by Alicia Esther Joy Ramírez, a student on the Master's Degree in Computer Engineering at the Complutense University of Madrid.

The objective of this project is to develop a web application to manage activities in fronton courts. This application is aimed at improving the quality and equality of the game in athletes from different disciplines (Frontón, Basque Pelota, Frontennis, etc.). Its purpose is to keep athletes informed about sports facilities and tournaments, they will also be able to see the availability of fields to make their reservation and in turn they will be able to register for tournaments, also with this application they will be able to keep track of their games and see their performance.

This application is made in Php and Mysql, to be able to publish it in a web version and at the same time be responsive for mobile and tablet versions. It also makes use of Google Charts API to show performance results, in turn a Chatbot designed as a virtual assistant was integrated to advise athletes on the search.

This application is open source and easily accessible, you can download it at the following link. [\(Link\)](#)

Keywords

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

ÍNDICE DE CONTENIDOS

Dedicatoria	III
Agradecimientos	V
Resumen.....	VII
Abstract.....	IX
Índice de contenidos	X
Índice de figuras	XIV
Índice de tablas.....	XIX
<u>Capítulo 1 - Introducción</u>	1
1.1 Motivación	2
1.2 Objetivos.....	2
1.3 Plan de trabajo	3
<u>Capítulo 2 - Estado del arte</u>	5
2.1 Gestión de Actividades deportivas	5
2.2 Google Charts API	6
2.3 Inteligencia Artificial	8
2.4 Chatbot	9
2.5 Proyectos similares	11
2.5.1 Runkeeper	11
2.5.2 Tennis Score Trancker.....	11
2.5.3 Clupik	11
<u>Capítulo 3 - Desarrollo del proyecto</u>	12
3.1 Especificación de Requisitos	12
3.1.1 Especificación de Actores	12

3.1.2 Casos de uso	14
3.1.3 Especificación de casos de uso	16
3.1.4 Requisitos funcionales.....	26
3.1.5 Requisitos no funcionales	29
3.2 Arquitectura del proyecto	32
3.3 Modelo de Datos.....	34
3.3.1 Modelo Entidad-Relación	34
3.3.2 Modelo lógico	35
3.3.3 Base de datos.....	36
<u>Capítulo 4 - Tecnologías y Herramientas</u>	41
4.1 Tecnologías web.....	41
4.1.1 PHP	41
4.1.2 MySQL	42
4.1.3 Xampp	42
4.1.4 Google Chart API.....	42
4.1.5 Node.Js	43
4.1.6 Express.js.....	44
4.1.7 Dialogflow.....	44
<u>Capítulo 5 - Implementación</u>	45
5.1 Aplicación web	46
5.1.1 Página Inicio	46
5.1.2 Login y registro.....	47
5.1.3 Menú principal	50
5.1.4 Perfil de usuario	51
5.1.5 Desempeño	53

5.1.6 Torneos	55
5.1.7 Canchas y reservas.....	59
5.2 Chatbot	65
5.2.1 Creación de agente.....	65
5.2.2 Primera conversación.....	67
5.2.3 Selección noticias	68
5.2.4 Selección frontón	69
5.2.5 Selección cancha.....	70
5.2.6 Resultados chatbot.....	71
<u>Capítulo 6 - Conclusiones y trabajo futuro</u>	75
6.1 Resultado frente a los objetivos.....	75
6.1.1 Realizar una aplicación web	75
6.1.2 Proporcionar un buscador de canchas.....	75
6.1.3 Registrarse en el sistema.....	75
6.1.4 Proporcionar listado de torneos	75
6.1.5 Sistema de reserva	76
6.1.6 Agregar desempeño	76
6.1.7 Ver desempeño en gráfico de Google chart	76
6.1.8 Disponer de un asistente virtual	76
6.2 Trabajo futuro	77
6.2.1 Añadir un filtrado en el <u>gráfico</u> desempeño	77
6.2.2 Insertar ubicación geográfica	77
6.2.3 Entrenar con las funcione a nuestro asistente.....	77
6.2.4 Asistente virtual de voz	77
Chapter - 7. Introduction	79

7.1. Motivation.....	80
7.2. Objetives.....	80
Chapter - 8 Conclusions and future work.....	83
Bibliografía.....	87
Apéndices	91

ÍNDICE DE FIGURAS

Figura 1-1. Plan de trabajo.....	3
Figura 2-1. Gestión de la calidad.....	6
Figura 2-2. Google Charts Ejemplo	7
Figura 2-3. Problema de control de un robot móvil.....	9
Figura 2-4. Captura del video explicativo de Oracle	10
Figura 3-1. Caso de uso - Deportista.....	14
Figura 3-2. Caso de uso – Club.....	15
Figura 3-3. Caso de uso - Administrador.....	15
Figura 3-4. Caso de uso - Público.....	16
Figura 3-5. Caso de uso 1 - Registrar en el sistema.....	16
Figura 3-6. Caso de uso 2 - Iniciar sesión	17
Figura 3-7. Caso de uso 3 - Buscar cancha.....	17
Figura 3-8. Caso de uso 4 - Reserva Cancha.....	18
Figura 3-9. Caso de uso 5 - Ver Torneo	18
Figura 3-10. Caso de uso 6 - Inscribirse Torneo.....	19
Figura 3-11. Caso de uso 7 - Agregar Desempeño	19
Figura 3-12. Caso de uso 9 - Editar Perfil.....	20
Figura 3-13. Caso de uso 10 - Dar baja.....	20
Figura 3-14. Caso de uso 11 - Cerrar sesión.....	21
Figura 3-15. Caso de uso 12 - Agregar Cancha	21
Figura 3-16. Caso de uso 13 - Eliminar cancha	22
Figura 3-17. Caso de uso 14 - Editar cancha	22
Figura 3-18. Caso de uso 15 - Agregar torneo	23

Figura 3-19. Caso de uso 19 - Eliminar torneo	23
Figura 3-20. Caso de uso 17 - Editar torneo.....	24
Figura 3-21. Caso de uso 18 - Gestionar Cancha.....	24
Figura 3-22. Caso de uso 19 - Gestionar torneos	25
Figura 3-23. Caso de uso 20 - Gestionar usuarios.....	25
Figura 3-24. Arquitectura del proyecto	33
Figura 3-25. Arquitectura chatbot.....	34
Figura 3-26. Modelo Entidad-Relación del proyecto	34
Figura 3-27. Modelo lógico Cancha - Reserva	35
Figura 3-28. Modelo lógico Torneo - inscripción	35
Figura 4-1. Ejemplo código	43
Figura 4-2. Ejemplo flujo Dialogflow	44
Figura 5-1. Diagrama de despliegue	45
Figura 5-2. Página de inicio	46
Figura 5-3. Pantalla – Registro deportista	47
Figura 5-4. Pantalla - Iniciar sesión.....	47
Figura 5-5. Registrar en el sistema	48
Figura 5-6. Iniciar sesión.....	49
Figura 5-7. Cerrar sesión	49
Figura 5-8. Menú principal deportista	50
Figura 5-9. Código menú principal	50
Figura 5-10. Perfil deportista.....	51
Figura 5-11. Código - Perfil deportista.....	51
Figura 5-12. Perfil club.....	52
Figura 5-13. Código - Dar de baja	52

Figura 5-14. Desempeño deportista.....	53
Figura 5-15. Agregar Desempeño.....	54
Figura 5-16. Código - Google Chart	54
Figura 5-17. Código – Gráfica	54
Figura 5-18. Código - Eliminar Desempeño	55
Figura 5-19. Torneos deportistas	55
Figura 5-20. Ver torneo	56
Figura 5-21. Inscripción torneo.....	56
Figura 5-22. Código - Inscripción torneo	57
Figura 5-23. Popup – Inscripción.....	57
Figura 5-24. Agregar Torneo club.....	58
Figura 5-25. Código - Agregar Torneo club.....	58
Figura 5-26. Buscador cancha – pública.....	59
Figura 5-27. Buscador cancha – Deportista	60
Figura 5-28. Lista Reservas.....	60
Figura 5-29. Ver Cancha	61
Figura 5-30. Reserva cancha	62
Figura 5-31. Cambio estado reserva.....	62
Figura 5-32. Tabla Reserva	63
Figura 5-33. Reserva cancha	63
Figura 5-34. Agregar Cancha - Club	64
Figura 5-35. Diagrama de secuencia Primera conversación	65
Figura 5-36. Creando agente	66
Figura 5-37. Lista de Intents.....	66
Figura 5-38. Primera conversación	67

Figura 5-39. Diagrama de secuencia - Tipo deporte	68
Figura 5-40. Selección de noticias.....	68
Figura 5-41. Diagrama de secuencia - Selección frontón	69
Figura 5-42. Selección frontón	69
Figura 5-43. Diagrama de secuencia - Cancha.....	70
Figura 5-44. Selección cancha.....	70
Figura 5-45. Asistente virtual - chartbot.....	71
Figura 5-46. Chatbot.....	71
Figura 5-47. Chatbot – Noticias	72
Figura 5-48. Chatbot - Canchas.....	73

ÍNDICE DE TABLAS

Tabla 3-1. Actor principal Deportista	12
Tabla 3-2. Actor Principal Club	12
Tabla 3-3. Actor Administrador.....	13
Tabla 3-4. Actor Secundario Público	13
Tabla 3-5. Requisito funcional 01	26
Tabla 3-6. Requisito funcional 02.....	26
Tabla 3-7. Requisito funcional 03.....	26
Tabla 3-8. Requisito funcional 04.....	27
Tabla 3-9. Requisito funcional 05.....	27
Tabla 3-10. Requisito funcional 06.....	27
Tabla 3-11. Requisito funcional 07.....	28
Tabla 3-12. Requisito funcional 08.....	28
Tabla 3-13. Requisito funcional 09	28
Tabla 3-14. Requisito funcional 10	28
Tabla 3-15. Requisito funcional 11	29
Tabla 3-16. Requisito funcional 12.....	29
Tabla 3-17. Requisito no funcional 01.....	29
Tabla 3-18. Requisito no funcional 02.....	30
Tabla 3-19. Requisito no funcional 03.....	30
Tabla 3-20. Requisito no funcional 04.....	30
Tabla 3-21. Requisito no funcional 05.....	31
Tabla 3-22. Requisito no funcional 06.....	31
Tabla 3-23. Requisito no funcional 07.....	31

Tabla 3-24. Requisito no funcional 08.....	32
Tabla 3-25. Requisito no funcional 09.....	32
Tabla 5-1. Frases de entrenamiento.....	67
Tabla 5-2. Parámetro identificador	67

Capítulo 1 - Introducción

En estos últimos años hemos visto la gran necesidad de las personas de desconectarse del trabajo, de lograr una estabilidad física y mentalmente mediante el deporte acudiendo a espacios de esparcimiento, instalaciones deportivas y debido a este crecimiento nos encontramos bajo una masa grande de personas que acuden a dichas instalaciones sin control alguno y desconociendo sobre actividades como torneos u otros.

A su vez este crecimiento trae descontento a pequeños y grandes deportistas al momento de acudir a las pocas instalaciones públicas que encontramos en nuestra comunidad y al momento de ir estas se encuentran llenas teniendo que esperar para hacer uso de las canchas de hasta 3 horas, y a no ver una buena gestión administrativa nos vemos ante personas que se apoderan de las canchas impidiendo a que otros deportistas jueguen libremente.

El deporte no tiene que ser un obstáculo para nadie al contrario el deportista tiene que sentirse libre de hacer sus actividades, por este motivo se realizó este proyecto que cuenta con una mejor gestión brindando a clubs y deportistas estar seguros y libres de jugar.

Frontón-Tenis es una aplicación pensada en el deportista y mejorar su calidad del servicio, Esta aplicación le permite estar enterado sobre la disponibilidad de canchas deportivas hacer sus reservas, ver los últimos torneos del mes y tiene un apartado para colocar su desempeño, a su vez tiene un asistente virtual para el público en general en cual le irá asesorando en su visita y búsqueda sobre noticias, torneos, canchas para mejorar la experiencia de usuario.

1.1 Motivación

Este proyecto se desarrolló por experiencias vividas en primera persona, siendo un referente de muchos deportistas que somos discriminadas al momento de ir a jugar a canchas públicas por ser mujeres se considera que este deporte es más para el género masculino, siendo un retraso en el tiempo en el que vivimos, También por ser pocas las canchas para practicar frontón, suele estar muy lleno gastando tiempo de espera de entre 2 o 3 horas para que los anteriores jugadores terminen.

Con este proyecto quiero ayudar a muchos deportistas a tener a la mano una herramienta de búsqueda y ponerlos en contacto con clubs, torneos, deportistas y a la vez tener información actualizada para los amantes de los deportes.

En función de porque se realizó una aplicación web, es porque está orientada a todos los usuarios que puedan hacer uso de internet, ya que este sistema se adapta a cualquier dispositivo web o móvil. Esta aplicación tiene una parte pública que no necesariamente tienes que estar registrado para poder ver, esto facilita a que cualquier usuario tenga acceso a un deporte seguro y de calidad.

1.2 Objetivos

El principal objetivo es diseñar una aplicación web para gestionar actividades en canchas de frontón principalmente en instalaciones públicas de la comunidad de Madrid con expansión a diferentes provincias.

- Realizar una aplicación web para ayudar a la gestión de actividades en canchas de frontón.
- Proporcionar un buscador para público general y usuarios registrados, para encontrar una cancha por ciudad o provincia.
- Brindar al usuario la opción de poder registrarse para hacer uso de la aplicación interna.
- Proporcionar al usuario un listado de torneos que se están promocionando para participar.

- Ofrecer al usuario un sistema de reserva para seleccionar una fecha y hora para poder utilizar las instalaciones deportivas.
- El usuario podrá agregar su desempeño para tener un registro de sus puntos y juegos ganados.
- Ver desempeño mediante una gráfica en Google chart.
- Disponer de un asistente virtual que lo ayude en su búsqueda.

1.3 Plan de trabajo

Para llevar a cabo este proyecto se realizó un conjunto de tareas marcadas que fueron completadas progresivamente. Las principales etapas son:

- Estado del arte: En esta tarea se recopiló información de distintos medios para poder realizar este proyecto y se clasificó en base a su desarrollo, encontrándose con herramientas de gran usabilidad como Google charts, Inteligencia artificial, chatbot.
- Desarrollo del proyecto: En esta etapa definimos los requerimientos de la aplicación, arquitectura, modelo de datos y base de datos.
- Tecnologías y herramientas: Analizamos las nuevas tecnologías e instalamos para implementar nuestra aplicación.
- Implementación: Finalmente implementamos y programamos la aplicación y ajustamos el informe final seguidamente.



Figura 1-1. Plan de trabajo

Capítulo 2 - Estado del arte

En este capítulo se estudia y analiza el estado de arte en relación con el proyecto, con el fin de poner en contexto al lector y pueda ser de ayuda en su conceptualización del desarrollo que se llevará a cabo en los siguientes apartados.

2.1 Gestión de Actividades deportivas

En estos momentos la gestión deportiva tiene dos posibilidades, seguir como hasta ahora o cambiar. Este cambio debe producirse bajo el conocimiento de la realidad en cada contexto. No cabe duda, de que no se puede hablar de una única gestión, pero sí de una "buena gestión deportiva". Para ello, este cambio debe producirse por una combinación de diferentes acciones, siendo algunas de estas:

1. El conocimiento del valor del suelo público en instalaciones deportivas de cada municipio-lugar. Ya que este, nos dará la posibilidad del cambio, tanto del tipo de gestión que se esté realizando en las organizaciones deportivas, como los nuevos planteamientos de necesidades de nuevas instalaciones deportivas. Para dar forma a este cambio, posiblemente las administraciones deban vender y cambiar el uso del suelo para dar salida a los requerimientos de la sociedad y sus necesidades de práctica deportiva.

2. La formación de los recursos humanos de estas organizaciones, es el primer paso para poder entender cuáles son las acciones posibles para alcanzar la eficacia absoluta. El liderazgo junto con el entrenamiento del talento, deben intervenir en la gestión pública. "El gestor de las organizaciones deportivas, crea valor".

3. El trabajo en equipo entre arquitectos, gestores, usuarios, empresarios, junto con la investigación I+D+i, dentro de la perspectiva de la gestión, ayudarán a conseguir el gran cambio que se necesita.

La impresionante revolución que supuso para el deporte y la actividad deportiva la construcción de gran cantidad de instalaciones durante la década de los años 80, llevó consigo la necesidad de crear e implantar servicios deportivos que llevaran a cabo la labor de su gestión, así como la labor de contratar técnicos y crear programas de

actividades que completarán las posibilidades que ofrecían las nuevas instalaciones. El planteamiento inicial de rentabilizar socialmente las instalaciones construidas, así como el aumento de la oferta de práctica deportiva, derivó en un espectacular aumento de la actividad deportiva de los ciudadanos y cada vez más, las administraciones públicas van dando respuesta al reclamo que la ciudadanía viene haciendo para que se preste un mejor servicio. [1]



Figura 2-1. Gestión de la calidad

2.2 Google Charts API

Las API de Google, o interfaces de programación de aplicaciones, permiten a los desarrolladores integrar los servicios de Google con otros servicios. Estos servicios incluyen Funciones de búsqueda de Google, traducciones, YouTube, calendarios y mapas en sitios web. Los desarrolladores también pueden usar las API de Google para mejorar la inteligencia de aplicaciones telefónicas, como la recuperación de coordenadas geográficas o el cálculo de los tiempos de entrega o servicios de navegación. En este taller usaremos API de gráficos de Google.

Google Charts "le permite crear gráficos y aplicaciones de informes sobre datos estructurados y ayuda a integrarlos directamente en su sitio web" (Google Charts, Preguntas frecuentes). Los datos viven dentro del HTML. (página web) archivo que es descargado por el usuario cuando va a su sitio. Esto significa que todos los datos que entran en cada visualización son procesados localmente: no se transfieren datos a

Google. Los datos que se transfieren entre el archivo HTML y Google son las configuraciones de datos y gráficos. Estos se envían a través de un formato URL que especifica qué tipo de gráfico está creando y las opciones que desea aplicar a los datos en el archivo HTML. Ellos le dicen a su navegador cómo dibujar el gráfico en su página. Los gráficos se representan mediante HTML5/SVG y se pueden ver en navegadores web, teléfonos inteligentes, o tabletas.

Para comunicarse con la API de Google y crear sus gráficos, necesitará usar JavaScript. JavaScript es uno de los tres lenguajes de programación de la web. Están:

1. HTML define el contenido de una página, como texto, enlaces e imágenes.
2. CSS especifica el diseño de la página, como estilos y cómo HTML se mostrarán los elementos.
3. JavaScript programa el comportamiento de las páginas web, como qué sucede cuando un usuario se desplaza sobre un gráfico de barras o lo que sucede cuando un usuario hace clic en un elemento de la página web.

