

**U**niversidade

de **V**igo

**E**scuela **S**uperior de **I**ngeniería **I**nformática

|  |
| --- |
| Memoria del Trabajo de Fin de Grado que presenta  **D. Brais Domínguez Álvarez**  para la obtención del Título de Graduado en Ingeniería Informática  **Gestión Integral de una Academia de Tenis** |

|  |  |
| --- | --- |
|  | , 2018  **Trabajo de Fin de Grao Nº**:  **Tutor/a:**  **Área de conocimiento:**  **Departamento:** |

Contenido

[Introducción 3](#_Toc513389347)

[Objetivos 3](#_Toc513389348)

[Solución Propuesta 4](#_Toc513389349)

[Planificación y Seguimiento 6](#_Toc513389350)

[Arquitectura 7](#_Toc513389351)

[Tecnologías e Integración de Productos de Terceros 7](#_Toc513389352)

[Especificación y Análisis de Requisitos 7](#_Toc513389353)

[Modelo de Dominio 7](#_Toc513389354)

[Análisis de Requisitos 8](#_Toc513389355)

[Requisitos no Funcionales 8](#_Toc513389356)

[Requisitos Funcionales 8](#_Toc513389357)

[Casos de Uso 12](#_Toc513389358)

[Diseño del Software 17](#_Toc513389359)

[Diseño Dinámico 17](#_Toc513389360)

[Diseño Estático 20](#_Toc513389361)

[Gestión de Datos e Información 21](#_Toc513389362)

[Diagrama Entidad-Relación 21](#_Toc513389363)

[Estructura Física de las Tablas 22](#_Toc513389364)

[Pruebas Llevadas a Cabo 25](#_Toc513389365)

[Gestionar Usuario 25](#_Toc513389366)

[Manual de Usuario 31](#_Toc513389367)

[Principales Aportaciones 31](#_Toc513389368)

[Conclusiones 31](#_Toc513389369)

[Vías de Trabajo Futuro 31](#_Toc513389370)

[Referencias 31](#_Toc513389371)

[Anexos 31](#_Toc513389372)

# Introducción

La actividad física y la práctica del deporte, se encuentran en constante desarrollo y crecimiento desde el siglo XX, generando una importante oferta y demanda de productos y servicios deportivos, hasta el punto de producir un fuerte impacto en nuestra sociedad, llegando a considerarse el deporte a día de hoy, como un espejo o modelo de la sociedad.

El deporte, escolar y federado principalmente, ha sido fomentado y llevado a cabo en su mayoría, por voluntarios, cuya labor ha sido y seguirá siendo, fundamental para el desarrollo del deporte como lo entendemos actualmente.

No obstante, existe una tendencia, por parte de las instituciones y entidades, a ofertar un deporte de mayor calidad y a contar con profesionales con una formación adecuada, que permita satisfacer las exigencias de unos usuarios, que requieren de servicios y actividades más satisfactorias, ya que no sólo buscan el ocio en el deporte, si no que demandan una instrucción que les permita alcanzar la excelencia en su nivel y el éxito deportivo, independientemente de que se dediquen al ejercicio de alta competición o aficionado.

Para mejorar la profesionalización de este servicio, se han establecido estudios y formaciones en el campo del deporte, abarcando desde la Formación Profesional hasta el Grado Universitario, dando lugar a entrenadores y monitores altamente competentes.

Finalmente, para que estos profesionales puedan llevar a cabo su labor con mayor eficacia, necesitarán contar con el material y las herramientas adecuadas, no sólo deportivas, si no de gestión, para complementar a las anteriores y garantizar una atención de calidad y personalizada a cada usuario, obteniendo beneficio de este modo, tanto deportistas como entrenadores/monitores.

# Objetivos

Este proyecto pretende servir de ayuda a la profesionalización del deporte, en particular, del tenis, uno de los más populares y en auge actualmente.

Consistirá en una herramienta que facilite a una academia de tenis:

1. Gestión de usuarios.
2. Gestión de espacios
3. Gestión de cursos.
4. Gestión de torneos, con sus respectivos cuadros y partidos.
5. Gestión de eventos.
6. Gestión de reservas de cursos de los clientes.
7. Gestión de reservas de eventos de los clientes.
8. Gestión de reservas de torneos de los clientes.
9. Notificación de las diferentes incidencias que puedan surgir a cualquier miembro que forme parte de la academia.

Al mismo tiempo, simplificará a los deportistas:

1. Obtener la información que precisen sobre los servicios que demanden de la academia y sus características (horarios, lugares, costes).
2. Gestión de sus reservas a cursos.
3. Gestión de sus reservas a eventos.
4. Gestión de sus reservas a torneos.
5. Comunicación con sus compañeros, monitores o responsables de la academia para la resolución de cualquier conflicto.

Garantizará:

1. Privacidad de sus usuarios y gestión de permisos.
2. Control de errores que se pudieran producir al introducir los datos en la aplicación.

El objetivo final, será proporcionar una herramienta de calidad, con una baja curva de aprendizaje, que facilite su uso a cualquier usuario de la aplicación (responsables de la academia, entrenadores y deportistas) y favorezca a la profesionalización de la práctica y docencia del tenis.

# Solución Propuesta

Para cumplir con los objetivos detallados anteriormente, se propone la creación de una aplicación web desarrollada en PHP, con MySQL como sistema gestor de base de datos, que pueda ser accedida desde internet mediante un navegador.

Se ha optado por una aplicación web, para facilitar el trabajo a los usuarios, ya que, de esta manera, se evitará la instalación de software en sus equipos, así como problemas de compatibilidad con sistemas operativos o de almacenamiento, pudiendo también proporcionarles actualizaciones de manera transparente y automática.

En cuanto a la elección del PHP como lenguaje de programación y MySQL como gestor de base de datos, se ha hecho teniendo en cuenta la experiencia y conocimientos previos en estas tecnologías y en la extensa documentación disponible, que permitiría encontrar soluciones a cualquier problema que surja durante el desarrollo del proyecto.

Una vez se acceda a la aplicación, se mostrará una página de bienvenida donde, los responsables de la academia, podrán especificar información detallada (fechas, horarios, precios…) acerca de los cursos, eventos y torneos disponibles para que los clientes (alumnos y competidores) puedan consultarla en cualquier momento (Objetivo 10).

Desde el menú principal de la aplicación, los responsables de la academia tendrán acceso a diferentes enlaces desde donde podrá acceder a páginas que le permitan:

* Añadir nuevos usuarios a la aplicación, ver una lista con todos los usuarios del sistema o con los que encajen con una búsqueda, ver información detallada sobre cada uno de estos usuarios, modificar la información de los usuarios, eliminar a usuarios de la aplicación (Objetivo 1).
* Añadir nuevos espacios a la aplicación, ver una lista con todos los espacios del sistema o con los que encajen con una búsqueda, ver información detallada sobre cada uno de estos espacios, modificar la información de los espacios, eliminar espacios de la aplicación (Objetivo 2).
* Añadir nuevos cursos a la aplicación, ver una lista con todos los cursos del sistema o con los que encajen con una búsqueda, ver información detallada sobre cada uno de estos cursos, modificar la información de los cursos, eliminar cursos de la aplicación (Objetivo 3).
* Añadir nuevos torneos a la aplicación, ver una lista con todos los torneos del sistema o con los que encajen con una búsqueda, ver información detallada sobre cada uno de estos torneos, modificar la información de los torneos, eliminar torneos de la aplicación (Objetivo 4).

Desde la página de información de un torneo, los responsables de la academia, tendrán acceso a diferentes enlaces desde donde podrán acceder a páginas que les permitan:

* Añadir nuevos cuadros a la aplicación, ver una lista con todos los cuadros del sistema o con los que encajen con una búsqueda, ver información detallada sobre cada uno de estos cuadros, modificar la información de los cuadros, eliminar cuadros de la aplicación (Objetivo 4).

Desde la página de información de un torneo, los responsables de la academia, tendrán acceso a diferentes enlaces desde donde podrán acceder a páginas que les permitan:

* + Añadir nuevos partidos a la aplicación, ver una lista con todos los partidos del sistema o con los que encajen con una búsqueda, ver información detallada sobre cada uno de estos partidos, modificar la información de los partidos, eliminar partidos de la aplicación (Objetivo 4).
* Añadir nuevos eventos a la aplicación, ver una lista con todos los eventos del sistema o con los que encajen con una búsqueda, ver información detallada sobre cada uno de estos eventos, modificar la información de los eventos, eliminar a eventos de la aplicación (Objetivo 5).

Desde el menú principal de la aplicación, los responsables de la academia tendrán acceso a diferentes enlaces desde donde podrá acceder a páginas que le permitan:

* Ver un listado de las solicitudes de inscripción en los cursos o las que encajen con una búsqueda, ver la información de una solicitud, aceptar solicitudes, cancelar solicitudes, eliminar solicitudes (Objetivo 5).
* Ver un listado de las solicitudes de inscripción en los eventos o las que encajen con una búsqueda, ver la información de una solicitud, aceptar solicitudes, cancelar solicitudes, eliminar solicitudes (Objetivo 7).
* Ver un listado de las solicitudes de inscripción en los torneos o las que encajen con una búsqueda, ver la información de una solicitud, aceptar solicitudes, cancelar solicitudes, eliminar solicitudes (Objetivo 8).

Tanto los responsables de la academia como los clientes, tendrán disponible desde el menú principal de la aplicación, enlaces a páginas que les permitan redactar mensajes y enviárselos a otro usuario de la aplicación. Cuando un usuario reciba un mensaje, se le dará la opción de leerlo y eliminarlo (Objetivos 9 y 14).

Los clientes de la aplicación tendrán acceso desde el menú principal, a enlaces que les permitan:

* Ver un listado de las solicitudes de inscripción que han realizado en los cursos o las que encajen con una búsqueda y ver la información de una solicitud o eliminarla (Objetivo 11).

Desde la lista de cursos, tendrán la opción de:

* + Realizar una solicitud de inscripción a un curso (Objetivo 11).
* Ver un listado de las solicitudes de inscripción que han realizado en los eventos o las que encajen con una búsqueda y ver la información de una solicitud o eliminarla (Objetivo 12).

Desde la lista de eventos, tendrán la opción de:

* + Realizar una solicitud de inscripción a un evento (Objetivo 12).
* Ver un listado de las solicitudes de inscripción que han realizado en los torneos o las que encajen con una búsqueda y ver la información de una solicitud o eliminarla (Objetivo 13).

Desde la lista de torneos, tendrán la opción de:

* + Realizar una solicitud de inscripción a un torneo (Objetivo 13).

La aplicación contará con una gestión de permisos que impedirá a el acceso a usuarios no identificados en el sistema, así como restringir las funcionalidades a cada tipo de usuario y ocultar la información privada de cada uno de ellos (Objetivo 15).

Cuando un usuario trate de introducir información a en el sistema, la aplicación analizará cada uno de estos datos y determinará si son válidos o no, en el caso de que no lo sean, impedirá que el usuario lo introduzca y le pedirá que lo intente de nuevo correctamente (Objetivo 16).

Para el desarrollo del proyecto se seguirá el proceso de desarrollo del software RUP (Proceso Racional Unificado) que presenta tres características principales:

* Dirigido por casos de uso: Permiten especificar los requisitos del sistema.
* Centrado en la arquitectura: A medida que los casos de uso se especifican, se descubre más de la arquitectura, lo que lleva a su vez a la maduración de los casos de uso hasta disponer de una arquitectura estable.
* Iterativo e incremental: El trabajo se divide en partes más pequeñas que resultan en un incremento (crecimiento del producto).

RUP se repite a lo largo de una serie de ciclos que, al concluir, darán lugar a una versión final del producto y consta de cuatro fases:

* Inicio: Descripción del producto y análisis de negocio.
* Elaboración: Definición en detalle de todos los casos de uso.
* Construcción: Implementación del diseño realizado.
* Transición: Prueba del producto en un entorno real.

Se ha escogido RUP debido a su flexibilidad para seleccionar los componentes del proceso que mejor se ajustan a las necesidades del proyecto, a que puede ser aplicado tanto a pequeños como a grandes proyectos y a la extensa documentación disponible sobre esta metodología.

Para este proyecto, contaremos con dos iteraciones, la primera de ellas constará de XXX horas y las cuatro fases anteriormente descritas.

En la fase de inicio se definirá el problema y permitirá familiarizarse y preparar el entorno. Además, se llevarán a cabo el modelo de dominio y se definirán los requisitos de la aplicación.

En la fase de elaboración se definirán los casos de uso, se desarrollarán los diagramas de clases y de secuencia, se creará el modelo entidad relación para la base de datos y un prototipo falso de la aplicación.

En la fase de construcción, se creará la base de datos y se desarrollarán las funcionalidades de la aplicación, realizando las pruebas necesarias para comprobar su correcto funcionamiento.

La fase de pruebas, finalizará la primera iteración y en ella se realizarán las pruebas globales de la aplicación.

La segunda iteración, constará de dos fases, construcción y pruebas, en la fase de construcción se corregirán todos los errores de implementación que hayan surgido durante la primera iteración detectados en las pruebas globales de la primera iteración y en la fase de pruebas, se llevarán a cabo las pruebas globales de la aplicación con las correcciones añadidas en la fase anterior.

Durante todas las fases de cada una de las iteraciones, se irá completando la documentación del proyecto, desarrollándose por tanto desde el inicio hasta el final del proyecto de manera continua.

# Planificación y Seguimiento

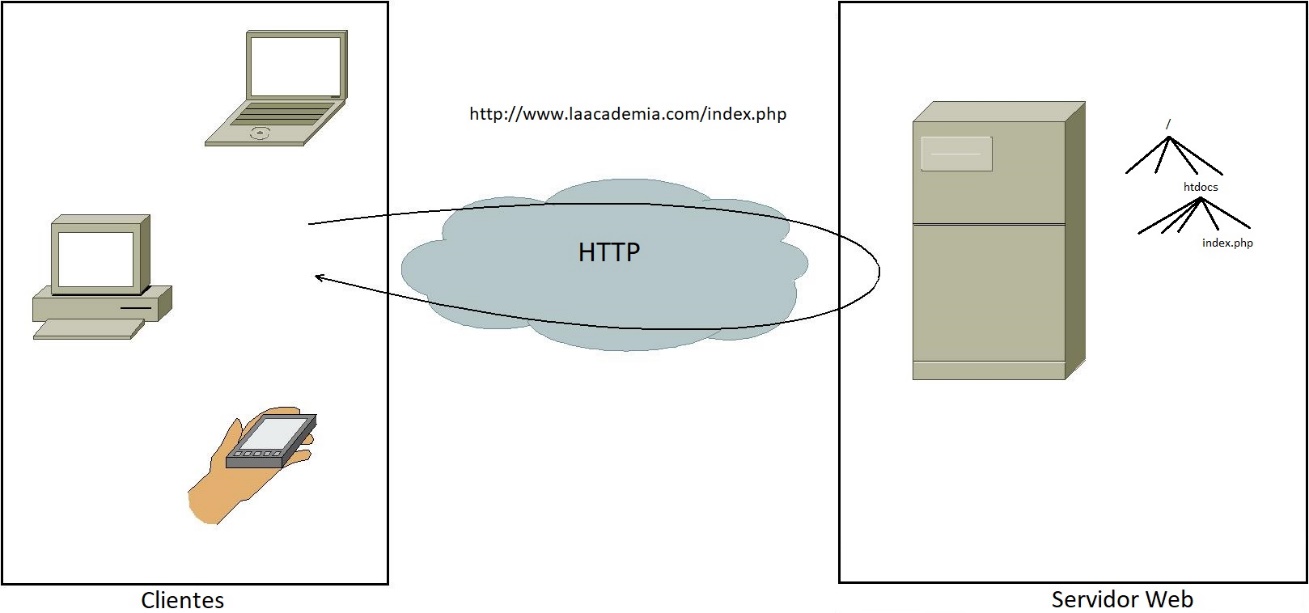
La limitación temporal de este proyecto es de 300 horas, para cumplir este requisito y no exceder la duración máxima, se ha realizado una planificación estimando una dedicación diaria de 4 horas de trabajo de lunes a viernes, lo que se traduce en 20 horas semanales de trabajo a lo largo de 15 semanas de trabajo.