En este taller, tocaremos elementos selectos de estos idiomas que nos ayudarán a construir nuestros gráficos. No necesita experiencia en estos idiomas más allá de lo que se presentará. Vamos a confiar en el paquete de programación estadística R para ayudarnos a ir más allá de la brecha de conocimiento y pídale que escriba HTML, CSS y JavaScript para nosotros. [2]

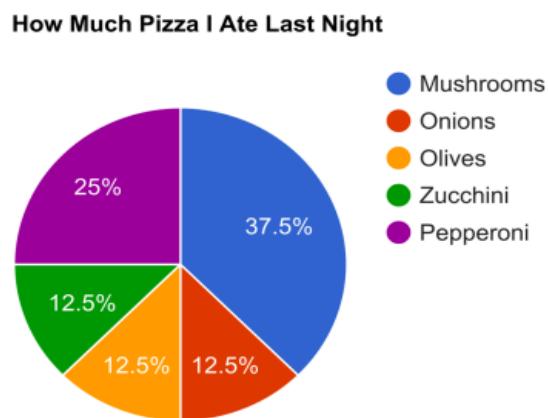


Figura 2-2. Google Charts Ejemplo

2.3 Inteligencia Artificial

Antes de definir el término Inteligencia Artificial, deberíamos definir qué entendemos por inteligencia. La Real Academia de la Lengua Española nos define la inteligencia como “Potencia intelectual: facultad de conocer, de entender o comprender”. Una vez definido esto, la inteligencia artificial se podría definir de la misma forma, pero aplicado a las máquinas. De esta forma, cuando nos dicen que una lavadora es inteligente tendríamos que preguntar qué conocimiento o comprensión tiene dicha lavadora del proceso que realiza. Como podemos observar, este término es muy ambiguo y se han aducido múltiples argumentos a favor y en contra de la inteligencia artificial en las máquinas.

Nosotros vamos a utilizar la definición de inteligencia artificial que pensamos que es más cercana a la realidad. La propuso Marvin Minsky, uno de los pioneros de la IA, y dice así: “La inteligencia artificial es la ciencia de construir máquinas para que hagan cosas que, si las hicieran los humanos, requeriría inteligencia”.

Podemos pensar en la IA como en aquella ciencia que incorpora conocimiento a los procesos o actividades que estos tengan éxito. Un ejemplo es el ajedrez. Es impensable que un computador evalúe todas las posibles jugadas del ajedrez. En vez de esto, se incorpora conocimiento en el proceso de búsqueda de la mejor jugada en forma de jugadas predefinidas o procedimientos de evaluación “inteligentes”.

Diversas son las áreas donde la IA se presenta en mayor o menor medida. A continuación, se comentan brevemente algunas de estas áreas:

- **Tratamiento de lenguajes naturales** Este campo se puede englobar aplicaciones que realicen traducciones entre idiomas, interfaces hombre-máquina que permitan interrogar una base de datos o dar órdenes a un sistema operativo, etc., de manera que la comunicación sea más amigable al usuario.
- **Sistemas expertos** En esta área están englobados aquellos sistemas donde la experiencia de personal cualificado se incorpora a dichos sistemas para conseguir deducciones más cercanas a la realidad.

- **Robótica** Navegación de robots móviles, control de brazos de robots, ensamblajes de piezas, etc.
- **Problema de percepción: visión y habla** Reconocimiento de objetos y del habla, detección de defectos en piezas por medio de visión, apoyo en diagnósticos médicos, etc.
- **Aprendizaje** Modelización de conductas para su posterior implantación en computadoras. [3]

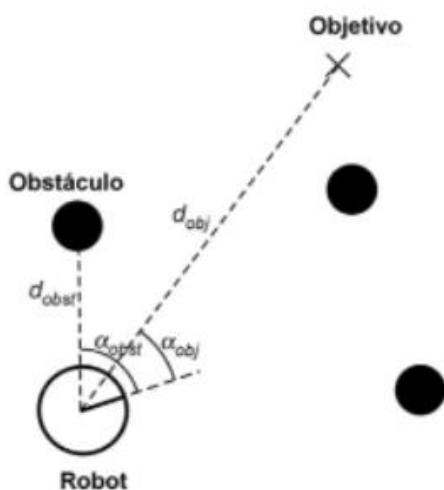


Figura 2-3. Problema de control de un robot móvil

2.4 Chatbot

En el nivel más básico, un chatbot es un programa informático que simula y procesa una conversación humana (ya sea escrita o hablada), lo que permite a los humanos interactuar con dispositivos digitales como si se estuvieran comunicando con una persona real. Los chatbots pueden ser tan simples como programas rudimentarios que responden a una consulta simple con una respuesta de una sola línea, o tan sofisticados como asistentes digitales que aprenden y evolucionan para brindar niveles crecientes de personalización a medida que recopilan y procesan información.

Probablemente hayas interactuado con un chatbot, lo sepas o no. Por ejemplo, está en su computadora investigando un producto y aparece una ventana en su pantalla que le pregunta si necesita ayuda. O tal vez va camino a un concierto y usa su

teléfono inteligente para solicitar un viaje a través del chat. O es posible que haya usado comandos de voz para pedir un café en la cafetería de su vecindario y recibió una respuesta que le indica cuándo estará listo su pedido y cuánto costará. Todos estos son ejemplos de escenarios en los que podría encontrarse con un chatbot.

Impulsados por IA, reglas automatizadas, procesamiento de lenguaje natural (NLP) y aprendizaje automático (ML), los chatbots procesan datos para brindar respuestas a solicitudes de todo tipo.

Hay dos tipos principales de chatbots.

- **Los chatbots orientados a tareas (declarativos)** son programas de un solo propósito que se enfocan en realizar una función. Usando reglas, NLP y muy poco ML, generan respuestas automatizadas pero conversacionales a las consultas de los usuarios. Las interacciones con estos chatbots son altamente específicas y estructuradas y son más aplicables a las funciones de soporte y servicio: piense en preguntas frecuentes sólidas e interactivas.
- **Los chatbots basados en datos y predictivos (conversacionales)** a menudo se denominan asistentes virtuales o asistentes digitales, y son mucho más sofisticados, interactivos y personalizados que los chatbots orientados a tareas. Estos chatbots son conscientes del contexto y aprovechan la comprensión del lenguaje natural (NLU), NLP y ML para aprender sobre la marcha. Aplican inteligencia predictiva y análisis para permitir la personalización en función de los perfiles de usuario y el comportamiento del usuario en el pasado. [4]

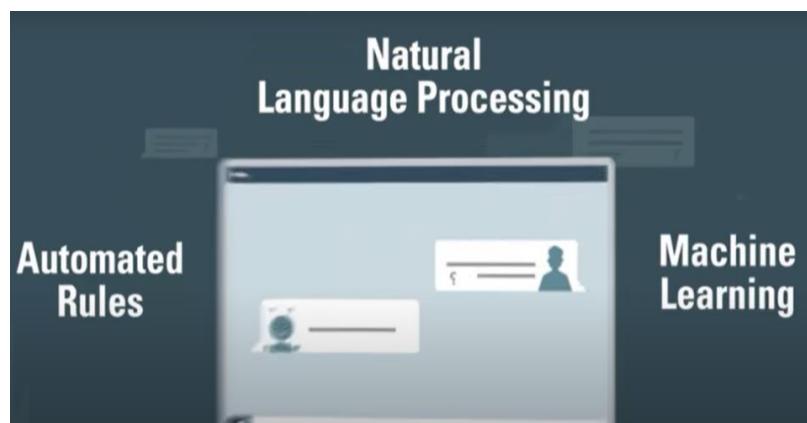


Figura 2-4. Captura del video explicativo de Oracle

2.5 Proyectos similares

A continuación, veremos las aplicaciones que hay en el mercado y que tienen tareas similares a este proyecto con distintas funcionalidades añadidas o limitadas por un presupuesto.

2.5.1 Runkeeper

Es una aplicación diseñada para todos los corredores y que cuenta con rutinas, desafíos mensuales y mucho más para ayudarte a llegar al siguiente nivel como deportistas. Fija objetivos de running y entrenamientos, registra tu progreso y comparte tus vivencias con la comunidad. [5]

2.5.2 Tennis Score Trancker

Esta herramienta posee un sistema marcador, que va a ayudar a registrar la puntuación de los partidos de tenis, tanto individuales como dobles. Además, tiene varias opciones de estadísticas y hasta tres modos de puntuación. [6]

2.5.3 Clupik

Es una App para administrar Clubs y tener una comunicación con los participantes y gestionar campeonatos, entrenamientos, pero el costo es de 39 euros mensuales con limitaciones [7]

Capítulo 3 - Desarrollo del proyecto

En este capítulo se describe cómo se llevó a cabo el desarrollo del proyecto, haciendo uso de las especificaciones de los requisitos, casos de usos y arquitectura del proyecto.

3.1 Especificación de Requisitos

Para llevar a cabo el desarrollo del proyecto, primero debemos tener especificados quienes son los actores que interactúan con el sistema y qué tareas son las que hace cada uno, en el siguiente apartado se detallan.

3.1.1 Especificación de Actores

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

AC1	Deportista
Versión	Nº 1
Dependencias	Administrador
Descripción	Este actor es el que hace uso de la aplicación para hacer sus reservas, gestión e inscripción en los torneos.
Comentarios	Actor Principal

Tabla 3-1. Actor principal Deportista

AC2	Club
Versión	Nº 1
Dependencias	Administrador
Descripción	Este Actor se Encarga de subir las Canchas y Torneos
Comentarios	Actor Principal

Tabla 3-2. Actor Principal Club

AC3	Administrador
Versión	Nº 1
Dependencias	Sistema
Descripción	Este Actor es el encargado de administrar y gestionar la aplicación
Comentarios	Actor Principal

Tabla 3-3. Actor Administrador

AC4	Público
Versión	Nº 1
Dependencias	Administrador
Descripción	Este actor puede hacer búsqueda de canchas y ver torneos
Comentarios	Actor Secundario

Tabla 3-4. Actor Secundario Público

3.1.2 Casos de uso

En este apartado podemos ver de manera gráfica los casos de uso del sistema y los actores principales y secundarios. En la figura 3-1. Caso de uso – deportista es nuestro actor principal en el cual está registrado en el sistema y tiene acceso a las distintas actividades. En la figura 3-2. Caso de uso - Club, es un segundo actor principal porque es el que ingresa al sistema para agregar canchas y torneos que luego el deportista puede ver su publicación. En la siguiente figura 3-3. Es nuestro administrador del general que tiene que gestionar a todos los actores involucrados dentro del sistema, figura 3-4. Es nuestro actor secundario por ser un actor público y limitado a solo visualizar torneos o canchas.

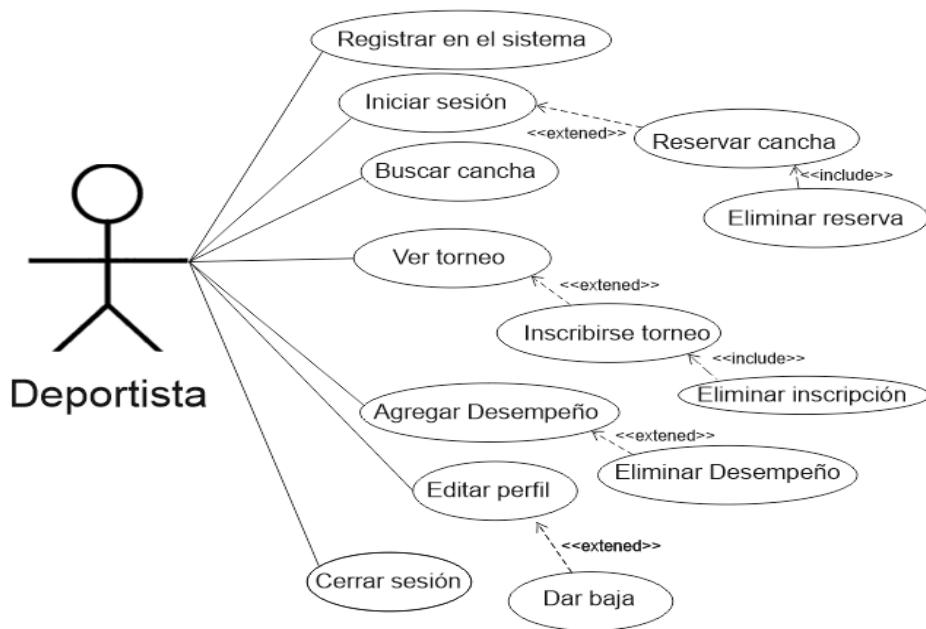


Figura 3-1. Caso de uso - Deportista

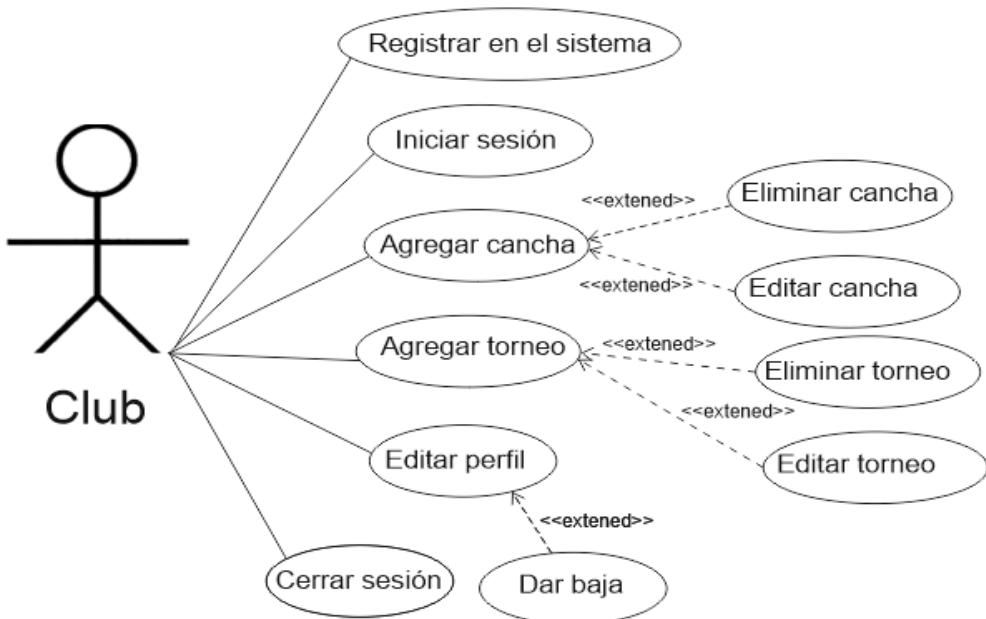


Figura 3-2. Caso de uso – Club

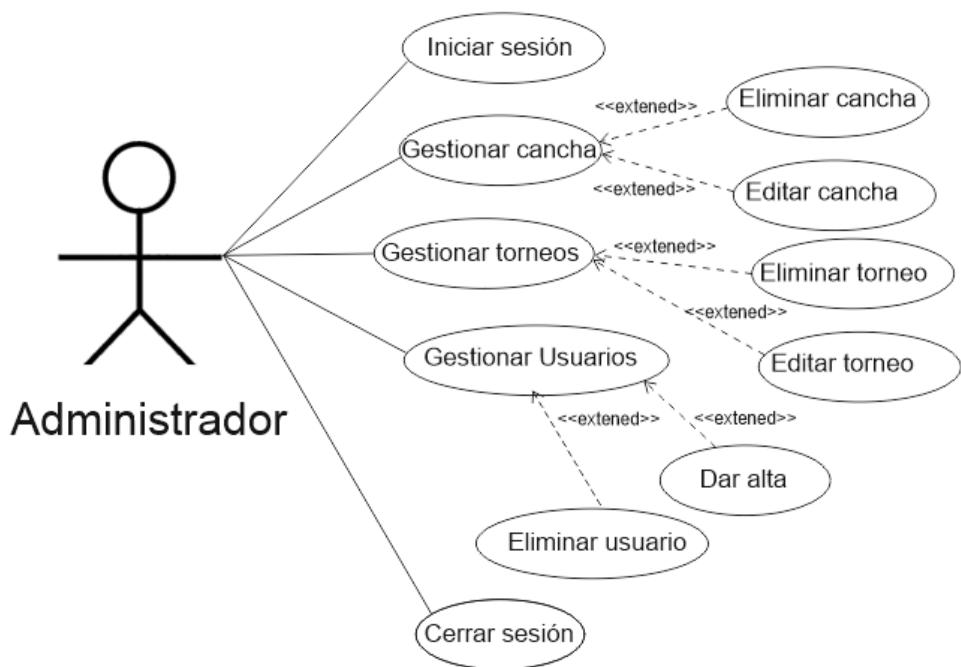


Figura 3-3. Caso de uso - Administrador

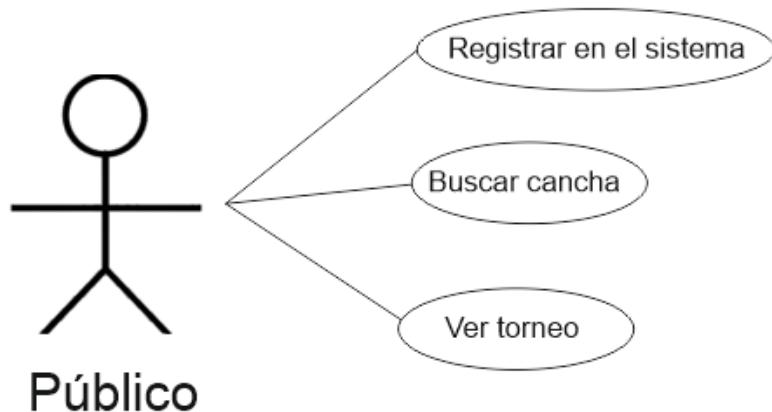


Figura 3-4. Caso de uso - Público

3.1.3 Especificación de casos de uso

En este apartado se describe cada uno de los casos de uso

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 le den de alta	
Salidas	Mensaje "Datos guardados dentro de 24 horas estará dado 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	Ingresar sus datos
	2	El Sistema valida que los datos tienen el formato correcto y que no tenga 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 le den de alta	
Salidas	El usuario ingresa 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 valida que los datos
	3	El Sistema le redireccionara al menú principal
Secuencia fallo	Paso	Acción
	1	Si lo datos de entrada son incorrectos o están vacíos se le mostrara un mensaje "Verifique sus Datos".
	2	Si no se puede ingresar se le mostrara 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	Ingresa hace una búsqueda
	2	Ingresa 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 mostrara con rojo
	2	No se mostrará la lista de horas
Importancia	Vital	
Estabilidad	Alta	

Figura 3-8. Caso de uso 4 - Reserva 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 muestra al usuario 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 valida 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 actualiza 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 valida 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 "Esta seguro 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 ingresa los datos de la nueva cancha
	2	El sistema valida los datos y los inserta
	3	El sistema le muestra su cancha
Secuencia fallo	Paso	Acción
	1	Datos incorrectos
Importancia	Vital	
Estabilidad	Alta	

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 valida 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 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 valida 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	
Postcondición Si éxito	Torneo 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 valida los datos
	3	El sistema muestra el torneo agregado
Secuencia fallo	Paso	Acción
	1	Si los datos son incorrectos o vacíos se le mostrara 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 eliminados	
Postcondición Si éxito	Torneo 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 valida 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 mostrara 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 editar un torneo	
Entradas	Datos Torneo	
Precondiciones	El usuario tiene que estar registrado	
Salidas	Datos editados	
Postcondición Si éxito	Torneo 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 valida los datos
	3	El sistema muestra el torneo actualizado
Secuencia fallo	Paso	Acción
	1	Si los datos son incorrectos o vacíos se le mostrara 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 editados, eliminados	
Postcondición Si éxito	Datos gestionados con éxito	
Postcondición Si fallo	Se muestra mensaje error	
Actores	AC3	
Secuencia normal	Paso	Acción
	1	El usuario selecciona Canchas que quiere gestionar, editar, eliminar
	2	El sistema valida los datos
	3	El sistema muestra la cancha actualizada
Secuencia fallo	Paso	Acción
	1	Si los datos son incorrectos o vacíos se le mostrara 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	Datos editados, eliminados	
Postcondición Si éxito	Datos 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 valida los datos
	3	El sistema muestra el torneo actualizado
Secuencia fallo	Paso	Acción
	1	Si los datos son incorrectos o vacíos se le mostrara 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	Usuarios editados, eliminados, dados de alta	
Postcondición Si éxito	Datos 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 valida 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 mostrara un mensaje
Importancia	Vital	
Estabilidad	Alta	

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

3.1.4 Requisitos funcionales

Este apartado muestra los requisitos funcionales del sistema que son declaraciones de los servicios que se prestará al usuario.