Siguiendo con la metodología descrita en el apartado anterior, se ha distribuido el tiempo disponible en dos iteraciones y se le ha asignado a cada fase y a cada tarea la duración como se ve en la siguiente tabla:

|  |  |  |
| --- | --- | --- |
| Fase | Tarea | Estimación (horas) |
| Iteración 1 | | |
| Inicio | Definición del problema | 4 |
| Familiarización y preparación del entorno | 4 |
| Modelo de Dominio | 1 |
| Análisis de Requisitos | 3 |
| Documentación | 8 |
| Elaboración | Casos de Uso | 8 |
| Diagramas de Clases | 8 |
| Diagramas de Secuencia | 28 |
| Modelo Entidad-Relación | 8 |
| Prototipo Falso | 12 |
| Documentación | 8 |
| Construcción | Implementación de la Base de Datos | 4 |
| Implementación de las funcionalidades | 88 |
| Pruebas funcionales | 12 |
| Documentación | 24 |
| Pruebas | Pruebas globales | 8 |
| Documentación | 4 |
| Iteración 2 | | |
| Construcción | Implementación de correcciones en las funcionalidades y la Base de Datos | 32 |
| Pruebas funcionales | 8 |
| Documentación | 16 |
| Pruebas | Pruebas globales | 8 |
| Documentación | 4 |
| Total |  | 300 |

# Arquitectura

La arquitectura general de la aplicación se puede ver en la imagen siguiente:



La arquitectura estará basada en el modelo cliente-servidor que consta de tres componentes principales:

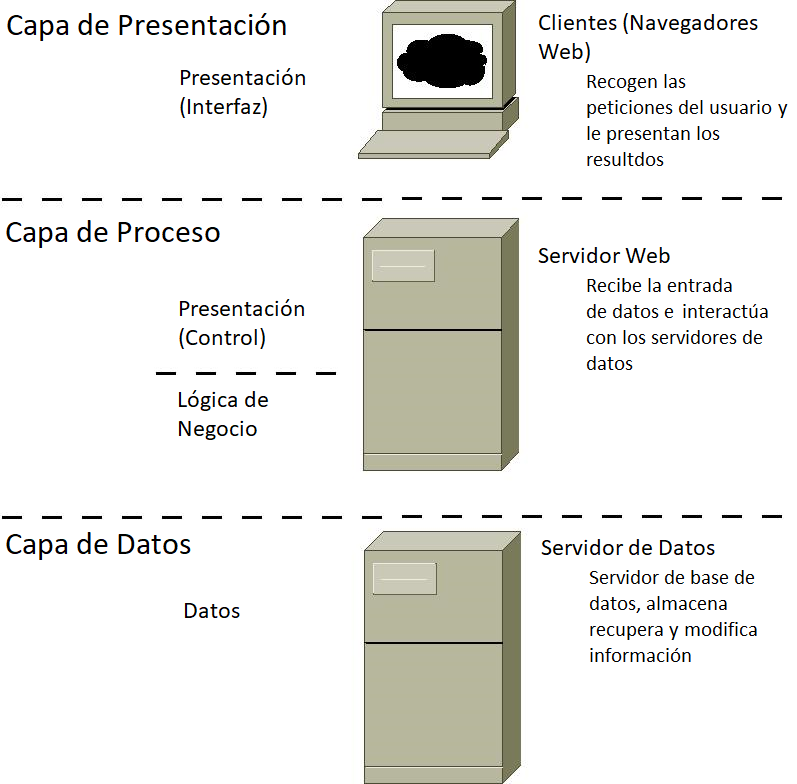
1. Servidor web: Será el encargado de distribuir las páginas de información formateada a los clientes que lo soliciten.
2. Conexión de red: Las peticiones de los clientes se realizarán a través de peticiones HTTP a través de la red.
3. Clientes: Realizan las peticiones al servidor y visualizan la información por medio de navegadores web

Al tratarse de una aplicación web, contaremos con tres elementos fundamentales:

1. Lógica de negocio: Conjunto de operaciones que permiten proveer el servicio.
2. Administración de los datos: Manipulación de bases de datos y archivos.
3. Interfaz: Presenta la información a los usuarios por medio de navegadores web.

Estos tres componentes serán modelados mediante un modelo de tres capas denominado modelo-vista-controlador, cada una de las capas, representa un elemento que procesa información y éstas son:

1. Capa de presentación: Recoge la información del usuario y la envía al servidor, posteriormente reciben la información de respuesta del servidor y la presentan al usuario
2. Capa de proceso: Recibe la entrada de datos de la capa de presentación, interactúa con la capa de datos para realizar operaciones y devuelve los resultados a la capa de presentación
3. Capa de datos: Almacena, recupera y modifica los datos.



Para llevar a cabo este modelo se ha empleado un framework que facilite el desarrollo del proyecto y permita cumplir con los requisitos de diseño de cualquier aplicación web, éstos son:

* Separación de la lógica de negocio y la presentación para una mayor flexibilidad y división de responsabilidades.
* Punto de entrada común de las peticiones.
* Plantillas de página que favorezcan la flexibilidad con elementos comunes de las vistas
* Internacionalización, que permita disponer de una aplicación multi-idioma.
* Seguir el patrón PRG (POST, REDIRECT, GET):
  + Nunca mostrar páginas como respuesta a POST
  + Siempre cargar páginas usando GET
  + Navegar de POST a GET usando REDIRECT

# Tecnologías e Integración de Productos de Terceros

A continuación, se enumeran las distintas tecnologías utilizadas durante el desarrollo del proyecto, así como una breve descripción para cada una de ellas.

**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.

Se ha escogido este lenguaje de programación por la experiencia previa con otros proyectos, la extensa documentación disponible gracias a su gran comunidad y por estar orientado a objetos y a al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en bases de datos.

**HTML**

HTML, que significa Lenguaje de Marcado para Hipertextos (HyperText Markup Language) es el elemento de construcción más básico de una página web y se usa para crear y representar visualmente una página web. Determina el contenido de la página web, pero no su funcionalidad.

Se ha escogido esta tecnología por su facilidad de uso y por ser un estándar aceptado por todos los navegadores web, permitiendo mostrar texto de forma estructurada.

**CSS**

CSS es el lenguaje para describir la presentación de páginas web, incluidos los colores, el diseño y las fuentes. Le permite a uno adaptar la presentación a diferentes tipos de dispositivos, como pantallas grandes, pantallas pequeñas o impresoras.

CSS es independiente de HTML y se puede usar con cualquier lenguaje de marcado basado en XML. La separación de HTML de CSS facilita el mantenimiento de sitios, el intercambio de hojas de estilo entre páginas y la personalización de páginas en diferentes entornos

XAMPP (MySQL, APACHE).

Se ha utilizado esta tecnología debido a su flexibilidad para modificar el estilo de la web, permitiendo separar los archivos de contenido de los de presentación, lo que nos permite unificar el diseño del sitio web, no necesitando duplicar código.

**MySQL**

MySQL es la base de datos de código abierto más popular del mundo, proporciona sistemas de almacenamiento transaccionales y no transaccionales donde las funciones SQL están implementadas usando una librería altamente optimizada.

Se ha escogido esta tecnología debido a su gran comunidad y documentación, además su seguridad y escalabilidad también han sido importantes al tomar esta elección, ya que soporta grandes bases de datos y proporciona un sistema de privilegios y contraseñas flexible y seguro.

**PhpMyAdmin**

PhpMyAdmin es una herramienta de software libre escrita en PHP, destinada a manejar la administración de MySQL a través de la Web, es compatible con una amplia gama de operaciones en MySQL y MariaDB.

Las operaciones de uso frecuente (administración de bases de datos, tablas, columnas, relaciones, índices, usuarios, permisos, etc.) se pueden realizar a través de la interfaz de usuario, y también tiene la capacidad de ejecutar directamente cualquier declaración de SQL.

Se ha escogido esta herramienta debido a la sencillez de su interfaz gráfica, que permite trabajar más cómodamente con bases de datos y reducir tiempos de trabajo.

**MySQL Workbech**

MySQL Workbench es una herramienta visual unificada para arquitectos de bases de datos, desarrolladores y DBA. MySQL Workbench proporciona modelado de datos, desarrollo de SQL y herramientas integrales de administración para la configuración del servidor, administración de usuarios y respaldo.

Se ha escogido esta herramienta por la posibilidad de desarrollar diagramas de entidad-relación de manera visual e intuitiva y por permitir generar scripts SQL a partir del modelo creado y viceversa.

**Java Script**

JavaScript es un robusto lenguaje de programación que puede ser aplicado a un documento HTML y usado para crear interactividad dinámica en los sitios web.

Se ha escogido esta tecnología por su compatibilidad con la mayoría de navegadores y por ser un complemento a otras tecnologías como HTML, permitiendo añadir nuevas funcionalidades como la validación de datos en cliente.

**Visual Paradigm**

Visual Paradigm es una herramienta de modelado UML.

Se ha escogido esta herramienta por la compatibilidad con los estándares UML a la hora de desarrollar los diagramas necesarios para el proyecto y su capacidad para generar informes y documentación.

**GitHub**

GitHub es una plataforma de desarrollo colaborativo para alojar proyectos usando es sistema de control de versiones Git.

Se ha escogido esta herramienta debido a que permite alojar el repositorio de código del proyecto en la nube, por poseer un visor de código en el que consultar desde el navegador el contenido de cualquier fichero y por permitir un control de versiones con el que registrar los cambios realizados durante su desarrollo y poder revertirlos.

**Microsoft Word**

Microsoft Word es una herramienta orientada al procesamiento de textos.

Se ha escogido esta herramienta por su idoneidad a la hora de realizar la documentación del proyecto, pudiendo crear un documento bien estructurado para el que se proporcionan facilidades para la inserción de imágenes, tablas, índices y muchos otros contenidos.

**Microsoft Project**

Microsoft Project es una herramienta de trabajo para administradores y jefes de proyectos, utilizada para organizar y seguir las tareas de forma eficaz, para evitar así los retrasos y no salirse del presupuesto previsto.

Se ha escogido esta herramienta por la facilidad que proporciona a la hora de realizar diagramas de Gantt para describir el tiempo de dedicación al proyecto desarrollado.

**Navegadores web (Chrome y Firefox)**

Chrome y Firefox son aplicaciones que permiten el acceso a la Web, interpretando la información de distintos tipos de archivos y sitios web para que éstos puedan ser visualizados.

La funcionalidad básica de un navegador web es permitir la visualización de documentos de texto, con recursos multimedia incrustados.

Se han escogido estos navegadores por ser dos de los más extendidos entre el público general y por su respeto en los estándares de la W3C (World Wide Web Consorcium) para asegurar el funcionamiento correcto de los sitios web

**Atom**

Atom es un editor de texto personalizable.

Se ha escogido este editor por su compatibilidad con los lenguajes anteriormente descritos, como PHP, HTML, CSS o JavaScript, además de su integración con Git, su personalización y la gran cantidad de funcionalidades extra que proporciona gracias a diferentes plugins.

**BOOTSTRAP**

Bootstrap es un kit de herramientas de código abierto para desarrollar con HTML, CSS y JS.

Se ha empleado este framework por el empleo de componentes básicos para el desarrollo web como HTML, CSS y JavaScript. Permite simplificar el proceso de maquetación web, proporcionando un diseño responsive.

La forma en que se ha integrado Bootstrap al proyecto es mediante Bootstrap CDN, que evita descargar el framework y tener que transferirlo a nuestro servidor.

Para implementarlo, se debe indicar el enlace de la plantilla de estilo en nuestras páginas HTML, así como los enlaces a los scripts que necesitarán algunos de los componentes de Bootstrap que requieren del uso de JavaScript.

**Framework MVC**

Framework modelo-vista-controlador escrito en PHP que proporciona una estructura básica para el desarrollo de una aplicación web.

Se ha escogido este framework debido a su fácil curva de aprendizaje y a que proporciona una estructura básica obliga a cumplir los requisitos esenciales que debe cumplir una aplicación web de manera correcta.

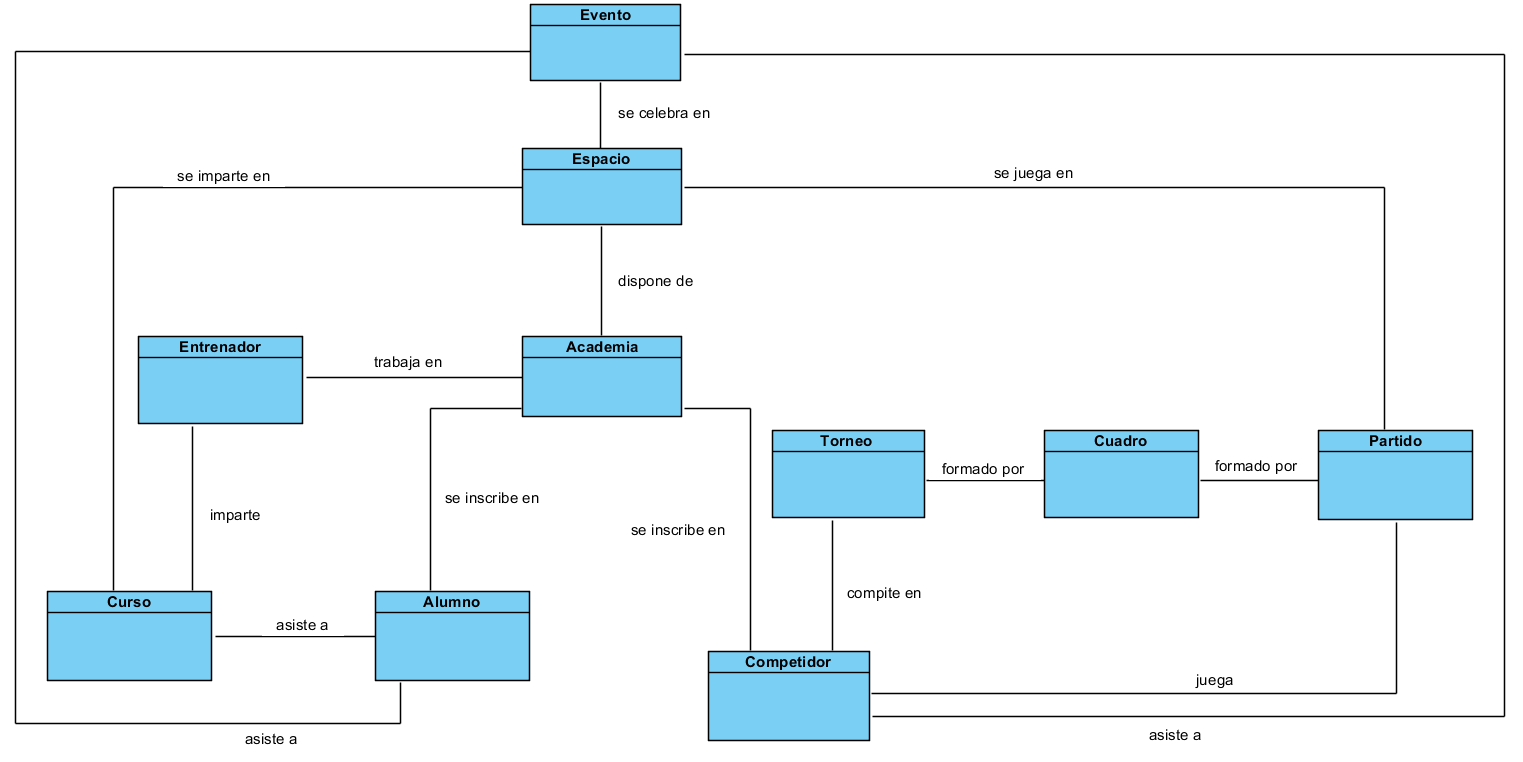
Para implementar este framework, es necesario un servidor web Apache 2, junto con MySQL 5 y PHP 5. Se debe crear una base de datos que permita acceder a la aplicación base y posteriormente completar o añadir las nuevas funcionalidades necesarios para el proyecto.