Código	REQF-01	Versión	0.1	Año	2022		
Fuente	Alicia Joy						
Prioridad	Alta		Módulo	Aplicación			
Descripción	El sistema debe permitirle al usuario tener una interfaz amigable para interactuar con el usuario.						
Observaciones	Ninguna						

Tabla 3-5. Requisito funcional 01

Código	REQF-02	Versión	0.1	Año	2022		
Fuente	Alicia Joy						
Prioridad	Alta		Módulo	Aplicación			
Descripción	El sistema debe permitirle al usuario poder registrarse en la aplicación						
Observaciones	Ninguna						

Tabla 3-6. Requisito funcional 02

Código	REQF-03	Versión	0.1	Año	2022		
Fuente	Alicia Joy						
Prioridad	Alta		Módulo	Aplicación			
Descripción	El sistema debe proveer autenticarse con sus credenciales para identificarse y poder ingresar						
Observaciones	Debe tener un usuario y contraseña						

Tabla 3-7. Requisito funcional 03

Código	REQF-04	Versión	0.1	Año	2022		
Fuente	Alicia Joy						
Prioridad	Alta		Módulo	Aplicación			
Descripción	El sistema debe permitir al usuario visualizar sus datos y su perfil						
Observaciones	Ninguna						

Tabla 3-8. Requisito funcional 04

Código	REQF-05	Versión	0.1	Año	2022		
Fuente	Alicia Joy						
Prioridad	Alta		Módulo	Aplicación			
Descripción	El sistema debe permitir al usuario poder agregar su desempeño						
Observaciones	Ninguna						

Tabla 3-9. Requisito funcional 05

Código	REQF-06	Versión	0.1	Año	2022		
Fuente	Alicia Joy						
Prioridad	Alta		Módulo	Aplicación			
Descripción	El sistema debe permitir al usuario poder hacer una búsqueda de canchas						
Observaciones	Ninguna						

Tabla 3-10. Requisito funcional 06

Código	REQF-07	Versión	0.1	Año	2022		
Fuente	Alicia Joy						
Prioridad	Alta		Módulo	Aplicación			
Descripción	El sistema debe permitir al usuario poder darse de baja						
Observaciones	Ninguna						

Tabla 3-11. Requisito funcional 07

Código	REQF-08	Versión	0.1	Año	2022		
Fuente	Alicia Joy						
Prioridad	Alta		Módulo	Aplicación			
Descripción	El sistema debe permitir al usuario poder eliminar su inscripción del torneo						
Observaciones	Ninguna						

Tabla 3-12. Requisito funcional 08

Código	REQF-09	Versión	0.1	Año	2022		
Fuente	Alicia Joy						
Prioridad	Alta		Módulo	Aplicación			
Descripción	El sistema debe permitir al usuario cerrar su sesión						
Observaciones	Ninguna						

Tabla 3-13. Requisito funcional 09

Código	REQF-10	Versión	0.1	Año	2022		
Fuente	Alicia Joy						
Prioridad	Alta		Módulo	Aplicación			
Descripción	El sistema debe permitir al usuario poder hacer una reserva de una cancha						
Observaciones	Debe seleccionar primero la cancha						

Tabla 3-14. Requisito funcional 10

Código	REQF-11	Versión	0.1	Año	2022		
Fuente	Alicia Joy						
Prioridad	Alta		Módulo	Aplicación			
Descripción	El sistema debe permitir al usuario eliminar su reserva de cancha						
Observaciones	Ninguna						

Tabla 3-15. Requisito funcional 11

Código	REQF-12	Versión	0.1	Año	2022		
Fuente	Alicia Joy						
Prioridad	Alta		Módulo	Aplicación			
Descripción	El sistema debe permitir al usuario interactuar con el chatbot						
Observaciones	Ninguna						

Tabla 3-16. Requisito funcional 12

3.1.5 Requisitos no funcionales

En este apartado se detalla los requisitos no funcionales en base a las propiedades que brinda el sistema tales como: Rendimiento, disponibilidad entre otros que se describen en las siguientes tablas.

Código	REQNF-01	Versión	0.1	Año	2022		
Fuente	Alicia Joy						
Prioridad	Alta		Módulo	Aplicación			
Descripción	El sistema debe garantizar la disponibilidad las 24h/7 de operatividad						
Observaciones	Ninguna						

Tabla 3-17. Requisito no funcional 01

Código	REQNF-02	Versión	0.1	Año	2022		
Fuente	Alicia Joy						
Prioridad	Alta		Módulo	Aplicación			
Descripción	El sistema debe garantizar la seguridad y confidencialidad de la información.						
Observaciones	Ninguna						

Tabla 3-18. Requisito no funcional 02

Código	REQNF-03	Versión	0.1	Año	2022		
Fuente	Alicia Joy						
Prioridad	Alta		Módulo	Aplicación			
Descripción	El sistema debe garantizar el tiempo de respuesta en menos de 3 segundos al buscar información o hacer consultas.						
Observaciones	Buscadores y respuesta chatbot						

Tabla 3-19. Requisito no funcional 03

Código	REQNF-04	Versión	0.1	Año	2022		
Fuente	Alicia Joy						
Prioridad	Alta		Módulo	Aplicación			
Descripción	El sistema debe soportar el manejo de gran cantidad de información durante sus consultas.						
Observaciones	Ninguna						

Tabla 3-20. Requisito no funcional 04

Código	REQNF-05	Versión	0.1	Año	2022				
Fuente	Alicia Joy								
Prioridad	Alta	Módulo		Aplicación					
Descripción	El sistema debe funcionar en distintos tipos de sistemas operativos y plataformas.								
Observaciones	Ninguna								

Tabla 3-21. Requisito no funcional 05

Código	REQNF-06	Versión	0.1	Año	2022				
Fuente	Alicia Joy								
Prioridad	Alta	Módulo		Aplicación					
Descripción	El sistema debe permitir almacenar grandes volúmenes de datos								
Observaciones	Ninguna								

Tabla 3-22. Requisito no funcional 06

Código	REQNF-07	Versión	0.1	Año	2022				
Fuente	Alicia Joy								
Prioridad	Alta	Módulo		Aplicación					
Descripción	El sistema debe mostrar opciones y módulos de administración								
Observaciones	Eliminar, editar datos								

Tabla 3-23. Requisito no funcional 07

Código	REQNF-08	Versión	0.1	Año	2022				
Fuente	Alicia Joy								
Prioridad	Alta	Módulo		Aplicación					
Descripción	El sistema debe permitir un alto grado de estabilidad en el sistema								
Observaciones	Ninguna								

Tabla 3-24. Requisito no funcional 08

Código	REQNF-09	Versión	0.1	Año	2022				
Fuente	Alicia Joy								
Prioridad	Alta	Módulo		Aplicación					
Descripción	El sistema debe permitir un alto grado de operatividad								
Observaciones	Ninguna								

Tabla 3-25. Requisito no funcional 09

3.2 Arquitectura del proyecto

Este proyecto se optó por una arquitectura Cliente-Servidor para poder llegar a cualquier usuario independientemente de que sistema operativo o plataforma utilice para ingresar, reduciendo al mínimo los requisitos de la plataforma en la que se conecte. [8]

La arquitectura cliente servidor tiene dos partes claramente diferenciadas, por un lado, la parte del servidor y por otro la parte de cliente o grupo de clientes donde lo habitual es que un servidor sea una máquina bastante potente con un hardware y software específico que actúa de depósito de datos y funcione como un sistema gestor de base de datos o aplicaciones.

Para entender este modelo vamos a nombrar y definir a continuación algunos conceptos básicos que lo conforman.

- **Red:** Una red es un conjunto de clientes, servidores y base de datos unidos de una manera física o no física en el que existen protocolos de transmisión de información establecidos.
- **Cliente:** El concepto de cliente hace referencia a un demandante de servicios, este cliente puede ser un ordenador como también una aplicación de informática, la cual requiere información proveniente de la red para funcionar.
- **Servidor:** Un servidor hace referencia a un proveedor de servicios, este servidor a su vez puede ser un ordenador o una aplicación informática la cual envía información a los demás agentes de la red.
- **Base de datos:** Son bancos de información ordenada, categorizada y clasificada que forman parte de la red, que son sitios de almacenaje para la utilización de los servidores y también directamente de los clientes. [9]

En la siguiente figura 3-24. Se puede observar la arquitectura del proyecto, en este podemos observar a nuestro servidor web en Apache PHP y nuestra base de datos en MySQL donde almacenamos nuestros datos, también intervienen Dialogflow y Google charts.

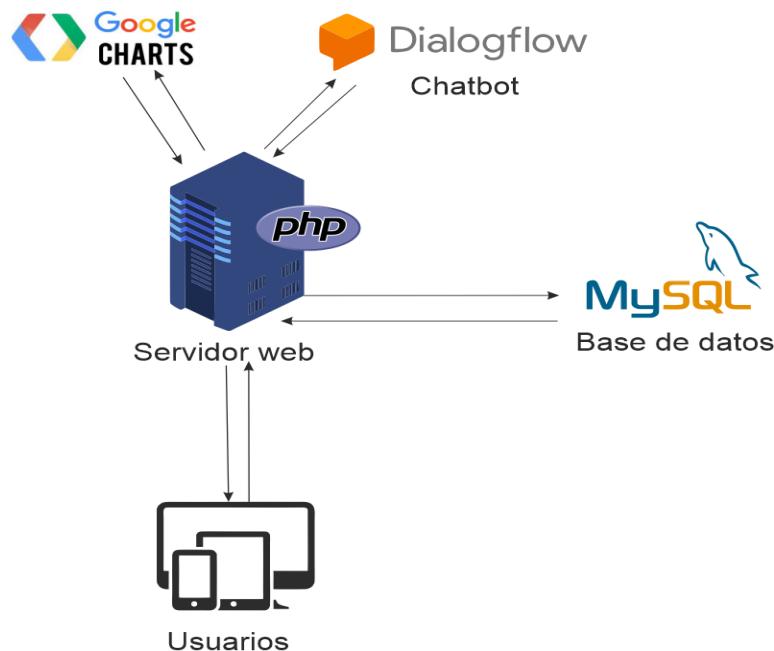


Figura 3-24. Arquitectura del proyecto

En la siguiente figura podemos ver la arquitectura de nuestro chatbot en Dialogflow.

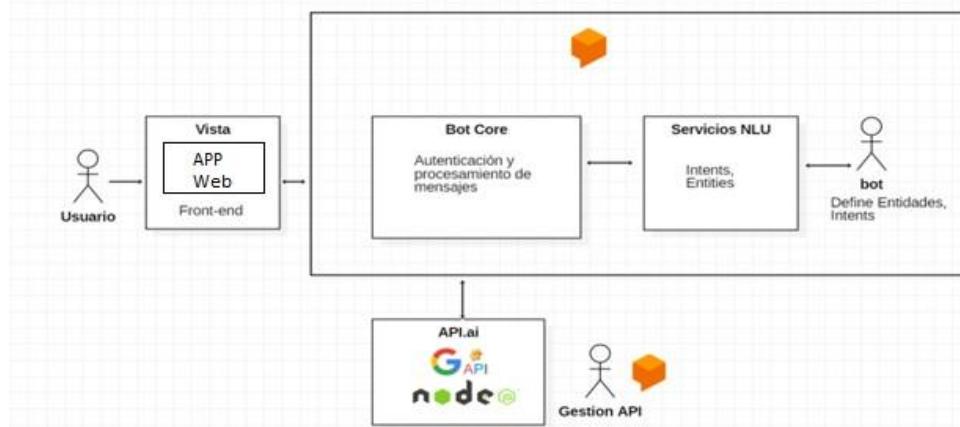


Figura 3-25. Arquitectura chatbot

3.3 Modelo de Datos

En este apartado se revisará la base de datos y los componentes que la componen tal vemos en la figura 3-26.

3.3.1 Modelo Entidad-Relación

En esta figura vemos claramente la estructura de nuestra base de datos, consta de 9 tablas que están relacionadas.

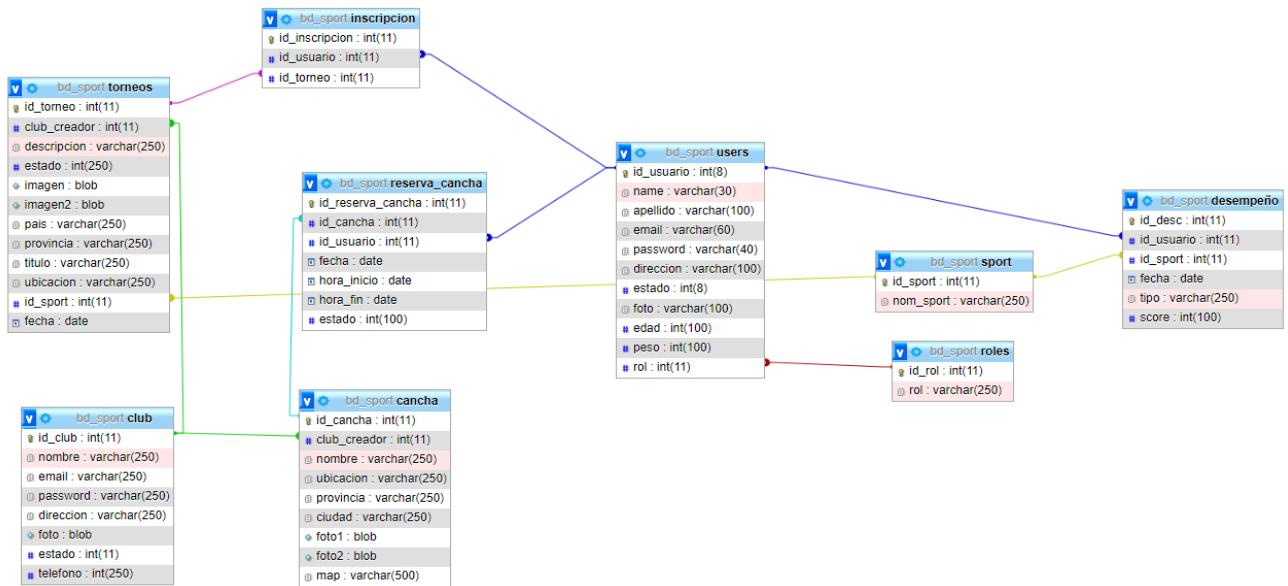


Figura 3-26. Modelo Entidad-Relación del proyecto

3.3.2 Modelo lógico

Este modelo nos muestra sus entidades y relaciones que interactúan dentro del sistema.

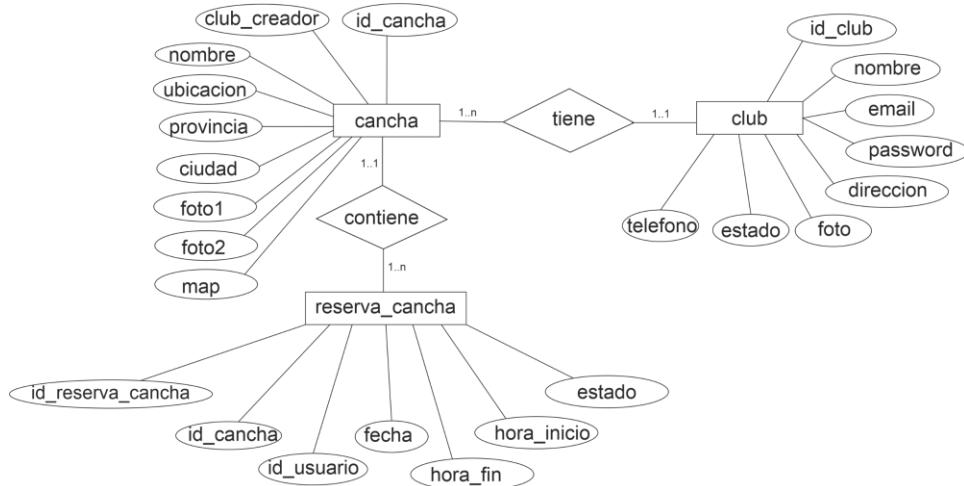


Figura 3-27. Modelo lógico Cancha - Reserva

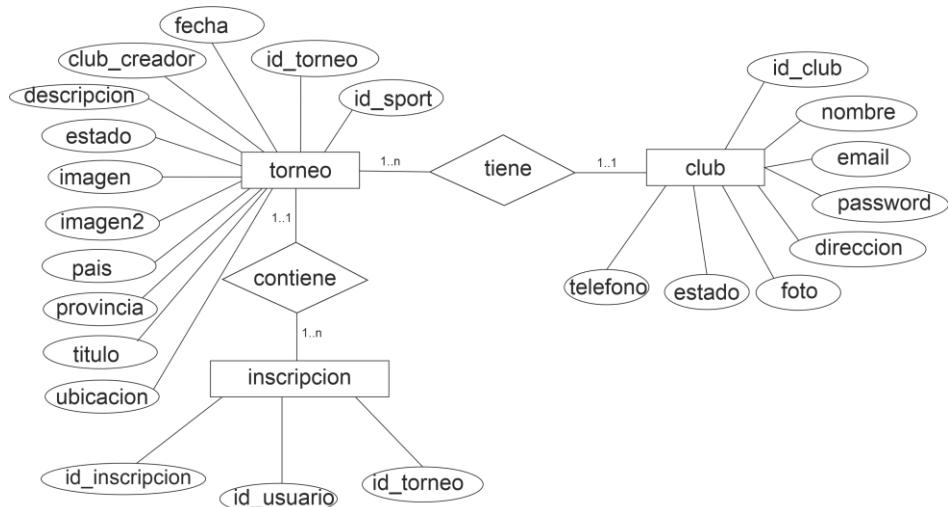


Figura 3-28. Modelo lógico Torneo - inscripción

3.3.3 Base de datos

Para almacenar los datos que ingresan al sistema se ha utilizado una base de datos relacional en MySQL y está diseñada con nueve tablas para almacenar toda la información ingresada por los usuarios.

En los siguientes apartados se describe cada una de ellas.

3.3.3.1 Tabla users

- **Descripción:** En esta tabla se guardan todos los datos personales del usuario y con este se puede registrar en cualquier torneo o hacer su reserva en cancha.
- **Columnas:**
 - id_usuario: Id con el cual cada usuario se identifica únicamente dentro del sistema.
 - name: nombre del usuario
 - apellido: apellido del usuario
 - email: correo electrónico por el cual podemos notificar cualquier incidencia al usuario.
 - password: contraseña para poder ingresar al sistema
 - dirección: dirección física del usuario
 - estado: con este campo podemos saber si está activo o baja para el administrador pueda activar
 - foto: es una foto del usuario registrado
 - edad: la edad del usuario
 - peso: el peso que tiene actualmente el deportista
 - rol: para saber si es un usuario o un administrador

3.3.3.2 Tabla cancha

- **Descripción:** En esta tabla se guardan los datos de la cancha que fue creada por un club para luego poder hacer las reservas.
- **Columnas:**
 - id_cancha: Id con el cual cada cancha se identifica únicamente dentro del sistema.
 - club_creador: identificador que se guarda de cada club que lo creo
 - nombre: nombre por el cual se registra el nombre de cada 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: es la foto de logo del club
 - foto2: es una foto de la cancha para dar mejor visibilidad al usuario
 - map: ubicación para Google map

3.3.3.3 Tabla reserva_cancha

- **Descripción:** En esta tabla se guardan los datos relacionados a la reserva de cada cancha que fue creada por un club.
- **Columnas:**
 - id_reserva_cancha: Id con el cual cada reserva se identifica únicamente dentro del sistema.
 - id_cancha: identificador que se guarda de cada cancha de la cual se está haciendo la reserva.
 - Id_usuario: el identificador de cada usuario para guardar la reserva.
 - fecha: es la fecha en la cual quiere hacer su reserva

- hora_inicio: la hora en la que inicia su juego.
- hora_fin: la hora que finaliza su juego.
- estado: es para indicar en qué estado se encuentra la reserva, libre o reservado.