# Especificación y Análisis de Requisitos

En esta sección se muestra el modelo de dominio, los requisitos funcionales y no funcionales y los diagramas de casos de uso de la aplicación, que son básicos en la metodología de desarrollo RUP.

## Modelo de Dominio

Se presenta a continuación un modelo de dominio que permita representar el vocabulario y conceptos clave del problema, así como identificar las entidades involucradas y sus relaciones.



## Análisis de Requisitos

Clasificamos los requisitos en funcionales y no funcionales, en los funcionales, se declararán los servicios que proveerá el sistema, mientras que en los no funcionales se definirán criterios para evaluar la calidad de la aplicación.

### Requisitos no Funcionales

RNF1 Debe funcionar correctamente con independencia del sistema operativo y navegador en el que se utilice.

RNF2 Debe tener una interfaz adaptada a dispositivos móviles.

RNF3 Debe ser fácil de entender y emplear por los usuarios.

RNF4 La curva de aprendizaje para que el usuario pueda manejar la aplicación con soltura debe ser baja.

RNF5 Debe contar con manuales de usuario.

### Requisitos Funcionales

RF1 Autenticar Usuario

Actores involucrados: Administrador, Entrenador, Alumno, Competidor

Descripción: Permite al usuario acceder al sistema introduciendo sus credenciales de acceso (nombre de usuario y contraseña).

RF2 Gestionar Usuario

RF2.1 Listar Usuarios

Actores involucrados: Administrador, Entrenador.

Descripción: Permite listar todos los usuarios del sistema.

RF2.2 Ver Usuario

Actores involucrados: Administrador, Entrenador.

Descripción: Permite ver en detalle los datos del usuario seleccionado.

RF2.3 Crear Usuario

Actores involucrados: Administrador.

Descripción: Permite crear un nuevo usuario en el sistema.

RF2.4 Modificar Usuario

Actores involucrados: Administrador.

Descripción: Permite modificar el usuario seleccionado en el sistema.

RF2.5 Eliminar Usuario

Actores involucrados: Administrador.

Descripción: Permite eliminar el usuario seleccionado en el sistema.

RF2.6 Buscar Usuarios

Actores involucrados: Administrador, Entrenador.

Descripción: Permite buscar usuarios que encajen con los parámetros de búsqueda indicados.

RF2.7 Ver Perfil

Actores involucrados: Administrador, Entrenador, Alumno, Competidor.

Descripción: Permite ver en detalle los datos del usuario que haya iniciado sesión en el sistema.

RF3 Gestionar Espacios

RF3.1 Listar Espacios

Actores involucrados: Administrador, Entrenador

Descripción: Permite listar todos los espacios del sistema.

RF3.2 Ver Espacio

Actores involucrados: Administrador, Entrenador

Descripción: Permite ver en detalle los datos del espacio seleccionado.

RF3.3 Crear Espacio

Actores involucrados: Administrador

Descripción: Permite crear un nuevo espacio en el sistema.

RF3.4 Modificar Espacio

Actores involucrados: Administrador

Descripción: Permite modificar el espacio seleccionado en el sistema.

RF3.5 Eliminar Espacio

Actores involucrados: Administrador

Descripción: Permite eliminar el espacio seleccionado en el sistema.

RF3.6 Buscar Espacios

Actores involucrados: Administrador, Entrenador

Descripción: Permite buscar espacios que encajen con los parámetros de búsqueda indicados.

RF4 Gestionar Cursos

RF4.1 Listar Cursos

Actores involucrados: Administrador, Entrenador, Alumno

Descripción: Permite listar todos los cursos del sistema.

RF4.2 Ver Curso

Actores involucrados: Administrador, Entrenador, Alumno

Descripción: Permite ver en detalle los datos del curso seleccionado.

RF4.3 Crear Curso

Actores involucrados: Administrador

Descripción: Permite crear un nuevo curso en el sistema.

RF4.4 Modificar Curso

Actores involucrados: Administrador, Entrenador

Descripción: Permite modificar el curso seleccionado en el sistema.

RF4.5 Eliminar Curso

Actores involucrados: Administrador

Descripción: Permite eliminar el curso seleccionado en el sistema.

RF4.6 Buscar Cursos

Actores involucrados: Administrador, Entrenador, Alumno

Descripción: Permite buscar cursos que encajen con los parámetros de búsqueda indicados.

RF5 Gestionar Reservas de Cursos

RF5.1 Listar Reservas de Cursos

Actores involucrados: Administrador, Alumno

Descripción: Permite listar todas las reservas de cursos del sistema.

RF5.2 Ver Reserva de Curso

Actores involucrados: Administrador, Alumno

Descripción: Permite ver en detalle los datos de la reserva seleccionada.

RF5.3 Confirmar Reserva de Curso

Actores involucrados: Administrador

Descripción: Permite confirmar la reserva hecha por un usuario para un curso.

RF5.4 Eliminar Reserva de Curso

Actores involucrados: Administrador, Alumno

Descripción: Permite eliminar la reserva de un curso seleccionada en el sistema.

RF5.5 Buscar Reservas de Cursos

Actores involucrados: Administrador, Alumno

Descripción: Permite buscar reservas de cursos que encajen con los parámetros de búsqueda indicados.

RF5.6 Crear Reserva de Curso

Actores involucrados: Alumno

Descripción: Permite crear una nueva reserva para el curso seleccionado.

RF6 Gestionar Eventos

RF6.1 Listar Eventos

Actores involucrados: Administrador, Entrenador, Alumno, Competidor

Descripción: Permite listar todos los eventos del sistema.

RF6.2 Ver Evento

Actores involucrados: Administrador, Entrenador, Alumno, Competidor

Descripción: Permite ver en detalle los datos del evento seleccionado.

RF6.3 Crear Evento

Actores involucrados: Administrador

Descripción: Permite crear un nuevo evento en el sistema.

RF6.4 Modificar Evento

Actores involucrados: Administrador, Entrenador

Descripción: Permite modificar el evento seleccionado en el sistema.

RF6.5 Eliminar Evento

Actores involucrados: Administrador

Descripción: Permite eliminar el evento seleccionado en el sistema.

RF6.6 Buscar Eventos

Actores involucrados: Administrador, Entrenador, Alumno, Competidor

Descripción: Permite buscar eventos que encajen con los parámetros de búsqueda indicados.

RF7 Gestionar Reservas Eventos

RF7.1 Listar Reservas de Eventos

Actores involucrados: Administrador, Alumno, Competidor

Descripción: Permite listar todas las reservas de eventos del sistema.

RF7.2 Ver Reserva de Evento

Actores involucrados: Administrador, Alumno, Competidor

Descripción: Permite ver en detalle los datos de la reserva seleccionada.

RF7.3 Confirmar Reserva de Evento

Actores involucrados: Administrador

Descripción: Permite confirmar la reserva hecha por un usuario para un evento.

RF7.4 Eliminar Reserva de Evento

Actores involucrados: Administrador, Alumno, Competidor

Descripción: Permite eliminar la reserva de un evento seleccionada en el sistema.

RF7.5 Buscar Reservas de Eventos

Actores involucrados: Administrador, Alumno, Competidor

Descripción: Permite buscar reservas de eventos que encajen con los parámetros de búsqueda indicados.

RF7.6 Crear Reserva de Evento

Actores involucrados: Alumno, Competidor

Descripción: Permite crear una nueva reserva para el evento seleccionado

RF8 Gestionar Torneos

RF8.1 Listar Torneos

Actores involucrados: Administrador, Entrenador, Competidor

Descripción: Permite listar todos los torneos del sistema.

RF8.2 Ver Torneo

Actores involucrados: Administrador, Entrenador, Competidor

Descripción: Permite ver en detalle los datos del torneo seleccionado.