3.3.3.4 Tabla club

- **Descripción:** En esta tabla se guardan los datos del club el cual es responsable de crear la cancha y los torneos.
- **Columnas:**
 - id_club: Id con el cual cada club se identifica únicamente dentro del sistema.
 - name: nombre del club
 - email: correo electrónico por el cual podemos notificar cualquier incidencia al usuario.
 - password: contraseña para poder ingresar al sistema
 - dirección: dirección física del club
 - estado: con este campo podemos saber si está activo o de baja para que el administrador pueda activar.
 - foto: es una foto del club
 - telefono: el número telefónico del club

3.3.3.5 Tabla torneos

- **Descripción:** En esta tabla se guardan los datos del torneo y este fue creado por el club.
- **Columnas:**
 - id_torneo: Id con el cual cada torneo se identifica únicamente dentro del sistema.
 - club_creador: identificador del club que lo creo

- descripcion: una descripción en la cual detalla el torneo ejemplo: quienes participan, invitación.
- estado: si está público o privado
- imagen: imagen publicitaria
- imagen2: imagen opcional, del local, deportistas.
- pais: es el país donde se realiza el torneo
- provincia: donde se realiza el torneo
- titulo: 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.

3.3.3.6 Tabla inscripción

- **Descripción:** En esta tabla se guardan los datos de la inscripción a cada torneo por parte de los deportistas.
- **Columnas:**
 - id_inscripcion: Id con el cual cada inscripción por parte de cada participante se está registrando.
 - id_usuario: identificador del usuario que se registró.
 - id_torneo: el identificador del torneo.

3.3.3.7 Tabla sport

- **Descripción:** En esta tabla se guardan los tipos de deportes frontón, frontenis, pelota vasca.
- **Columnas:**
 - id_sport: identificador de cada deporte.
 - nom_sport: nombre de cada deporte.

3.3.3.8 Tabla roles

- **Descripción:** En esta tabla donde se guardan los roles, usuario o administrador.
- **Columnas:**
 - id_rol: identificador único del rol.
 - nom_sport: nombre del rol, usuario o administrador.

3.3.3.9 Tabla desempeño

- **Descripción:** En esta tabla donde el registro de puntuación de cada deportista, su desempeño.
- **Columnas:**
 - id_desc: identificador único de cada desempeño.
 - id_usuario: identificador del usuario.
 - id_sport: identificador de cada deporte.
 - fecha: es la fecha en la que jugó y quiere registrar como desempeño.
 - tipo: identifica si su juego fue “Individual o Parejas”.
 - score: los puntos que hizo en el juego.

Capítulo 4 - Tecnologías y Herramientas

En este Capítulo se explica sobre las tecnologías que se utilizan en este proyecto y las herramientas que nos hicieron de utilidad con la implementación.

4.1 Tecnologías web

En los siguientes apartados se describe cada las tecnologías que instalamos e utilizamos para implementar y programar el sistema.

4.1.1 PHP

PHP (acrónimo recursivo de PHP: Hypertext Preprocessor) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML.

En lugar de usar muchos comandos para mostrar HTML (como en C o en Perl), las páginas de PHP contienen HTML con código incrustado. El código de PHP está encerrado entre las etiquetas especiales de comienzo y final <?php y ?> que permiten entrar y salir del "modo PHP".

Lo que distingue a PHP de algo del lado del cliente como Javascript es que el código es ejecutado en el servidor, generando HTML y enviándolo al cliente. El cliente recibirá el resultado de ejecutar el script, aunque no se sabrá el código subyacente que era. El servidor web puede ser configurado incluso para que procese todos los ficheros HTML con PHP, por lo que no hay manera de que los usuarios puedan saber qué se tiene debajo de la manga.

Lo mejor de utilizar PHP es su extrema simplicidad para principiantes, pero a su vez ofrece muchas características avanzadas para los programadores profesionales. No sienta miedo de leer la larga lista de características de PHP. [8]

4.1.2 MySQL

MySQL es la base de datos de código abierto más popular del mundo. Con su rendimiento comprobado, confiabilidad y facilidad de uso, MySQL se ha convertido en la opción de base de datos líder para aplicaciones basadas en la web, utilizadas por propiedades web de alto perfil como Facebook, Twitter, YouTube, Yahoo! y muchas más.

Oracle impulsa la innovación de MySQL, ofreciendo nuevas capacidades para impulsar la próxima generación de aplicaciones web, en la nube, móviles e integradas. Su alta disponibilidad a latencias y robustez hacen que sea una base de datos más confiable para este proyecto. [9]

4.1.3 Xampp

Es uno de los servidores web apache más utilizados en el mundo. El objetivo de XAMPP es construir una distribución fácil de instalar para que los desarrolladores entren en el mundo de Apache. Para que sea conveniente para los desarrolladores, XAMPP está configurado con todas las características activadas. En el caso de uso comercial, eche un vistazo a las licencias de productos, desde el punto de vista XAMPP el uso comercial también es gratuito. Actualmente hay distribuciones para Windows, Linux y OS X.

XAMPP es una compilación de software libre (comparable a una distribución de Linux), es gratuita y es libre de copiar bajo los términos de la Licencia Pública General de GNU. [10]

4.1.4 Google Chart API

Google Charts proporciona una manera perfecta de visualizar los datos en sitios web. Desde gráficos de líneas simples hasta mapas de árboles jerárquicos complejos, la galería de gráficos proporciona una gran cantidad de tipos de gráficos listos para usar.

La forma más común de usar Google Charts es con JavaScript simple que incrustas en tu página web. Carga algunas bibliotecas de Gráficos de Google, enumera los datos que se van a graficar, selecciona opciones para personalizar su gráfico y,

finalmente, crea un objeto de gráfico con un identificador que elija. Luego, más adelante en la página web, crea un <div> con esa identificación para mostrar el gráfico de Google.

Los gráficos se representan utilizando la tecnología HTML5 / SVG para proporcionar compatibilidad entre navegadores (incluido VML para versiones anteriores de IE) y portabilidad multiplataforma a iPhones, iPads y Android. Sus usuarios nunca tendrán que meterse con complementos o cualquier software. Si tienen un navegador web, pueden ver sus gráficos. [11]

4.1.5 Node.Js

Node.js está diseñado para crear aplicaciones de red escalables. En el siguiente ejemplo de "hola mundo", muchas conexiones se pueden manejar simultáneamente. En cada conexión, se activa la devolución de llamadas, pero si no hay trabajo por hacer, Node.js se suspenderá.

Node.js es similar en diseño e influenciado por sistemas como Ruby's Event Machine y Python's Twisted. Node.js lleva el modelo de eventos un poco más allá. Presenta un bucle de eventos como una construcción en tiempo de ejecución en lugar de como una biblioteca. [12]

```
const http = require('http');

const hostname = '127.0.0.1';
const port = 3000;

const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('Hello World');
});

server.listen(port, hostname, () => {
  console.log(`Server running at http://${hostname}:${port}/`);
});
```

Figura 4-1. Ejemplo código

4.1.6 Express.js

Es el marco web más popular para Node.js. Está diseñado para crear aplicaciones web y API y se ha llamado el marco de servidor estándar de facto para Node.js.

Crear un backend desde cero para una aplicación en Node.js puede ser tedioso y llevar mucho tiempo. Desde la configuración de puertos hasta los controladores de rutas, escribir todo el código repetitivo se aleja de lo que realmente importa, que es escribir la lógica de negocios para una aplicación. Mediante el uso de marcos web como Express.js, los desarrolladores pueden ahorrar tiempo y centrarse en otras tareas importantes. [13]

4.1.7 Dialogflow

Es una plataforma con comprensión del lenguaje natural que te facilita el diseño de una interfaz de usuario de conversación y su integración a tu aplicación para dispositivos móviles, aplicaciones web, dispositivos, bots, sistemas de respuesta de voz interactiva y más. Con Dialogflow, puedes proporcionar nuevas y atractivas formas para que los usuarios interactúen con tu producto.

Dialogflow puede analizar múltiples tipos de entradas de tus clientes, incluidas entradas de texto o audio (como las de un teléfono o una grabación de voz). También puede responder a tus clientes de varias maneras, ya sea a través de texto o con voz sintética.

Ofrece a los usuarios interactuar con un asistente virtual llamado agentes los cuales pueden resolver problemas y responder preguntas en base a su aprendizaje al hacer uso de Inteligencia artificial y lenguaje natural. [14]

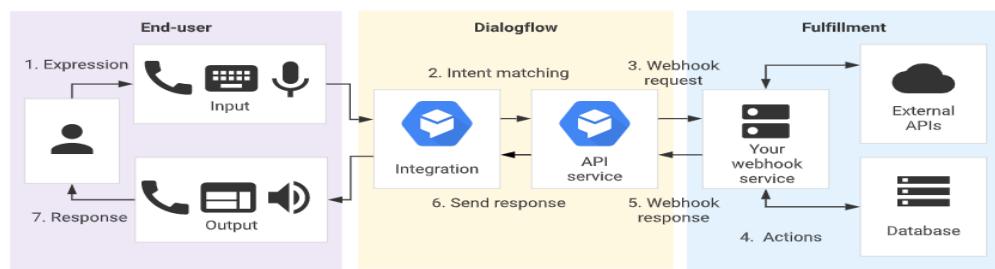


Figura 4-2. Ejemplo flujo Dialogflow

Capítulo 5 - Implementación

En este capítulo explicaremos la implementación y programación de nuestra aplicación, pero lo dividiremos en dos apartados aplicación web y chatbot.

El diseño de este proyecto es Cliente-Servidor el cual está implementado en PHP mediante un servidor Apache en Xampp y para la gestión de la base de datos utilizamos MySQL en Phpmyadmin para gestionar los datos.

En la figura 5-1 se puede apreciar el diagrama de despliegue en el cual se muestra al usuario y sus respectivas acciones dentro de la aplicación, servidor y el administrador de base de datos.

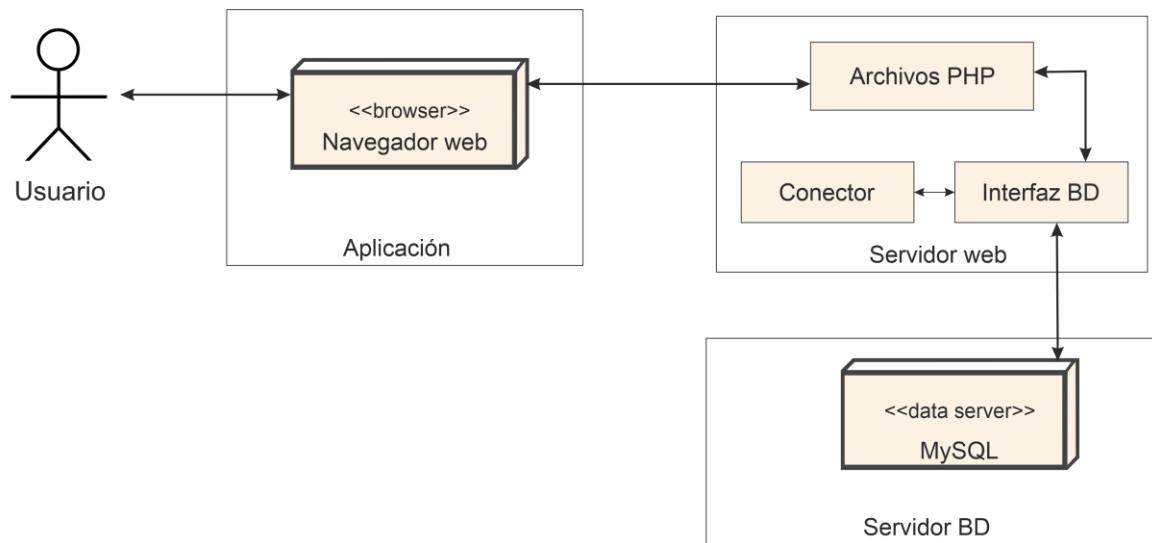


Figura 5-1. Diagrama de despliegue

5.1 Aplicación web

Esta aplicación cuenta con un diseño amigable para que el usuario pueda ingresar y navegar de una forma amigable e intuitiva. En el diseño de esta aplicación se tomó en cuenta el ámbito deportivo y se definió los colores en función al deporte.

En la aplicación interactúan 4 actores (público, el deportista, club y administrador) se irán describiendo en los siguientes apartados, el administrador será descrito en el apéndice C, por su extensión en los primeros, en el apéndice A se explica a detalle el manual de usuario y en el apéndice B el manual de instalación que se debe seguir para poder instalar.

5.1.1 Página Inicio

La primera interfaz nos encontramos con la página de inicio en esta se encuentra el menú principal y secciones para informar de manera didáctica al usuario cuales son las actividades de interés dentro de la aplicación, para el diseño se utilizó una plantilla [17] la cual adaptamos a nuestras necesidades con el objetivo brindar un diseño amigable y responsivo la cual hecha en Html y Css y Bootstrap.

En la figura 5-2 vemos nuestra aplicación y nuestro chatbot en la parte inferior

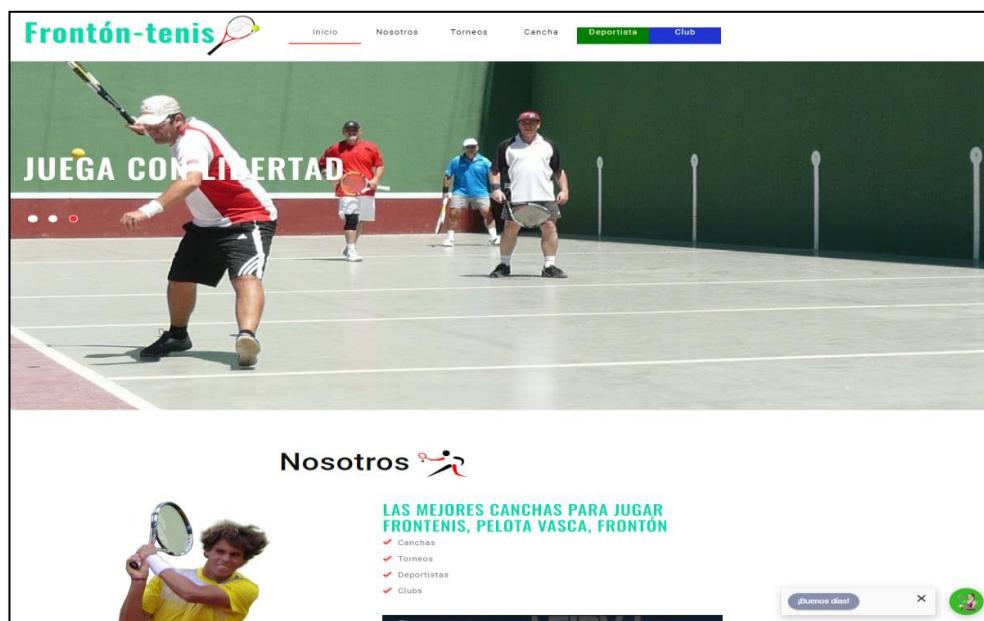


Figura 5-2. Página de inicio

5.1.2 Login y registro

En este apartado se realizó el registro (figura 5-3) e iniciar sesión (figura 5-4). El usuario que quiere hacer uso de la aplicación tiene que llenar un registro el cual se activará su sesión cuando el administrador le de alta, cuando finaliza su registro se le indica al usuario que tiene que esperar 24 horas para que su cuenta esté activa.



Figura 5-3. Pantalla – Registro deportista



Figura 5-4. Pantalla - Iniciar sesión

En la figura 5-5. podemos ver el código de registro como deportista, para hacer uso del sistema. Dentro del código podemos apreciar las validaciones que se hace a los campos cuando ingresa el usuario.

```
<?php
//revisa la sesion
session_start();
if(isset($_SESSION['usr_id'])) {
    header("Location: perfil.php");
}
include_once '../controller/dbconnect.php';
$error = false;
if (isset($_POST['signup'])) {
    $name = mysqli_real_escape_string($con, $_POST['name']);
    $email = mysqli_real_escape_string($con, $_POST['email']);
    $password = mysqli_real_escape_string($con, $_POST['password']);
    $cpassword = mysqli_real_escape_string($con, $_POST['cpassword']);
    $terminosycond = mysqli_real_escape_string($con, $_POST['terminosycond']);

    //Nombre sólo puede contener caracteres alfabéticos y espacio (esto varia segun requerimiento)
    if (!preg_match("/^[\wA-Z ]+$/i", $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 (!$error) {
        if(mysqli_query($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 {
            //Error si no se logra ingresar los datos con exito
            $errormsg =
                '<div class="alert alert-danger alert-dismissible fade in">
                    <a href="#" class="close" data-dismiss="alert" aria-label="close">&times;</a>
                    <strong>Error de registro!</strong> Verifica tus datos.
                </div>
            ';
        }
    }
}
?>
```

Figura 5-5. Registrar en el sistema

En esta figura 5-6 podemos ver el código para iniciar sesión validando el estado del usuario si el estado es 1 significa que está activo y puede ingresar, pero si no, se le muestra un mensaje indicando que todavía su cuenta está desactivada.

```
<?php
session_start();

if(isset($_SESSION['usr_id'])!="") {
    header("Location: perfil1.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: perfil1.php");
        }else{
            //si la cuenta todavía no está dado de alta por el usuario recibe un mensaje.
            $errormsg = "Esta cuenta está desactivada";
        }
    } else {
        $errormsg = "Revisa los datos!!!";
    }
}
?>
```

Figura 5-6. Iniciar sesión

En la figura 5-7 podemos ver el cierre de nuestra sesión si decidimos salir de la aplicación.

```
<?php
session_start();

if(isset($_SESSION['usr_id'])) {
    session_destroy();
    unset($_SESSION['usr_id']);
    unset($_SESSION['usr_name']);
    header("Location: login.php");
} else {
    header("Location: login.php");
}
?>
```

Figura 5-7. Cerrar sesión

5.1.3 Menú principal

En la figura 5-8 vemos la primera vista del deportista una vez ingresó con éxito, en esta se encuentra el menú de opciones, perfil, torneos, desempeño y canchas, cuando navegue internamente este menú pasará a la parte superior.



Figura 5-8. Menú principal deportista

```
<!-- Página 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>
        <div class="about-top">
            <div class="col-md-4 about-img-agile">
                <div style=" margin-left: 15px;">
                    <h3>Menú</h3><br>
                    <a href="perfildp.php"></a><br>
                    <a href="torneos.php"></a><br>
                    <a href="desempeño.php"></a><br>
                    <a href="canchas.php"></a><br>
                </div>
            </div>
            <div class="col-md-6 about-wel">
                <div class="col-md-6 col-md-offset-4 well">
                    <br><br>
                </div>
            </div>
        </div>
    </div>
<!-- //Página principal para deportista -->
```

Figura 5-9. Código menú principal

5.1.4 Perfil de usuario

En la figura 5-10 vemos la vista de perfil de usuario en el cual podemos apreciar los datos personales del usuario y su foto, estos pueden editarse en separado, también cuenta con el botón de *Dar de baja* si el usuario quiere eliminar su cuenta.

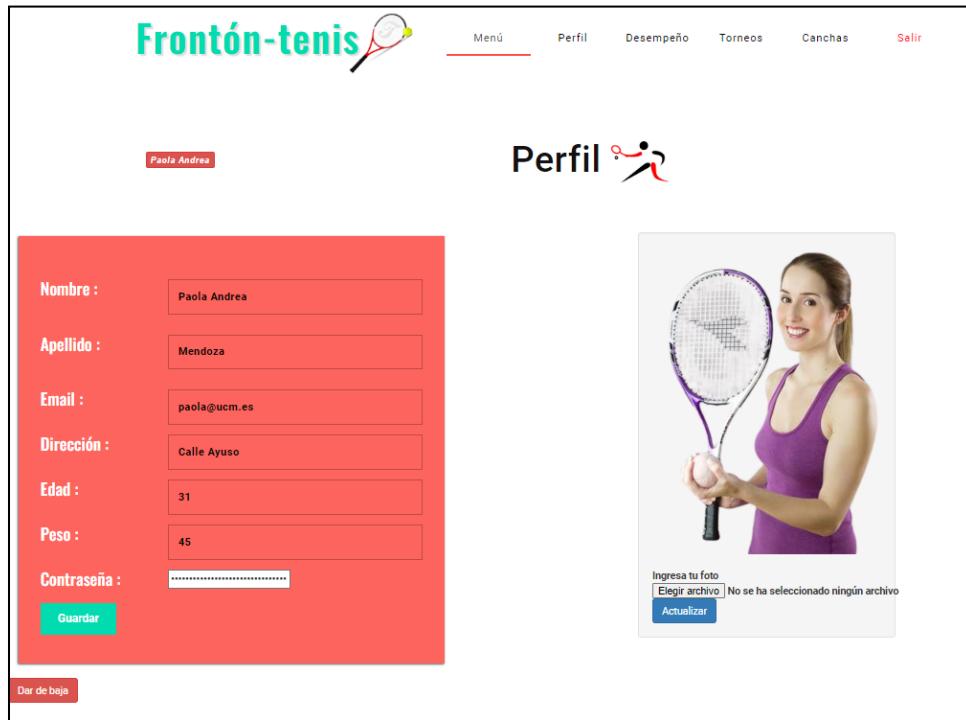


Figura 5-10. Perfil deportista

```
<?php
include '../controller/dbconnect.php';
session_start();
$id1 = $_SESSION['usr_id'];
$consulta = mysqli_query($con, "SELECT * FROM users WHERE id_usuario = '$id1'");
$valores = mysqli_fetch_array($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]'";
    $query_run =mysqli_query($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 5-11. Código - Perfil deportista

En la figura 5-12 vemos el perfil de club tiene el mismo diseño y el código es muy similar al del usuario por lo que se puede revisar en la carpeta view archivo perfil-club.php seguidamente en la figura 5-13 podemos ver el código Dar de baja.

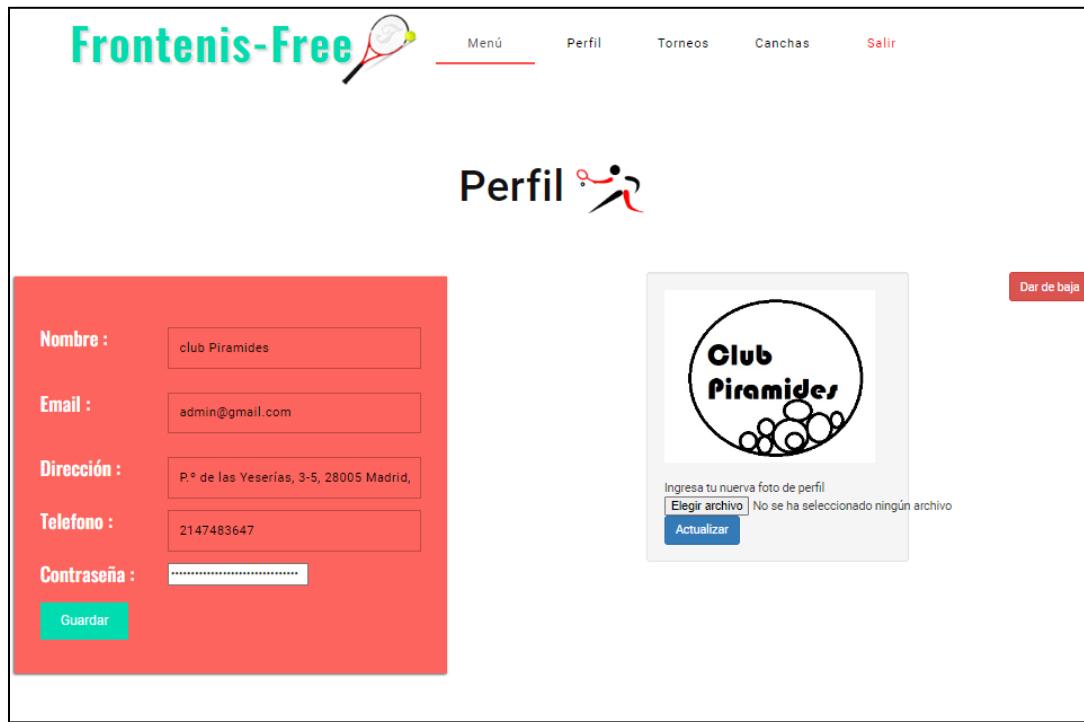


Figura 5-12. Perfil club

```
<?php
include '../controller/dbconnect.php';
$idcuenta = $_GET['id_usuario'];
$sql = "DELETE FROM users where id_usuario like $idcuenta";
$res =mysqli_query($con,$sql);
if(!$res){
    echo "Se dio de baja su cuenta";
}
else{
    header("Location:../view/login.php");
}
?>
```

Figura 5-13. Código - Dar de baja

5.1.5 Desempeño

En la siguiente figura 5-14 tenemos la vista desempeño en la cual el deportista puede ingresar sus puntos que hace en cada juego para llevar un control histórico, en esta vista también se implementó una gráfica que hace uso de Google chart el cual muestra el porcentaje por fecha y puntajes llamado score.

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



Figura 5-14. Desempeño deportista

```

<?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 5-15. Agregar Desempeño

El siguiente código corresponde a la implementación de Google chart que se encuentra en la carpeta view archivo desempeño.php para más información puede verse en la siguiente referencia [18].

```

<!--/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 5-16. Código - Google Chart

```

<div class="col-md-6 agileits_updates_grid_right">
<!-- Grafico Google Charts-->
|   <div id="piechart_3d" style="width: 900px; height: 500px;"></div>
<!-- //Grafico Google Charts-->
|   </div>
</div>

```

Figura 5-17. Código – Gráfica

El siguiente código corresponde a eliminar desempeño

```
<?php
include '../controller/dbconnect.php';
$id_desc = $_GET['id_desc'];
$sql = "DELETE FROM desempeño where id_desc like $id_desc";
$res = mysqli_query($con,$sql);
if(!$res){
    echo "No se Elimino";
}
else{
    header("Location:../view/desempeño.php");
}
?>
```

Figura 5-18. Código - Eliminar Desempeño

5.1.6 Torneos

En la siguiente figura podemos ver la vista de torneo, en esta el deportista puede visualizar los torneos subidos por el club, también cuenta con un listado de los torneos en los cuales está inscrito referente a la figura 5-21.

The screenshot shows the 'Frontenis-Free' website interface. At the top, there's a navigation bar with links: Menú, Perfil, Desempeño, Torneos, Canchas, and Salir. The 'Torneos' link is highlighted. The main content area is titled 'Torneos' with a small tennis player icon. It features two cards for tournaments:

- Torneo Femenino 2022** (España)
Ver publicación
- Segundo Torneo de Frontennis** (España)
Ver publicación

Below this, a blue bar contains the text 'Lista Torneos Inscritos'. Underneath is a search form with 'Agrupar de' set to '5' and a 'Resultados' dropdown. There's also a 'Buscar:' input field and an 'Opciones' button. A table lists the registered tournaments:

Nombre	Fecha	Ubicación	Opciones
Segundo Torneo de Frontennis	2022-07-30	Pº de las Yeserías, 3-5, 28005	Eliminar
Torneo Femenino 2022	2022-07-23	Urbanización Torre De Portaceli, Ctra Bétera- Olocau, Km 7,5, 46118	Eliminar

At the bottom, there are navigation buttons: Anterior, 1, and Siguiente.

Figura 5-19. Torneos deportistas

En esta figura 5-20 podemos ver el torneo seleccionado anteriormente en el apartado 5.1.6. a la vista tenemos el detalle de cada torneo y nos da un botón como opción si deseamos inscribirnos y se nos abrirá un popup figura 5-21.

The screenshot shows the Frontenis-Free website interface. At the top, there is a navigation bar with links: Menú (highlighted in red), Perfil, Torneos, Canchas, and Salir. The main title is "Torneo Femenino 2022" with a female icon. Below the title, there are two small images: one of people at a podium and another of a frontenis court. To the right, under "Detalles del Torneo", it says "Fecha: 2022-07-23". There are two expandable sections: "Descripción" (Description) which contains text about the tournament, and "Provincia" (Province) which shows participant statistics. Below these sections, there is a "Ubicación" (Location) section with the address "Urbanización Torre De Portaceli, Ctra Bétera-Olocau, Km 7.5, 46118" and an "Inscribirse" (Register) button.

Figura 5-20. Ver torneo

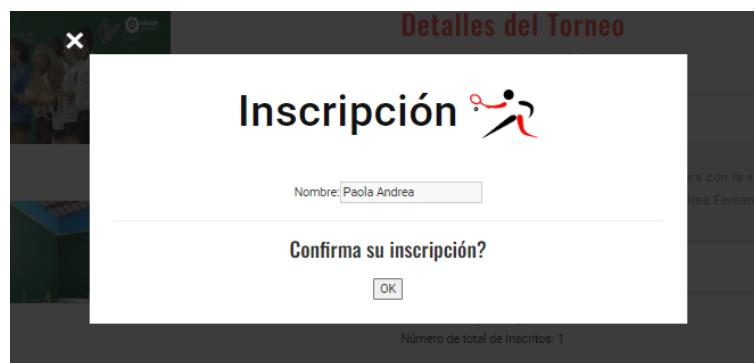


Figura 5-21. Inscripción torneo

El siguiente código podemos ver la inscripción al torneo el cual se encuentra en la carpeta model archivo add-desempeño.php

```
<?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("Error");
        window.location="../view/torneos.php";
    </script>';
    ;
}
else{
    echo '<script>
        alert("Duplicado");
        window.location="../view/torneos.php";
    </script>';
    ;
}
?>
```

Figura 5-22. Código - Inscripción torneo

En esta figura podemos el registro del torneo dentro de un popup lo podemos encontrar en view archivo ver-torneo.php

```
<div class="video-pop-wthree">
    <a style="color:blue; border:1px red" href="#small-dialog1" class="view play-icon popup-with-zoom-anim ">
        <i class="fa fa-pencil-square-o" aria-hidden="true"></i>Inscribirse</a>

        <!-- Popup Inscripción-->
        <div id="small-dialog1" class="mfp-hide w3ls_small_dialog wthree_pop">
            <h3 class="title">Inscripción
                </h3>
                <form action="../model/inscripcion.php" method="post" enctype="multipart/form-data">
                    <input type="hidden" name="id_torneo" value=<?php echo $id3;?>" >
                    <input type="hidden" name="id_usuario" value=<?php echo $usuario; ?>" >
                    Nombre:<input type="text" name="nombre" value=<?php echo $name; ?>" disabled>
                    <h3>Confirma su inscripción?</h3><br>
                    <input type="submit" value="OK">
                </form>
            </div>
        </div>
```

Figura 5-23. Popup – Inscripción

El agregar Torneos es una función que tiene el club en la figura 5-24 se puede observar con más detalle el formulario y una lista con los torneos que cada club administra, dispone de dos opciones editar o eliminar torneo.

Agregar Torneos





Datos

Formulario para agregar datos de un torneo:

Título:	<input type="text"/>
Fecha:	<input type="text"/> dd/mm/aaaa
País:	<input type="text"/>
Provincia:	<input type="text"/>
Ubicación:	<input type="text"/>
Descripción:	
Frontennis <input type="button" value="Estado:"/>	
<input type="button" value="Imagen Principal"/> Elegir archivo No se ha seleccionado ningún archivo <input type="button" value="Imagen 2"/> Elegir archivo No se ha seleccionado ningún archivo	
<input type="button" value="Publicar"/>	

Lista Torneos

Agrupar de

Resultados

Descripción	Ubicación	País	Provincia	Fecha	Acción
Primer Torneo femenino en Valencia, contara con la visita de grandes jugadores como, María Torres, Melisa Fernández	Urbanización Torre De Portaceli, Ctra Bétera-Olocau, Km 7,5, 46118	España	Valencia	2022-07-23	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
Segundo Torneo celebrado en Madrid, no te puedes perder esta gran fiesta del deporte, contara con máximos deportistas de élite.	Pº de las Yeserías, 3-5, 28005	España	Madrid	2022-07-30	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>

Buscar :

Anterior 1 Siguiente

Figura 5-24. Agregar Torneo club

```

<?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 5-25. Código - Agregar Torneo club

5.1.7 Canchas y reservas

El incorporar un buscador de canchas de frontón ayuda a los deportistas a disminuir el tiempo de búsqueda. En la figura 5-24. Podemos ver un buscador público para cualquier usuario sin necesidad de estar registrado.

En la parte superior tiene las opciones de registro dependiendo de si es un deportista o un club.

The screenshot shows a web application for searching tennis courts. At the top, there is a navigation bar with the logo 'Frontón-tenis' featuring a tennis racket icon. The menu items include 'Inicio' (Home), 'Torneos' (Tournaments), 'Canchas' (Courts), 'Deportista' (Athlete), and 'Club'. Below the menu, the title 'Buscador de Canchas' is displayed with a small tennis player icon. On the left, there is a search form with a dropdown for 'Agrupar de' set to '5' and a 'Resultados' button. A 'Buscar:' input field is present. The main content area contains a table listing five tennis clubs with their details:

Nombre ↑↓	Ubicación	Provincia	Ciudad	Opción
Club Frontón Orcasitas	C. del Camino Viejo de Villaverde, 28, 28041	Madrid	Madrid	Ver
Club Frontón Pirámides	P.º de las Yeserías, 3-5, 28005	Madrid	Madrid	Ver
Frontón Avda. Pablo Iglesias	Av. de Pablo Iglesias, 62, 28039 Madrid	Madrid	Madrid	Ver
Frontón de Ciudad Pegaso	Av. Séptima, 12, 28022	Madrid	Madrid	Ver
Frontón Municipal de Higueruelas	C. De Liria, 5, 46162 Higueruelas, Valencia	Valencia	Higueruelas	Ver

Below the table, there are navigation buttons for 'Anterior' (Previous), '1' (current page), '2', and 'Siguiente' (Next). At the bottom of the page is a large image of a tennis court with two players in action.

Figura 5-26. Buscador cancha – pública

En la figura 5-25 nos encontramos con un buscador, pero esta vez es interno del deportista registrado, este a su vez tiene un botón con un listado de sus reservas y el usuario pueda administrarlas.

Agrupar de
5 ▾

Resultados

Nombre	Ubicación	Provincia	Ciudad	Acción
Club Frontón Orcasitas	C. del Camino Viejo de Villaverde, 28, 28041	Madrid	Madrid	<button>Ver</button>
Club Frontón Pirámides	Pº de las Yeserías, 3-5, 28005	Madrid	Madrid	<button>Ver</button>
Frontón Avda. Pablo Iglesias	Av. de Pablo Iglesias, 62, 28039 Madrid	Madrid	Madrid	<button>Ver</button>
Frontón de Ciudad Pegaso	Av. Séptima, 12, 28022	Madrid	Madrid	<button>Ver</button>
Frontón Municipal de Higueruelas	C. De Liria, 5, 46162 Higueruelas, Valencia	Valencia	Higueruelas	<button>Ver</button>

Anterior **1** 2 Siguiente

Mis Reservas

Figura 5-27. Buscador cancha – Deportista

El listado de reservas nos sirve para ver la fecha y hora que tenemos reservado en cada cancha, contamos con una opción de eliminar.

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

Figura 5-28. Lista Reservas

Una vez hicimos una búsqueda como se vio en la figura 5-25 seleccionamos ver y se nos abrirá la siguiente figura 5-27 con el detalle de la cancha.

El detalle de esta cancha cuenta con dos fotos de la instalación deportiva, una ubicación en Google map, los horarios de entrada y salida de las canchas, y tenemos un botón Reserva el cual nos permite hacer una reserva de esta cancha.

Club Frontón Piramides





Detalles del Cancha

Horario: todos los días de 9:00am - 20:00pm

- Provincia

Madrid

- Ciudad

Madrid

[Reservar](#)

Ubicacion
Pº de las Yeserías, 3-5, 28005

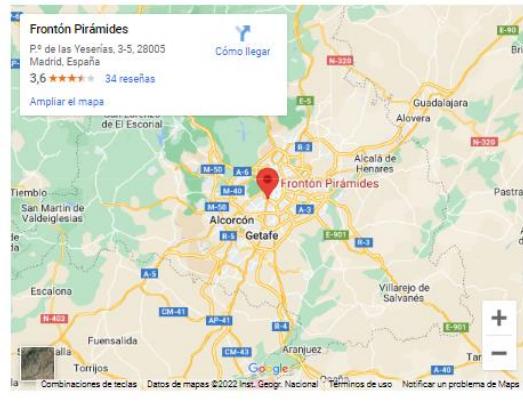


Figura 5-29. Ver Cancha

En la figura 5-28. Podemos ver el registro de nuestra reserva, en la parte superior de color verde nosotros seleccionamos la fecha y hora que queremos la reserva y guardamos con el botón Reserva, también podemos seleccionar el botón libre el cual está implementando para que se registre su reserva, cuenta con un buscador el cual filtra los datos por fecha.

The screenshot shows a reservation interface for a tennis court. At the top, there's a search bar with fields for 'Fecha' (dd/mm/aaaa) and 'Hora' (9:00 - 10:00), and a 'Reservar' button. Below the search bar is a table titled 'Resultados' with columns 'Fecha', 'Hora inicio', and 'Acción'. The table contains four rows for June 16, 2022:

Fecha	Hora inicio	Acción
2022-06-16	10:00-11:00	Libre
2022-06-16	11:00-12:00	Libre
2022-06-16	12:00-13:00	Reservado
2022-06-16	13:00-14:00	Reservado

At the bottom, there's a navigation bar with 'Anterior', a page number '1', and 'Siguiente'.

Figura 5-30. Reserva cancha

```
<?php
include '../controller/dbconnect.php';
include '../controller/datos.php';

$id4=$_GET['id_reserva_cancha'];
$estado=$_GET['estado'];
$querye="UPDATE `reserva_cancha` set estado = '$estado'
          WHERE id_reserva_cancha='$id4'";
mysqli_query($con,$querye);
header('location:canchas.php');
?>
```

Figura 5-31. Cambio estado reserva

La siguiente figura nos muestra la tabla de reserva la cual se encuentra en view archivo ver-cancha.php, también se puede apreciar el cambio de estado mencionado anteriormente y en la figura 5-30 muestra nuestra reserva implementada en model/reserva.php.