RF8.3 Crear Torneo

Actores involucrados: Administrador

Descripción: Permite crear un nuevo torneo en el sistema.

RF8.4 Modificar Torneo

Actores involucrados: Administrador

Descripción: Permite modificar el torneo seleccionado en el sistema.

RF8.5 Eliminar Torneo

Actores involucrados: Administrador

Descripción: Permite eliminar el torneo seleccionado en el sistema.

RF8.6 Buscar Torneos

Actores involucrados: Administrador, Entrenador, Competidor

Descripción: Permite buscar torneos que encajen con los parámetros de búsqueda indicados.

RF9 Gestionar Reservas Torneos

RF9.1 Listar Reservas de Torneos

Actores involucrados: Administrador, Competidor

Descripción: Permite listar todas las reservas de torneos del sistema.

RF9.2 Ver Reserva de Torneo

Actores involucrados: Administrador, Competidor

Descripción: Permite ver en detalle los datos de la reserva seleccionada de un torneo.

RF9.3 Confirmar Reserva de Torneo

Actores involucrados: Administrador

Descripción: Permite confirmar la reserva hecha por un usuario para un torneo.

RF9.4 Eliminar Reserva de Torneo

Actores involucrados: Administrador, Competidor

Descripción: Permite eliminar la reserva de un torneo seleccionada en el sistema.

RF9.5 Buscar Reservas de Torneos

Actores involucrados: Administrador, Competidor

Descripción: Permite buscar reservas de torneos que encajen con los parámetros de búsqueda indicados.

RF9.6 Crear Reserva de Torneo

Actores involucrados: Competidor

Descripción: Permite crear una nueva reserva para el torneo seleccionado

RF10 Gestionar Cuadros

RF10.1 Listar Cuadros

Actores involucrados: Administrador, Entrenador, Competidor

Descripción: Permite listar todos los cuadros de un torneo.

RF10.2 Ver Cuadro

Actores involucrados: Administrador, Entrenador, Competidor

Descripción: Permite ver en detalle los datos del cuadro seleccionado.

RF10.3 Crear Cuadro

Actores involucrados: Administrador

Descripción: Permite crear un nuevo cuadro en el torneo seleccionado.

RF10.4 Modificar Cuadro

Actores involucrados: Administrador

Descripción: Permite modificar el cuadro seleccionado en el sistema.

RF10.5 Eliminar Cuadro

Actores involucrados: Administrador

Descripción: Permite eliminar el cuadro seleccionado en el sistema.

RF11 Gestionar Partidos

RF11.1 Listar Partidos

Actores involucrados: Administrador, Entrenador, Competidor

Descripción: Permite listar todos los partidos de un cuadro.

RF11.2 Ver Partido

Actores involucrados: Administrador, Entrenador, Competidor

Descripción: Permite ver en detalle los datos del partido seleccionado.

RF11.3 Crear Partido

Actores involucrados: Administrador

Descripción: Permite crear un nuevo partido en el cuadro seleccionado.

RF11.4 Modificar Partido

Actores involucrados: Administrador, Entrenador

Descripción: Permite modificar el partido seleccionado en el sistema.

RF11.5 Eliminar Partido

Actores involucrados: Administrador

Descripción: Permite eliminar el partido seleccionado en el sistema.

RF11.6 Buscar Partidos

Actores involucrados: Administrador, Entrenador, Competidor

Descripción: Permite buscar partidos que encajen con los parámetros de búsqueda indicados.

RF12 Gestionar Notificaciones

RF12.1 Listar Notificaciones

Actores involucrados: Administrador, Entrenador, Alumno, Competidor.

Descripción: Permite listar todas las notificaciones que ha recibido.

RF12.2 Ver Notificación

Actores involucrados: Administrador, Entrenador, Alumno, Competidor.

Descripción: Permite ver en detalle la notificación seleccionada.

RF12.3 Enviar Notificación

Actores involucrados: Administrador, Entrenador, Alumno, Competidor.

Descripción: Permite enviar una notificación a otro usuario.

RF12.4 Eliminar Notificación

Actores involucrados: Administrador, Entrenador, Alumno, Competidor.

Descripción: Permite eliminar la notificación seleccionada.

RF12.5 Buscar Notificaciones

Actores involucrados: Administrador, Entrenador, Alumno, Competidor.

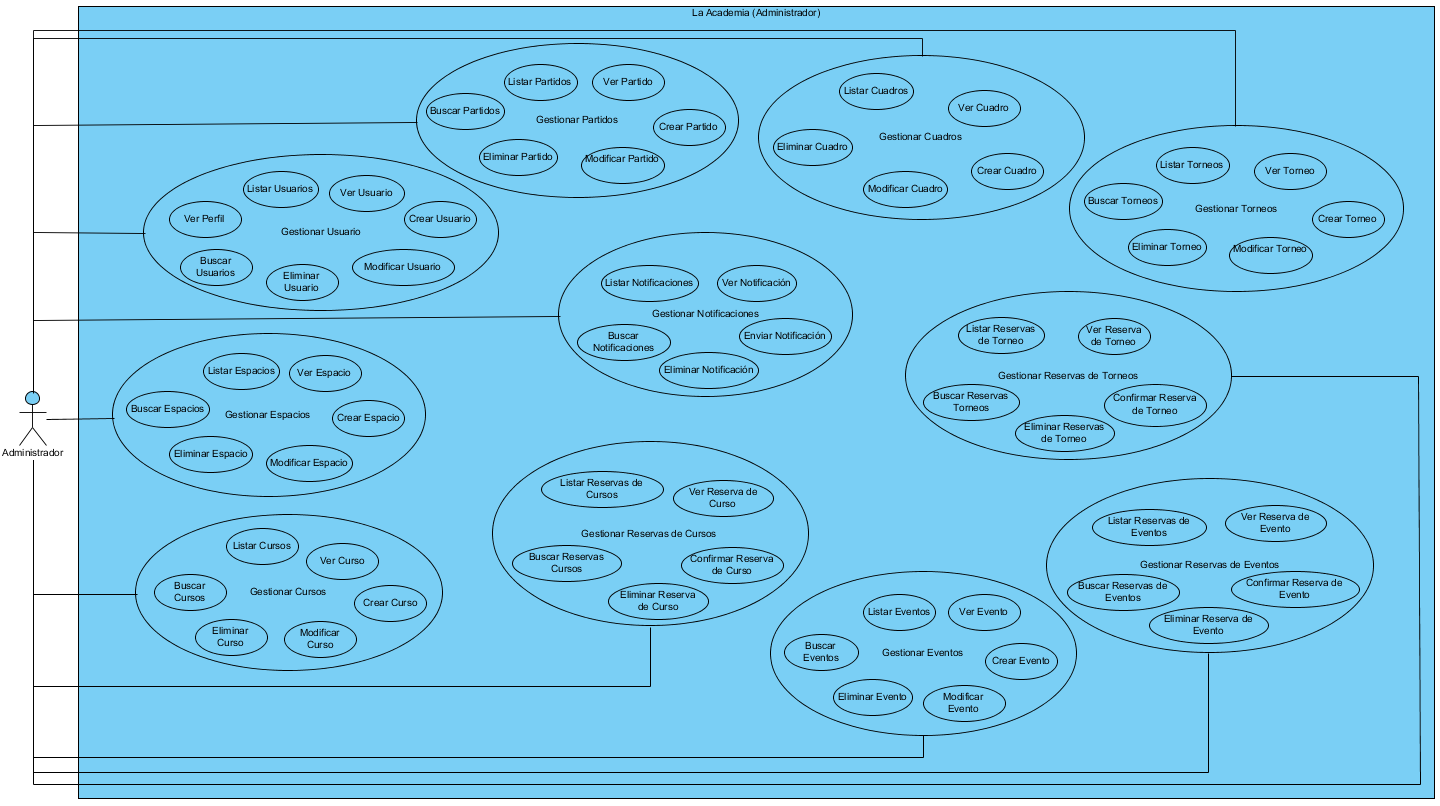
Descripción: Permite buscar notificaciones que encajen con los parámetros de búsqueda indicados.

## Casos de Uso

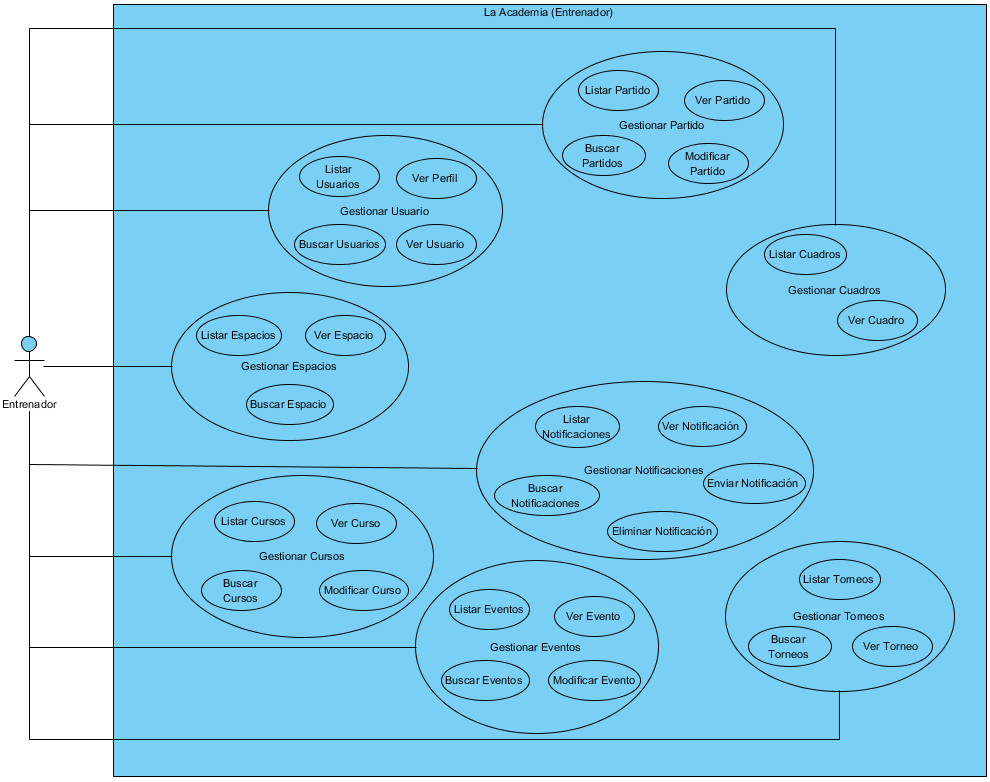
Se presentan los diagramas de casos de uso que, para facilitar su comprensión, se ha creado uno para cada actor identificado en el sistema.

Los actores identificados en el sistema son los siguientes:

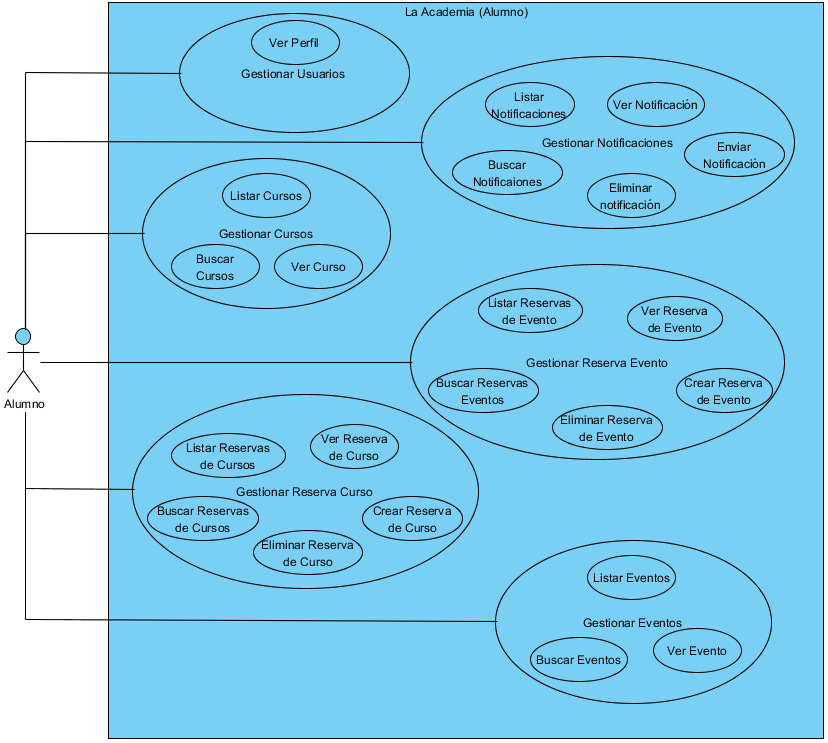
**Administrador**: Se encargará de gestionar cualquier funcionalidad del sistema.



**Entrenador**: Trabajador de la academia que se encarga de impartir sus cursos ayudándose de esta herramienta. Podrá consultar información sobre cualquier funcionalidad del sistema y además servirá de apoyo al administrador a la hora de gestionar cursos, eventos y partidos en los torneos. También podrá enviar y recibir notificaciones a cualquier otro usuario.



**Alumno**: Cliente de la academia que usará la aplicación para ver la oferta de cursos y eventos que ofrece la academia para apuntarse a los que le interesen. Además, podrá consultar su perfil y enviar notificaciones a otros usuarios



**Competidor**: Cliente de la academia que usará la aplicación para ver los torneos y eventos que ofrece la academia para apuntarse a los que le interesen. Además, podrá consultar su perfil y enviar notificaciones a otros usuarios



A continuación, se presentan las tablas con las descripciones de cada caso de uso:

|  |  |  |  |
| --- | --- | --- | --- |
| ID Caso de Uso | **Listar Usuarios** | | |
| Actor | Administrador/Entrenador | | |
| Descripción | El usuario puede ver una lista con todos los usuarios del sistema | | |
| Precondiciones | Estar logueado en el sistema | | |
| Postcondiciones | Ninguna | | |
| Flujo normal de eventos |  | Acción del Actor | Respuesta del Sistema |
| 1 | Solicita consultar usuarios |  |
| 2 |  | Recupera todos los usuarios y los muestra en la interfaz |

|  |  |  |  |
| --- | --- | --- | --- |
| ID Caso de Uso | **Ver Usuario** | | |
| Actor | Administrador/Entrenador | | |
| Descripción | El usuario puede ver en detalle los datos de un usuario del sistema | | |
| Precondiciones | Estar logueado en el sistema y en la vista de Listar Usuarios | | |
| Postcondiciones | Ninguna | | |
| Flujo normal de eventos |  | Acción del Actor | Respuesta del Sistema |
| 1 | Solicita consultar los datos de un usuario en particular |  |
| 2 |  | Recupera los datos del usuario seleccionado y los muestra en la interfaz |

|  |  |  |  |
| --- | --- | --- | --- |
| ID Caso de Uso | **Crear Usuario** | | |
| Actor | Administrador | | |
| Descripción | El usuario puede crear un nuevo usuario en el sistema | | |
| Precondiciones | Estar logueado en el sistema y en la vista de Listar Usuarios | | |
| Postcondiciones | Se actualiza la base de datos | | |
| Flujo normal de eventos |  | Acción del Actor | Respuesta del Sistema |
| 1 | Solicita crear un nuevo usuario |  |
| 2 |  | Muestra formulario con los campos requeridos para el nuevo usuario |
| 3 | Introduce los datos |  |
| 4 |  | Comprueba que los datos introducidos por el usuario son correctos [A1] |
| 5 |  | Almacena los datos y confirma la operación |
| [A1] |  | Acción del Actor | Respuesta del Sistema |
| 1 |  | Si alguno de los datos introducidos por el usuario es incorrecto, se envía un mensaje de error y solicita que se rellene de nuevo |
| 2 | Introduce los datos |  |