Fecha	Hora inicio	Acción
<?php \$query="SELECT * FROM reserva_cancha WHERE id_cancha='\$id7' "; \$resultado=mysqli_query(\$con,\$query); while (\$rows=mysqli_fetch_array(\$resultado)){ ?> <td><?php echo \$rows['fecha']?></td> <td><?php echo \$rows['hora_inicio']?></td> <td> <?php if(\$rows['estado']==0){ echo '<p>Libre </p>'; }else{ echo '<p>Reservado</p>'; } ?> </td> </tr> <?php } ?>		

Figura 5-32. Tabla Reserva

```
<?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 5-33. Reserva cancha

En esta vista el club puede agregar una cancha para informar a los deportistas las instalaciones. En el campo map se tiene que agregar un iframe de Google map(ver guía), los campos de logo, por si la cancha cuenta con ello y la segunda imagen se recomienda que coloquen fotos de la cancha para ver las condiciones de las canchas.

Agregar Cancha



Datos

Nombre Cancha:

Ciudad:

Provincia:

Ubicación:

Map:

Indicaciones

Logo: No se ha seleccionado ningún archivo.

Imagen: No se ha seleccionado ningún archivo

Agregar

Listar Canchas

Agrupar de

Resultados

Nombre	Ubicación	Provincia	Ciudad	Opciones
Club Frontón Orcasitas	C. del Camino Viejo de Villaverde, 28, 28041	Madrid	Madrid	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
Club Frontón Pirámides	Pº de las Yeserías, 3-5, 28005	Madrid	Madrid	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
Pirámides4	calle ayuso 23	Córdoba	MADRID	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>

Buscar :

[Anterior](#) **1** [Siguiente](#)

Figura 5-34. Agregar Cancha - Club

5.2 Chatbot

Para desarrollar el asistente virtual tipo Chatbot, requerimos de técnicas de procesamiento de lenguaje natural, para lo cual usaremos Dialogflow que es una herramienta de Google para crear chatbots, pensada para interacciones hombre computador. Esta API, basada en inteligencia artificial, permite a los usuarios interactuar con aplicaciones móviles/ web de forma natural mediante texto o voz. Para ello se tomó como arquitectura de modelo vista controlador como se observa en la siguiente figura 3-25, Se hace uso del lenguaje de programación JSON, HTML y CSS para la parte vista y modelo web. Para la parte web usaremos Node.js y creamos un servidor local para nuestro chatbot. La instalación de nuestro chatbot en nuestro servidor se verá en el apéndice B.

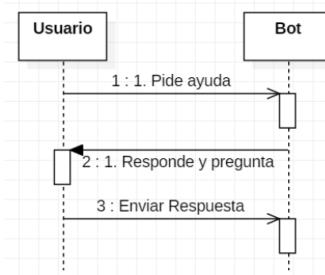


Figura 5-35. Diagrama de secuencia Primera conversación

5.2.1 Creación de agente

Necesitaremos estar registrados con una cuenta de google Dialogflow [19] para este proyecto se contó con una cuenta de estudiante gratis por un mes, los precios para proyectos futuros se encuentran está la siguiente referencia [20].

En la figura 5-36 podemos ingresar en la parte superior el nombre de nuestro agente “Asistente Sport”, luego podemos escoger el idioma y la zona esto es indiferente porque el agente puede aprender el idioma el cual nosotros lo entrenamos. Luego que se rellenen estos datos presionar el botón crear.

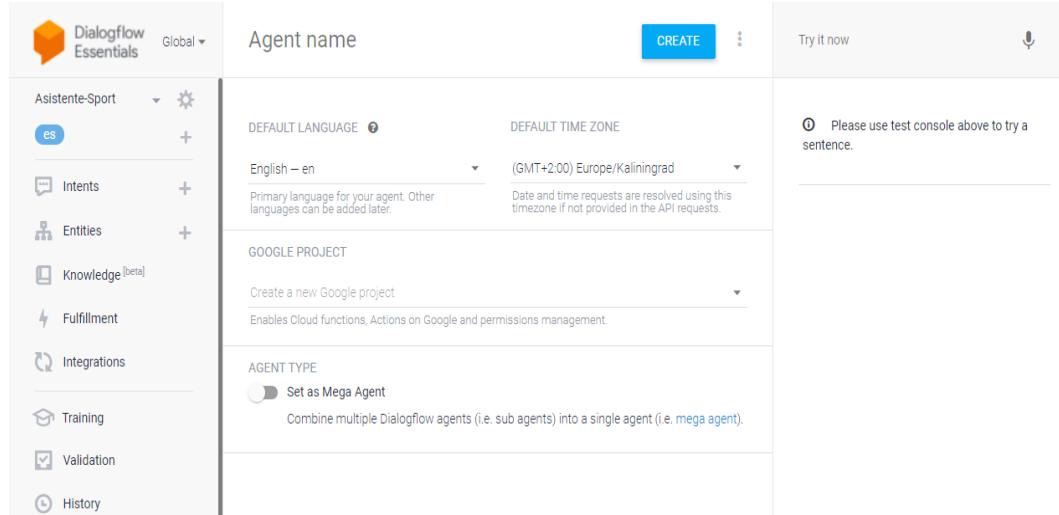


Figura 5-36. Creando agente

En la parte Izquierda tenemos el menú de paleta de conceptos que podemos configurar (Intents, Entities, Integración,Training).

Los Intents que vemos en la parte central son los comportamientos que se espera que se produzcan en la conversación con el usuario, aquí definimos frases, preguntas y clasificación de las noticias, canchas o torneos.

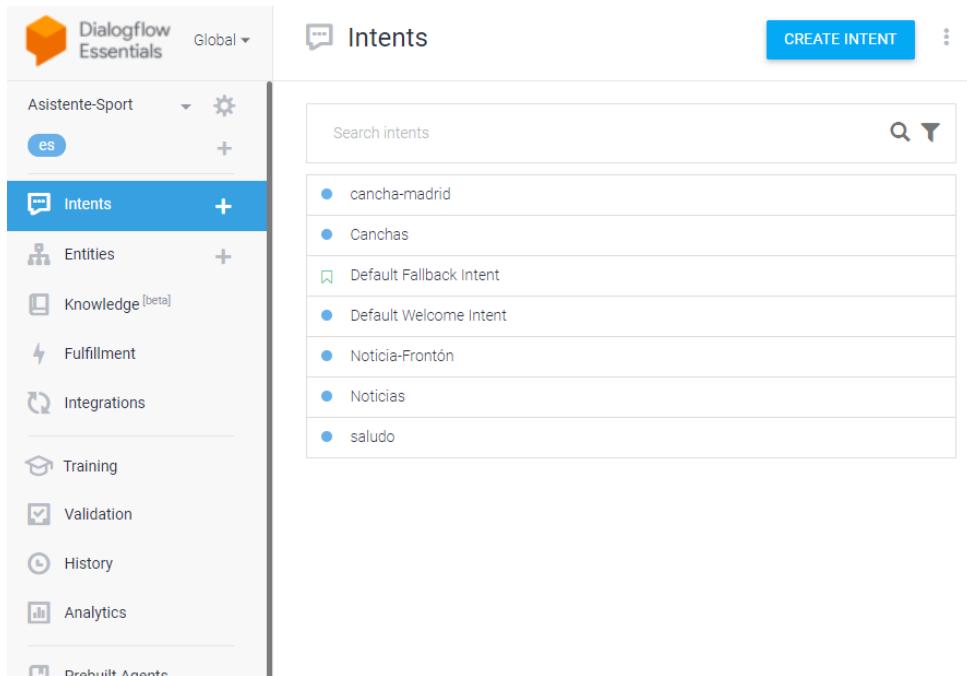


Figura 5-37. Lista de Intents.

5.2.2 Primera conversación

Esta primera intención es cuando el usuario comienza a preguntar o saludar a nuestro bot. Esta interacción se agrega a nuestro intents *Saludos* para que nuestro agente esté aprendiendo.

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

Tabla 5-1. Frases de entrenamiento

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

Tabla 5-2. Parámetro identificador

Intent Name	Text Response
Saludo	Bienvenido a tu asistente virtual!!, qué <u>estás</u> buscando noticias, torneos, canchas?

Figura 5-38. Primera conversación

5.2.3 Selección noticias

En la figura 5-39 podemos apreciar las siguientes frases de entrenamiento cuando el usuario indica que quiere una noticia, también en el apartado de respuestas agregamos una de tipo custom payload que significa su respuesta enriquecida esta mediante código.

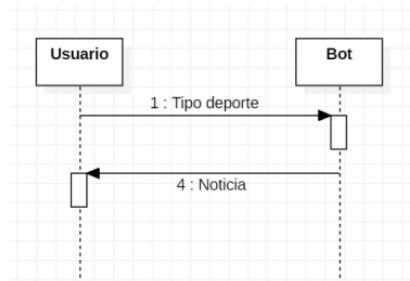


Figura 5-39. Diagrama de secuencia - Tipo deporte

The screenshot shows the Dialogflow Essentials interface for the "Noticias" intent. The left sidebar includes options like Asistente-Sport, Entities, Knowledge, Fulfillment, Integrations, Training, Validation, History, and Analytics. The main area shows the intent configuration:

- Intents**: A list of training phrases:
 - me puedes dar las ultimas noticias
 - news
 - Quiero ver noticias
 - Noticias
- Action and parameters**: A collapsed section.
- Responses**:
 - DEFAULT**:
 - Text Response**: Enter a text response
 - Custom Payload**:

```
1 {
2   "richContent": [
3     [
4       {
5         "type": "description",
6         "text": [
7           "Frontón",
8           "Frontenis",
9           "Pelota Vasca"
10        ],
11        "title": "Que deportes prefieres la noticia?"
12      }
13    ]
14 }
```

Figura 5-40. Selección de noticias.

5.2.4 Selección frontón

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

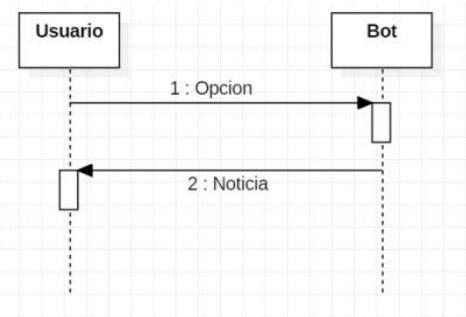


Figura 5-41. Diagrama de secuencia - Selección frontón

The screenshot shows the Dialogflow Essentials interface with the 'Noticia-Frontón' intent selected. The left sidebar includes options like Asistente-Sport, Entities, Knowledge, Fulfilment, Integrations, Training, Validation, History, Analytics, Prebuilt Agents, Docs, Trial Free, Upgrade, Support, Account, and Logout. The main panel displays the intent configuration with sections for Action and parameters, Responses, and a code editor for Custom Payload. The code editor contains the following JSON payload:

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

Figura 5-42. Selección frontón

5.2.5 Selección cancha

En las siguientes figuras podemos apreciar cuando el usuario selecciona que quiere ver una cancha de la provincia de Madrid, el agente clasifica y muestra el resultado filtrando por provincia, como se muestra en las figuras.

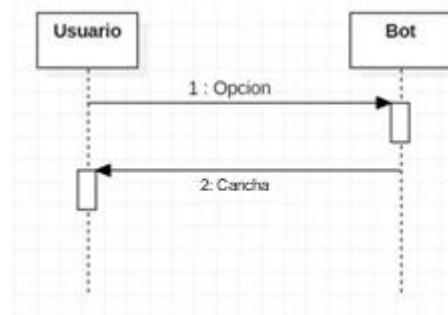


Figura 5-43. Diagrama de secuencia - Cancha

The screenshot shows the Dialogflow interface with two intent configurations side-by-side:

- Canchas Intent:**
 - Input Examples: "quiero ver canchas", "canchas".
 - Action and parameters: None.
 - Responses:
 - Text Response: "Enter a text response".
 - Custom Payload: A JSON snippet defining a rich card with a title, subtitle, and an image URL.
- cancha-madrid Intent:**
 - Input Examples: "quiero de madrid", "madrid", "a".
 - Action and parameters: None.
 - Responses:
 - Text Response: "Enter a text response".
 - Custom Payload: A JSON snippet defining a rich card with a title, subtitle, and an image URL.

Figura 5-44. Selección cancha

5.2.6 Resultados chatbot

En la parte inferior derecha nos encontramos con nuestro chatbot a la espera de que el usuario interactúe con él. En la figura 5-46 tenemos el código de integración dentro de nuestro proyecto, antes de insertarlo se instaló Node.js y Express los cuales en el apéndice B lo tendrás más explicado.



Figura 5-45. Asistente virtual - chartbot

```
<!-- 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>
  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 5-46. Chatbot

Las primeras conversaciones de nuestro chatbot son de bienvenida. El asistente pregunta al usuario que especifique su búsqueda, una vez el usuario le indica que quiere noticias el asistente le muestra un listado de deporte con referente a la noticia, seguidamente el usuario le informa que quiere una noticia de frontón a lo que el asistente envía la noticia que tiene actualmente.

Como se aprecia en la figura el asistente respondió con una imagen, título de la noticia y un link interno para dirigirlo a la noticia que se encontró.

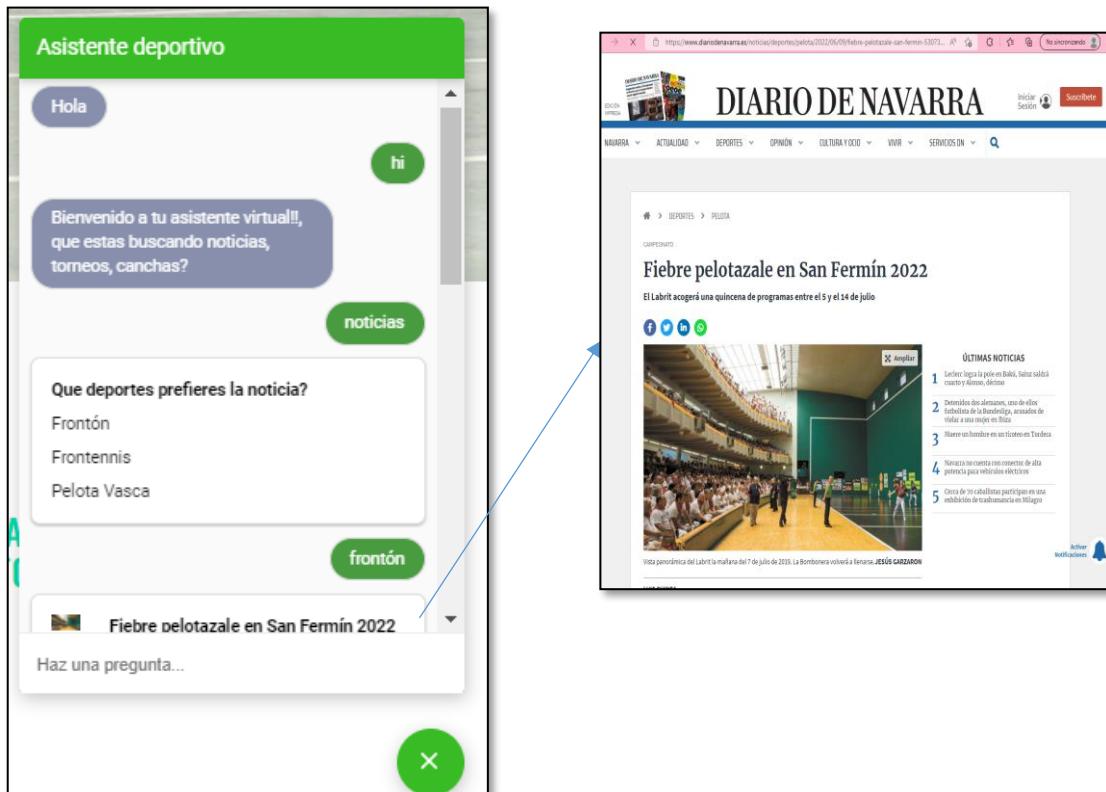


Figura 5-47. Chatbot – Noticias

En el apartado anterior se pudo apreciar el envío de noticias al usuario, pero a la vez el asistente le sigue preguntando si puede ayudarlo en otra búsqueda esta vez el usuario selección *cancha* (opción 3) la cual está integrada con una segunda intents, la provincia de la cancha, finalmente el asistente envía foto, título, dirección y un link interno para redirigirlo a la cancha encontrada en el sistema.

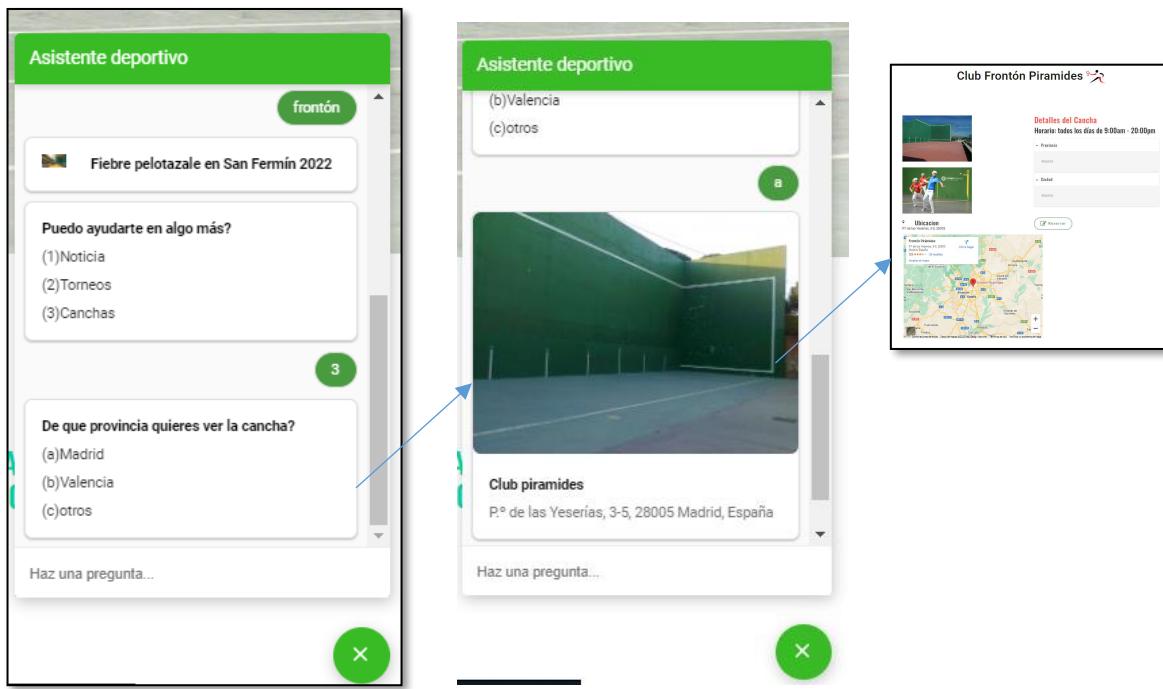


Figura 5-48. Chatbot - Canchas

Capítulo 6 - Conclusiones y trabajo futuro

En este capítulo se hablará sobre las conclusiones obtenidas al finalizar el proyecto, también se describirán algunas ideas para proyectos futuros y posibles implementaciones posteriormente.

6.1 Resultado frente a los objetivos

En este apartado pondremos frente a los resultados concluidos en base de los objetivos iniciales de este proyecto.

6.1.1 Realizar una aplicación web

Este objetivo se logró con éxito al tener implementada una aplicación web con sus respectivas funcionalidades que ayudan a mejorar la gestión y orden de las actividades en canchas de frontón.

6.1.2 Proporcionar un buscador de canchas

La idea principal era proporcionar un buscador para cualquier público el cual se logró satisfactoriamente como se muestra en el apartado 5.1.7. El buscador está implementado para público en general y a su vez en el buscador interno del deportista.

6.1.3 Registrarse en el sistema

El sistema permite al usuario poder registrarse correctamente, pero se le añadió una restricción de ingreso para que el usuario espere un máximo de 24 horas para que su cuenta esté activa.

6.1.4 Proporcionar listado de torneos

Este objetivo se concluyó en su totalidad, pero, además se integró una función para que el usuario pueda inscribirse en cada torneo, esto se almacena internamente para tener una mejor gestión de eventos.

6.1.5 Sistema de reserva

Este objetivo fue el más difícil de implementar ya que no podía cuadrar las horas, pero se logró con éxito y ahora cuenta con una doble funcionalidad interna en la reserva dando dos opciones al usuario a poder seleccionar su reserva mediante una tabla o hacerla directamente desde un formulario, esta función muestra al usuario las horas que están reservadas y libres.

6.1.6 Agregar desempeño

Se logró crear la funcionalidad donde el usuario puede agregar su desempeño de su juego, además cuenta con un listado histórico de sus juegos con un buscador.

6.1.7 Ver desempeño en gráfico de Google chart

Este objetivo innovador mediante Google chart se logró satisfactoriamente, el usuario puede visualizar sus puntuaciones mediante una gráfica, esto ayuda a visualizar más en porcentajes sus progresos. Aunque nos hubiera gustado agregarle más funcionalidades en trabajos futuros lo anexamos.

6.1.8 Disponer de un asistente virtual

Ya que la aplicación consistía en gestionar actividades deportivas, la idea de implementar un asistente virtual para que interactúe con el usuario y hacerlo más amigable nos motivó a integrarlo. La aplicación ahora cuenta con un chatbot que ayuda a mejorar las búsquedas de información al usuario este a su vez proporciona imágenes y datos de los que aprendió mediante el entrenamiento en Dialogflow.

6.2 Trabajo futuro

En este apartado describiremos las integraciones que me hubiera gustado implementar, pero por motivos de tiempo no se pudo concluir. Espero que en trabajos futuros se logre mejorar.

6.2.1 Añadir un filtrado en el gráfico desempeño

En desempeño se logró crear una gráfica mediante Google chart, pero se podría hacer un filtrado para que muestre solo un rango de fechas o por mes para que pueda ser más específico su resultado.

6.2.2 Insertar ubicación geográfica

El administrador al momento de agregar una cancha tiene que ingresar ciertos datos siendo uno de ellos el map el cual es ingresado manualmente, se podría implementar con la Api de Google map una inserción directa sin salir de la aplicación.

6.2.3 Entrenar con las funciones a nuestro asistente

En este proyecto se incluyó la integración de nuestro asistente, pero por motivos de tiempo solo está entrenado en algunas funcionalidades, en el futuro se podría entrenar mejor si se le agrega más funcionalidades ejemplo:

- Buscar noticias de una fecha específica.
- Buscar canchas que esté libres en una hora y fecha.
- Búsqueda de canchas más cerca a su casa.

6.2.4 Asistente virtual de voz

Con el objetivo de llegar a todo público, se podría agregar un asistente mediante voz para las personas ciegas o con alguna discapacidad. El asistente de voz ayudaría a las personas con alguna discapacidad para escribir o ver la aplicación directamente.

Dialogflow tiene integrado el asistente de voz, sólo quería añadirla como un nuevo intents e integrarlo.

Chapter - 7. Introduction

In recent years we have seen the great need for people to disconnect from work, to achieve physical and mental stability through sport by going to recreational spaces, sports facilities and due to this growth we find ourselves under a large mass of people who come to these facilities without any control and unaware of activities such as tournaments or others.

In turn, this growth brings dissatisfaction to small and large athletes when they go to the few public facilities that we find in our community and when they go they are full, having to wait up to 3 hours to use the courts, and not seeing good administrative management we see ourselves before people who take over the courts preventing other athletes from playing freely.

Sport does not have to be an obstacle for anyone, on the contrary, the athlete has to feel free to do their activities, for this reason this project was carried out, which has better management, providing clubs and athletes with being safe and free to play.

Frontón-Tennis is an application designed for athletes and to improve their quality of service. This application allows you to be aware of the availability of sports courts, make your reservations, see the latest tournaments of the month and has a section to place your performance, at your Sometimes it has a virtual assistant for the general public who will advise you on your visit and search for news, tournaments, courts to improve the user experience.

7.1. Motivation

This project was developed from experiences lived in the first person, being a reference for many athletes who are discriminated against when going to play on public courts because they are women, it is considered that this sport is more for the male gender, being a delay in time. in which we live, also because there are few courts to practice racquetball, it is usually very full, spending a waiting time of between 2 or 3 hours for the previous players to finish.

With this project I want to help many athletes to have a search tool at hand and put them in contact with clubs, tournaments, athletes and at the same time have updated information for sports lovers.

Depending on why a web application was made, it is because it is aimed at all users who can use the Internet, since this system adapts to any web or mobile device. This application has a public part that you do not necessarily have to be registered to see, this makes it easier for any user to have access to a safe and quality sport.

7.2. Objectives

The main objective is to design a web application to manage activities in fronton courts, mainly in public facilities in the community of Madrid with expansion to different provinces.

- Create a web application to help manage activities on fronton courts.
- Provide a search engine for the general public and registered users, to find a court by city or province.
- Provide the user with the option of being able to register to use the internal application.
- Provide the user with a list of tournaments that are being promoted to participate.
- Offer the user a reservation system to select a date and time to be able to use the sports facilities.

- The user can add their performance to have a record of their points and games won.
- See performance through a graph in Google chart.
- Have a virtual assistant to help you in your search.

Chapter - 8 Conclusions and future work

This chapter will talk about the conclusions obtained at the end of the project, it also describes some ideas for future projects and possible implementations later.

8.1 Result against the objectives

In this section we will confront the results concluded based on the initial objectives of this project.

8.1.1 Make a web application

This objective was successfully achieved by having implemented a web application with its respective functionalities that help improve the management and order of activities in fronton courts.

8.1.2 Provide a court finder

The main idea was to provide a search engine for any audience, which was successfully achieved as shown in section 5.1.7. The search engine is implemented for the general public and in turn in the internal search engine of the athlete.

8.1.3 Register in the system

The system allows the user to register correctly, but an entry restriction was added so that the user waits a maximum of 24 hours for his account to be active.

8.1.4 Provide list of tournaments

This objective was completed in its entirety, but, in addition, a function was integrated so that the user can register for each tournament, this is stored internally for better event management.

8.1.5 Reservation system

This objective was the most difficult to implement since it could not balance the hours, but it was successfully achieved and now has a double internal functionality in the

reservation giving two options to the user to be able to select their reservation through a table or make it directly from a form, this function shows the user the hours that are reserved and free.

8.1.6 Add Performance

It was possible to create the functionality where the user can add their performance of their game, it also has a historical list of their games with a search engine.

8.1.7 See performance in Google chart graph

This innovative objective using Google chart was successfully achieved, the user can visualize their scores through a graph, this helps to visualize their progress more in percentages. Although we would have liked to add more functionalities in future works, we will add it.

8.1.8 Have a virtual assistant

Since the application consisted of managing sports activities, the idea of implementing a virtual assistant to interact with the user and make it more user-friendly motivated us to integrate it. The application now has a chatbot that helps the user improve information searches, which in turn provides images and data from what it learned through training in Dialogflow.

8.2 Future work

In this section we will describe the integrations that I would have liked to implement, but for reasons of time it could not be concluded. I hope that future work will improve.

8.2.1 Add a filter in the performance graph

In performance, it was possible to create a graph using Google chart, but a filter could be made to show only a range of dates or by month so that its result can be more specific.

8.2.2 Insert geographic location

When adding a field, the administrator has to enter certain data, one of them being the map, which is entered manually. A direct insertion could be implemented with the Google map Api without leaving the application.

8.2.3 Train our assistant with the functions

In this project the integration of our assistant was included, but for reasons of time it is only trained in some functionalities, in the future it could be trained better if more functionalities are added, for example:

- Search for news from a specific date.
- Find courts that are free at a time and date.
- Search for courts closer to your home.

8.2.4 Virtual assistant of you

In order to reach the general public, a voice assistant could be added for people who are blind or have a disability. The vos assistant would help people with some disability to write or view the application directly.

Dialogflow has the voice assistant integrated, I just wanted to add it as a new intents and integrate it.

BIBLIOGRAFÍA

- [1] 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].
- [2] J. Martinez, «University of Houston,» Febrero 2018. [En línea]. Available: https://uh.edu/ir/reports/special-reports/conference-presentations/tair-2018_googlecharts_martinez.pdf.
- [3] 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.
- [4] ORACLE, «ORACLE,» 2022. [En línea]. Available: <https://www.oracle.com/chatbots/what-is-a-chatbot/>.
- [5] Runkeeper, «Runkeeper,» [En línea]. Available: <https://play.google.com/store/apps/details?id=com.fitnesskeeper.runkeeper.pro&hl=es&gl=US>.
- [6] T. S. Trancker. [En línea]. Available: <http://tennisscoretracker.com/>.
- [7] Clupik. [En línea]. Available: <https://clupik.com/>.
- [8] 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>.
- [9] A. Schiaffarino, «Infranetworking,» 2019. [En línea]. Available: <https://blog.infranetworking.com/modelo-cliente-servidor/>.
- [10] PHP.net, «PHP.net,» [En línea]. Available: <https://www.php.net/manual/es/intro-whatis.php>.
- [11] MySQL, «MySQL,» [En línea]. Available: <https://www.mysql.com/about/>.

- [12] ApacheFriends, «ApacheFriends,» [En línea]. Available: <https://www.apachefriends.org/about.html>.
- [13] GoogleCharts, «GoogleCharts,» [En línea]. Available: <https://developers.google.com/chart/interactive/docs>.
- [14] NodeJS, «NodeJS,» [En línea]. Available: <https://nodejs.org/en/about/>.
- [15] C. Academy, «Code Academy,» [En línea]. Available: <https://www.codecademy.com/article/what-is-express-js>.
- [16] GoogleCloud, «GoogleCloud,» [En línea]. Available: <https://cloud.google.com/dialogflow/docs>.
- [17] W3layouts, «W3layouts,» 2020. [En línea]. Available: <https://w3layouts.com/template/tennis-hub-sports-category-flat-bootstrap-responsive-web-template/>.
- [18] G. Chart, «Google,» [En línea]. Available: https://developers.google.com/chart/interactive/docs/quick_start.
- [19] Google, «Dialogflow,» [En línea]. Available: <https://dialogflow.cloud.google.com/>.
- [20] Google, «GoogleCloud,» [En línea]. Available: <https://cloud.google.com/dialogflow/pricing>.
- [21] Ionos, «Ionos,» 2019. [En línea]. Available: <https://www.ionos.es/digitalguide/servidores/herramientas/instala-tu-servidor-local-xampp-en-unos-pocos-pasos/>.
- [22] V. code. [En línea]. Available: <https://code.visualstudio.com/download>.
- [23] L. A. Bucki, Word 2013 Bible, John Wiley & Sons, 2013.
- [24] CFI, «Cursos de Formación en Informática,» [En línea]. Available: <http://cursosinformatica.ucm.es>. [Último acceso: 01 06 2019].

- [25] 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>.
- [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/>.

APÉNDICES

Apéndice A - Manual de usuario

En este apartado se explicará los pasos previos para que el usuario pueda ir navegando en la aplicación, el usuario puede hacer uso de cualquier navegador que tenga instalado en su ordenador, móvil, tablet, requerimientos mínimos.

A.1. Inicio de sesión y Registro

Una vez nos encontramos en la página de inicio nos dirigimos a la parte superior sobre el tipo de usuario, el botón verde si es que somos deportistas y azul si somos un club.



Figura A. 1. Seleccionar tipo usuario

Para hacer uso del sistema necesitamos registrarnos, ingresamos todos los campos y nos saldrá un mensaje indicándonos que nuestra cuenta estará activa dentro de 24 horas, si hubiera error en los datos se nos indicará en cada campo.



Figura A. 2. Registro usuario

Una vez pase las 24 horas ingresamos con las credenciales que ingresamos en el registro.



Figura A. 3. Inicio sesión

A.2. Menú principal

En esta vista podemos ver las opciones que tenemos para hacer uso, nuestro perfil, torneos, desempeño, canchas, si seleccionamos cualquiera de ellos nos dirige otra vista explicada en los siguientes apartados.



Figura A. 4. Menú principal

A.3. Perfil

Si nos encontramos en el perfil de usuario podemos agregar o editar nuestros datos personales, la foto de perfil se cambia independiente de los datos.

La imagen muestra la vista del perfil de Paola Andrea. En la parte superior, se ve el logo "Frontón-tenis" y los enlaces "Menú", "Perfil", "Desempeño", "Torneos", "Canchas" y "Salir". La sección central es "Perfil" con un icono de tenis. A la derecha de la sección de perfil hay una foto de Paola Andrea sosteniendo una raqueta. Debajo de la foto, hay un cuadro que dice "Ingrésa tu foto" y "Elegir archivo | No se ha seleccionado ningún archivo | Actualizar". A la izquierda, hay una lista de datos personales con campos editables: Nombre (Paola Andrea), Apellido (Mendoza), Email (paola@ucm.es), Dirección (Calle Ayuso), Edad (31), Peso (45) y Contraseña (campo oculto). Hay un botón "Guardar" y un botón "Dar de baja".

- 1. Cerrar sesión:** salir de la aplicación.
- 2. Dar de baja:** Eliminar nuestra cuenta.

Figura A. 5. Perfil de usuario

A.4. Desempeño

Esta parte nos permite ingresar nuestro desempeño de nuestros juegos para tener un registro de los puntos.

1. Formulario de desempeño, agregamos el tipo de deporte (Frontón, Frontenis, otros), Tipo de juego si es en pareja o individual, Score son los puntos.
2. Gráfico con nuestros resultados visto en porcentajes.
3. Buscador de nuestro desempeño, agregamos por fecha o filtramos por deporte.
4. Podemos eliminar nuestros puntos.



Figura A. 6. Desempeño deportista

A.5. Torneos

Cuando nos encontramos en torneos veremos los torneos que están activos, si seleccionamos alguno de ellos nos dirige al detalle del torneo como indica en el paso siguientes:

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

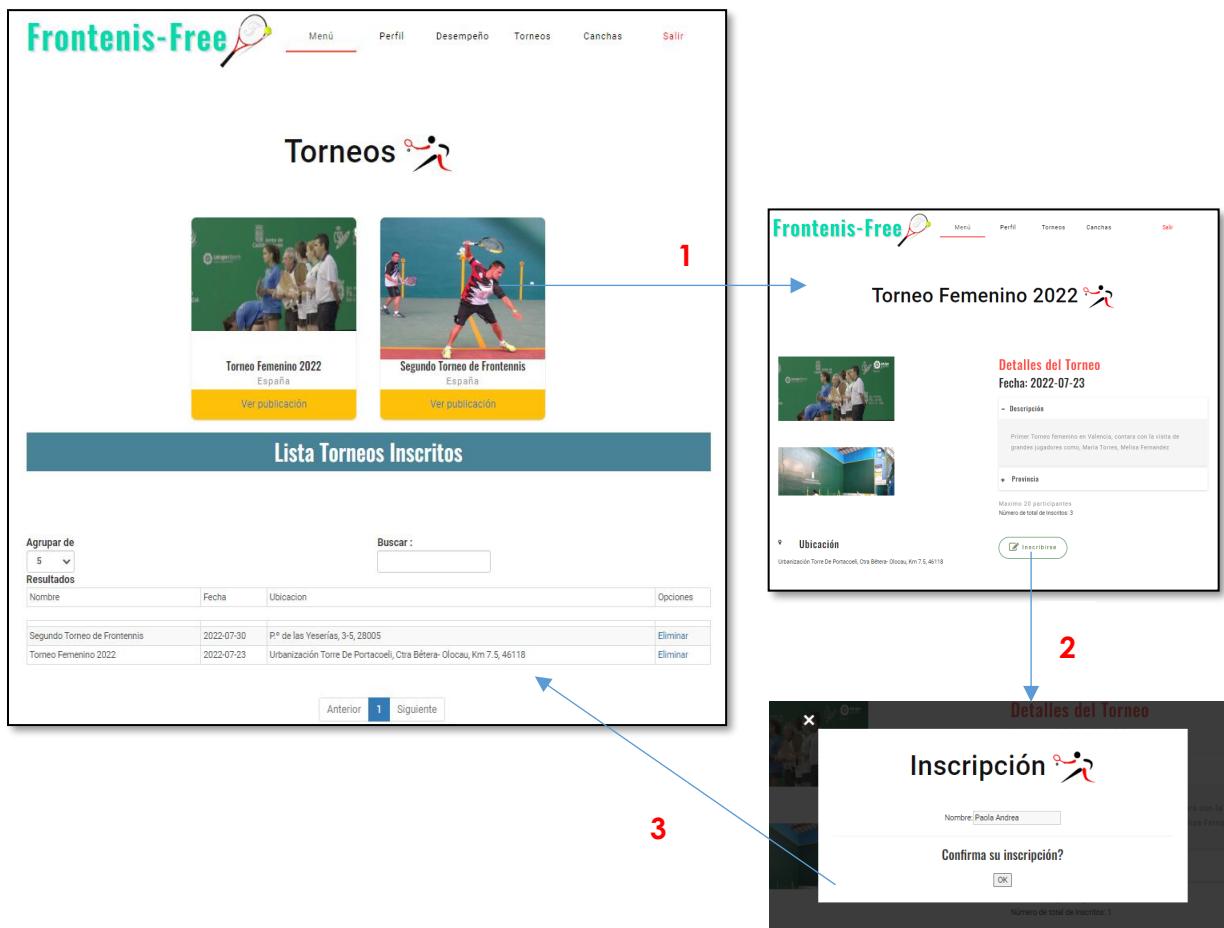


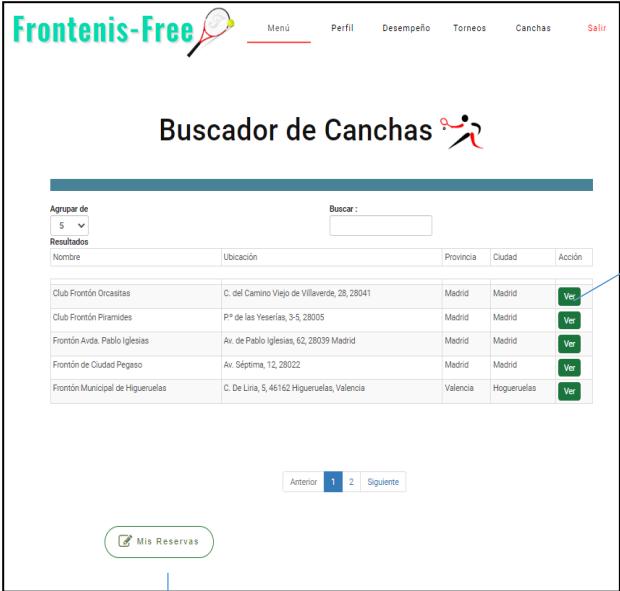
Figura A. 7. Torneos

A.6. Canchas y reserva

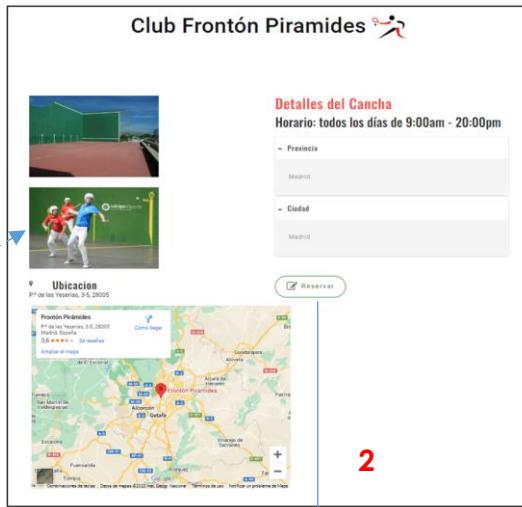
El buscador de canchas nos sirve para encontrar una cancha por provincia. A continuación, se describen los pasos.

1. Ver detalle de la cancha
2. Hacer reserva de la cancha por fecha y hora, podemos seleccionar los botones libres o rellenar el registro color verde.
3. Lista de reservas, nos da opción de eliminar reserva.

1



2



3



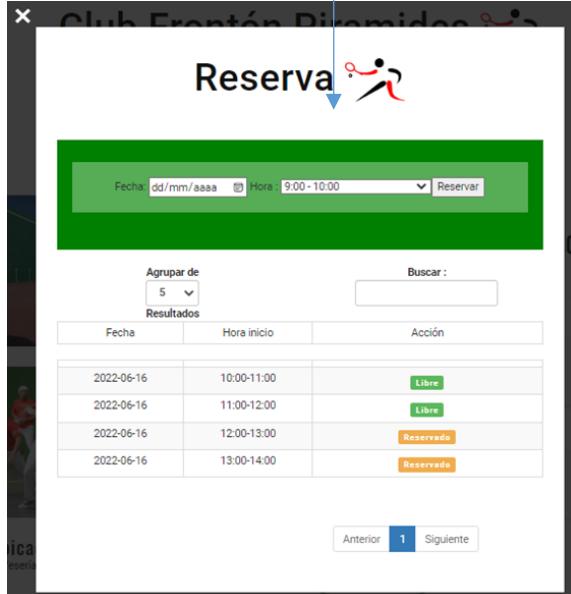


Figura A. 8. Ver cancha y reserva

Apéndice B - Manual de instalación

Nuestra aplicación esté alojado de manera local, para ellos se utilizó la plataforma Xampp como servidor y Visual code para la implementación y desarrollo del código.

B.1. Aplicación web

Puedes descargar la aplicación en nuestro repositorio ([Enlace](#)) en el encontrarás la carpeta raíz y la bd.

B.1.1. Servidor

1. Necesitaremos instalar Xampp desde la página oficial [12] puedes seguir estos pasos para instalarlo [21]

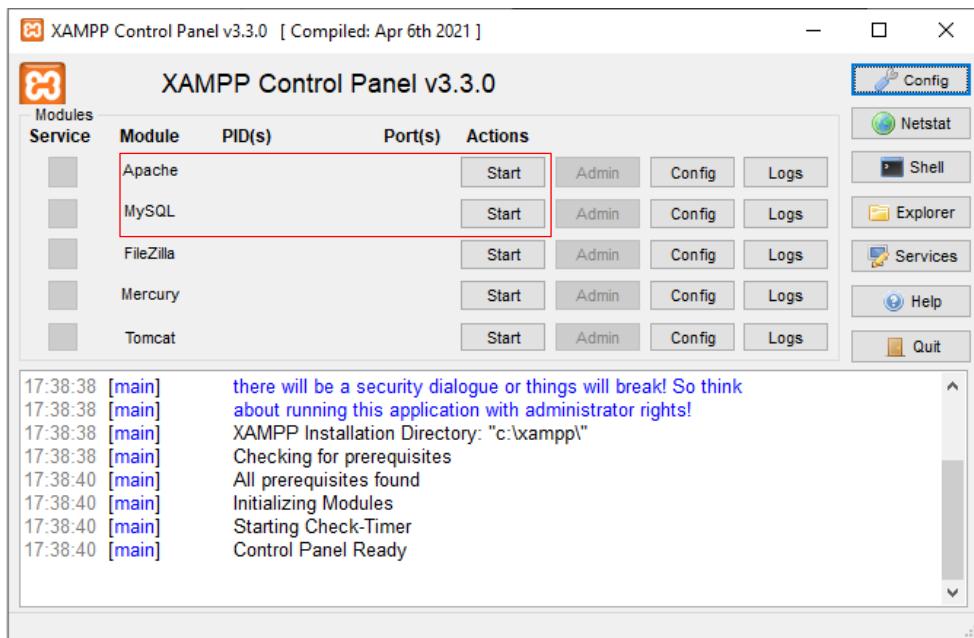


Figura B. 1. Xampp

2. La carpeta descargada tienes que agregarlo en C:\xampp\htdocs luego se inicia Apache y MySQL en nuestro Xampp.

3. Ahora abrimos en el navegador web <http://localhost/phpmyadmin> e importamos nuestra base de datos *bd_sport*.

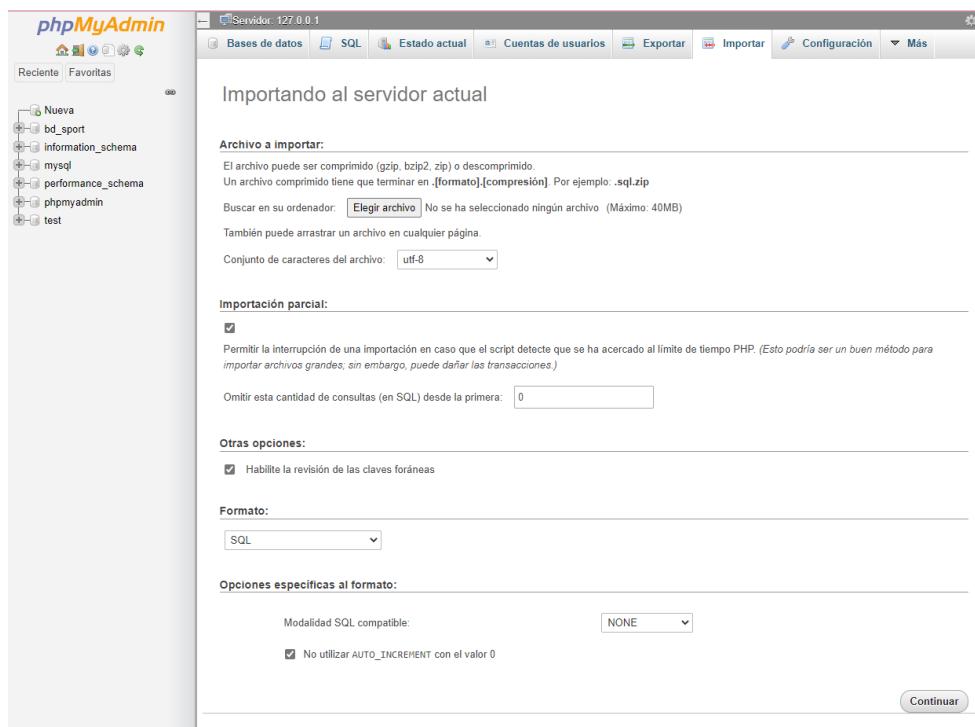


Figura B. 2. PhpMyAdmin

4. Luego abrimos en el navegador <http://localhost/fronton-tenis> Si todo va bien ya podemos ver nuestra aplicación.
5. Para editar el código puedes utilizar cualquier editor de código para este proyecto utilice Visual code [22]

B.2. Chatbot

Necesitamos instalar Node.js y express [23] a continuación se describen los pasos para instalar nuestros chatbot.

1. Abrimos un terminal en visual code y agregamos las siguientes instrucciones
 - npm init (Le damos enter a todo)
 - npm install express

Se nos instalará dentro de nuestro proyecto los módulos de Node y package.json.

2. Necesitaremos crear un API en Google Cloud, primero creamos un proyecto como se muestra en la siguiente figura. [24]

Google Cloud Platform Buscar Productos, recursos, documentos (/)

Proyecto nuevo

Tienes 9 projects restantes en tu cuota. Solicita un incremento o borra algunos proyectos. [Más información](#)

[MANAGE QUOTAS](#)

Nombre del proyecto * ?

ID de proyecto: asistente-virtual-353416. No se podrá cambiar más tarde. [EDITAR](#)

Organización * ▾

Selecciona una organización para vincularla a un proyecto. No podrás cambiar esta selección más adelante.

Ubicación * EXPLORAR

Organización o carpeta superior

[CREAR](#) [CANCELAR](#)

Figura B. 3. Crear proyecto en Google

3. Creamos una credencial como se muestra en la figura B.4

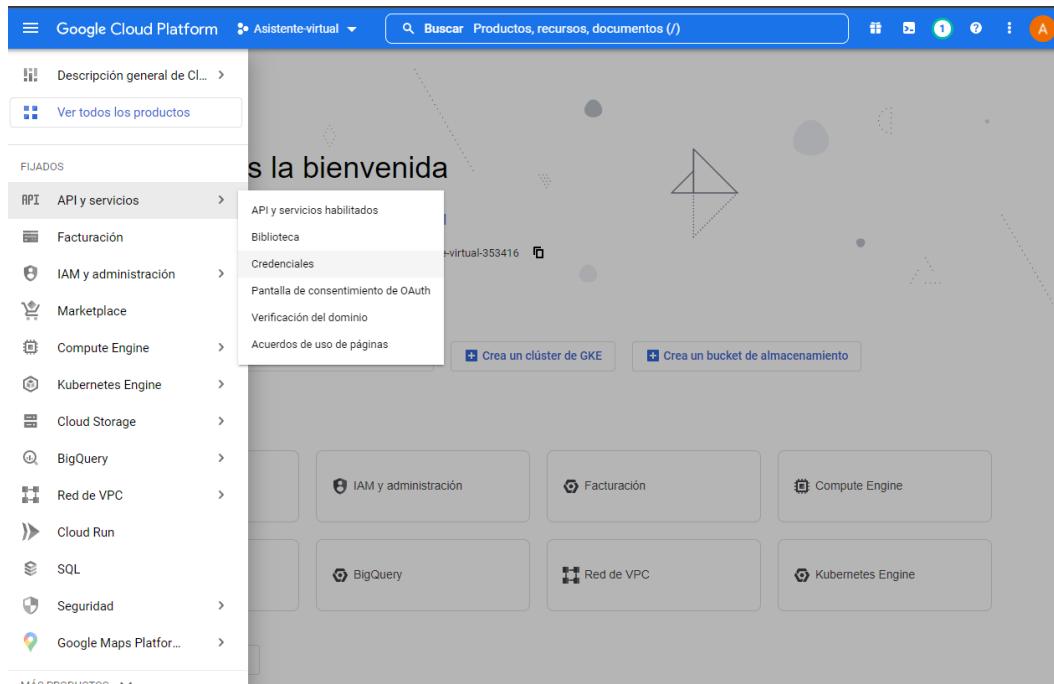


Figura B. 4. Credenciales

The screenshot shows the 'API y servicios' section of the Google Cloud Platform console. On the left, a sidebar lists various options: 'API y servicios habilitados', 'Biblioteca', 'Credenciales' (which is selected), 'Pantalla de consentimiento ...', 'Verificación del dominio', and 'Acuerdos de uso de páginas'. The main area is titled 'Credenciales' with buttons for '+ CREAR CREDENCIALES' and 'BORRAR'. A modal window is open, titled 'Clave de API', with the sub-section 'Claves de API'. It contains a warning 'Recuerda configurar la pantalla de consentimiento de OAuth con información sobre tu app.' Below this are sections for 'ID de cliente de OAuth 2.0', 'Cuentas de servicio', and 'Ayúdame a elegir' (with the note 'Responde algunas preguntas para decidir qué tipo de credencial usar'). At the bottom of the modal is a button 'CONFIGURAR PANTALLA DE CONSE'.

Figura B. 5. Clave API

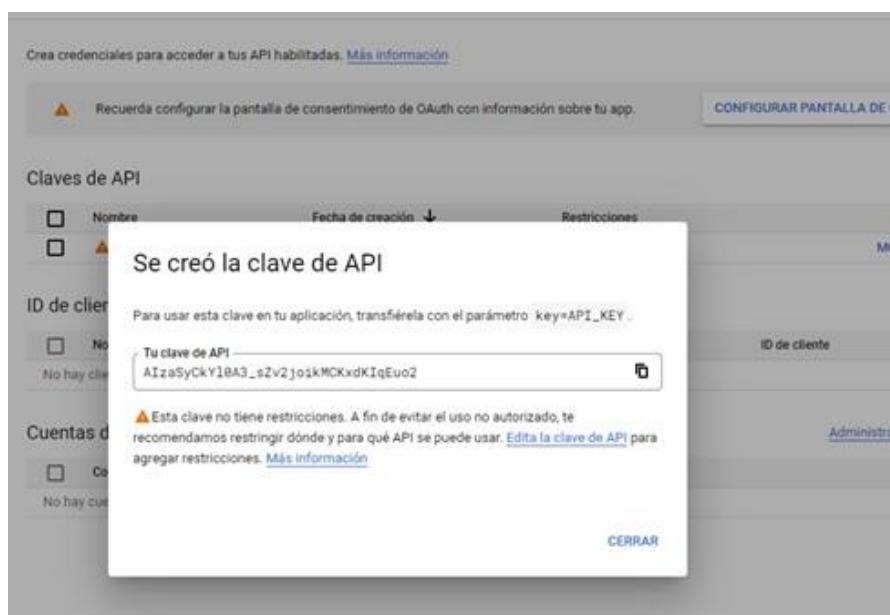


Figura B. 6. Clave Google

Con este código ya podemos enlazarlo en Dialogflow para conectarlo a nuestro chatbot.

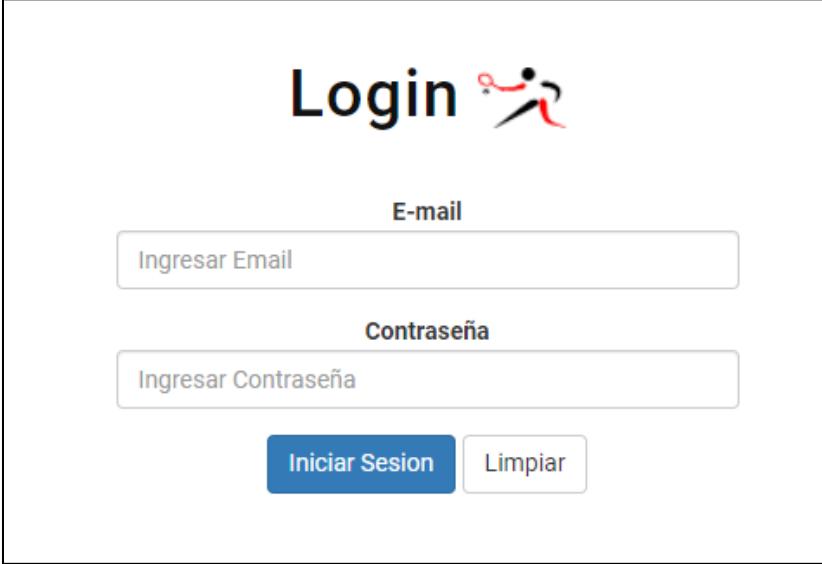
Apéndice C - Gestión administrador

A continuación, se mostrará la implementación que se tiene del administrador del sistema, los códigos de los siguientes apartados se encuentran dentro del proyecto en git.

C.1. Inicio sesión

El administrador ingresa mediante <http://localhost/Fronton-tenis/admin.php>

Sus credenciales son: pamela@gmail.com y su contraseña: 123456



The image shows a login form titled "Login" with a red tennis player icon. It has two input fields: "E-mail" and "Contraseña", both labeled "Ingresar Email" and "Ingresar Contraseña" respectively. Below the inputs are two buttons: a blue "Iniciar Sesión" button and a white "Limpiar" button.

Figura C. 1. Iniciar sesión – Administrador

En la figura C.2. podemos ver el menú principal que tiene el administrador y en la figura C.3. Nos encontramos con la gestión de usuarios en la cual el administrador puede dar de alta al usuario editar y eliminar.

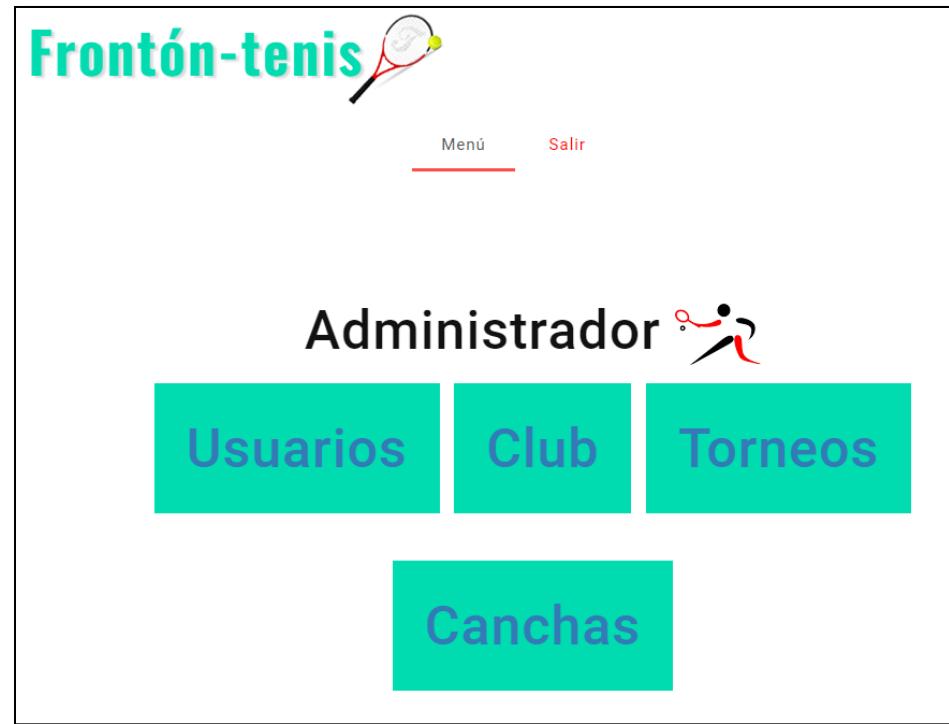


Figura C. 2. Menú - Administrador

Usuarios						
Agrupar de		Buscar :				
5						
Resultados						
ID ↑↓	Nombre	Apellidos	Email	Estado ↑↓	Opciones	
1	Alice	Ramirez	dc@gmail.com	Alta	editar	Eliminar
2	Alicia	Joy	alicejoyramirez@gmail.com	Activo	editar	Eliminar
3	Maria	Joy	ajoyer@ucm.es	Activo	editar	Eliminar
4	Pamela	Reyes	pamela@gmail.com	Activo	editar	Eliminar
5	Paola Andrea	Mendoza	paola@ucm.es	Activo	editar	Eliminar

Anterior 1 Siguiente

Figura C. 3. Gestión usuarios

Club 

Agrupar de Buscar :

Resultados

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

Anterior **1** Siguiente

Figura C. 4. Gestión club

Torneos 

Agrupar de Buscar :

Resultados

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

Anterior **1** Siguiente

Figura C. 5. Gestión Torneos