|  |  |  |  |
| --- | --- | --- | --- |
| ID Caso de Uso | **Crear Usuario** | | |
| Actor | Administrador | | |
| Descripción | El usuario puede crear un nuevo usuario en el sistema | | |
| Precondiciones | Estar logueado en el sistema y en la vista de Listar Usuarios | | |
| Postcondiciones | Se actualiza la base de datos | | |
| Flujo normal de eventos |  | Acción del Actor | Respuesta del Sistema |
| 1 | Solicita crear un nuevo usuario |  |
| 2 |  | Muestra formulario con los campos requeridos para el nuevo usuario |
| 3 | Introduce los datos |  |
| 4 |  | Comprueba que los datos introducidos por el usuario son correctos [A1] |
| 5 |  | Almacena los datos y confirma la operación |
| [A1] |  | Acción del Actor | Respuesta del Sistema |
| 1 |  | Si alguno de los datos introducidos por el usuario es incorrecto, se envía un mensaje de error y solicita que se rellene de nuevo |
| 2 | Introduce los datos |  |

|  |  |  |  |
| --- | --- | --- | --- |
| ID Caso de Uso | **Eliminar Usuario** | | |
| Actor | Administrador | | |
| Descripción | El usuario puede eliminar a un usuario del sistema realizando un borrado lógico de los datos | | |
| Precondiciones | Estar logueado en el sistema y y en la vista de Listar Usuarios | | |
| Postcondiciones | Se actualiza la base de datos | | |
| Flujo normal de eventos |  | Acción del Actor | Respuesta del Sistema |
| 1 | Solicita eliminar un usuario |  |
| 2 |  | Muestra la información del usuario seleccionado |
| 3 | Confirma la operación |  |
| 4 |  | Realiza el borrado lógico del usuario |

|  |  |  |  |
| --- | --- | --- | --- |
| ID Caso de Uso | **Buscar Usuarios** | | |
| Actor | Administrador/Entrenador | | |
| Descripción | El usuario puede buscar usuarios que encajen con los parámetros de búsqueda indicados. | | |
| Precondiciones | Estar logueado en el sistema y en la vista de Listar Usuarios | | |
| Postcondiciones | Ninguna | | |
| Flujo normal de eventos |  | Acción del Actor | Respuesta del Sistema |
| 1 | Solicita buscar usuarios |  |
| 2 |  | Muestra formulario con los campos requeridos para buscar los usuarios |
| 3 | Introduce datos |  |
| 4 |  | Recupera los usuarios que coincidan con los parámetros de búsqueda y los muestra en la interfaz |

|  |  |  |  |
| --- | --- | --- | --- |
| ID Caso de Uso | **Ver Perfil** | | |
| Actor | Administrador/Entrenador/Alumno/Competidor | | |
| Descripción | El usuario puede ver en detalle sus datos | | |
| Precondiciones | Estar logueado en el sistema | | |
| Postcondiciones | Ninguna | | |
| Flujo normal de eventos |  | Acción del Actor | Respuesta del Sistema |
| 1 | Solicita ver su perfil de usuario |  |
| 2 |  | Recupera los datos del usuario y los muestra en la interfaz |

# Diseño del Software

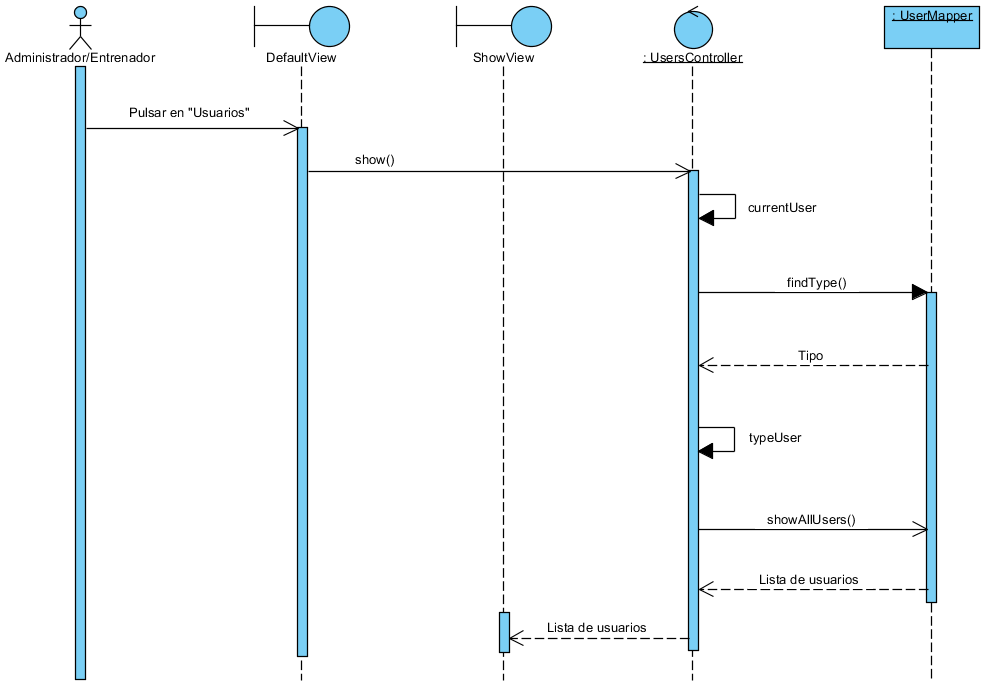
En esta sección se presentan tanto el diseño dinámico, que nos permitirá describir el comportamiento del sistema a lo largo del tiempo, como el diseño estático de la aplicación, que recabará los requerimientos y le dará forma a la estructura del sistema.

## Diseño Dinámico

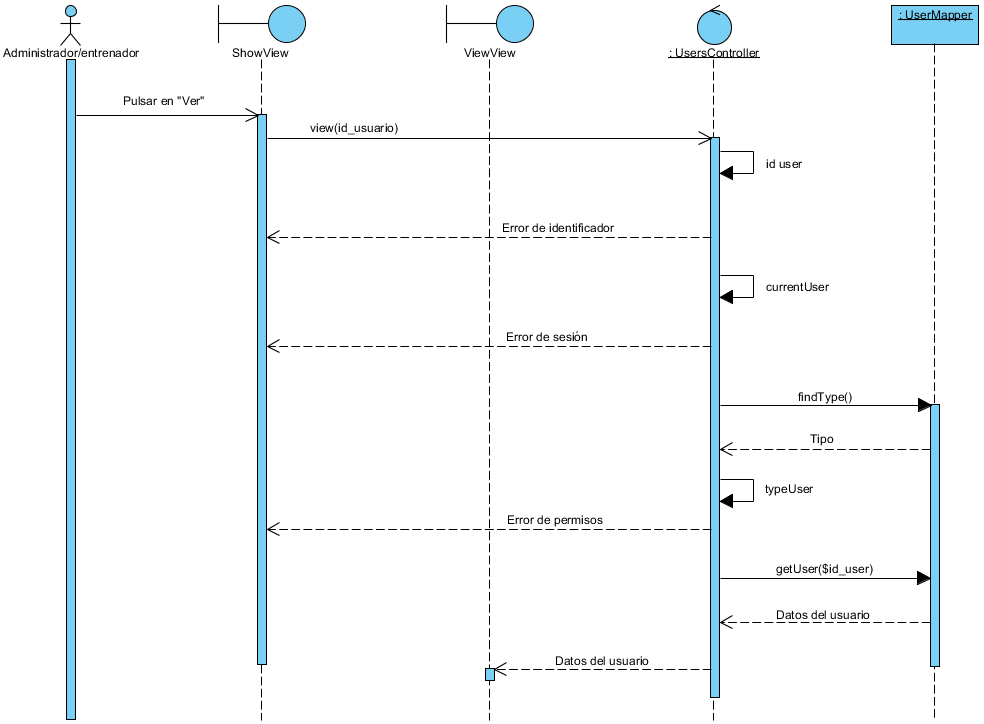
Se muestran los diagramas de secuencia detallados, que nos permiten ver las interacciones de los actores con la aplicación y el flujo de ejecución del programa para cada funcionalidad.

**Gestionar Usuarios**

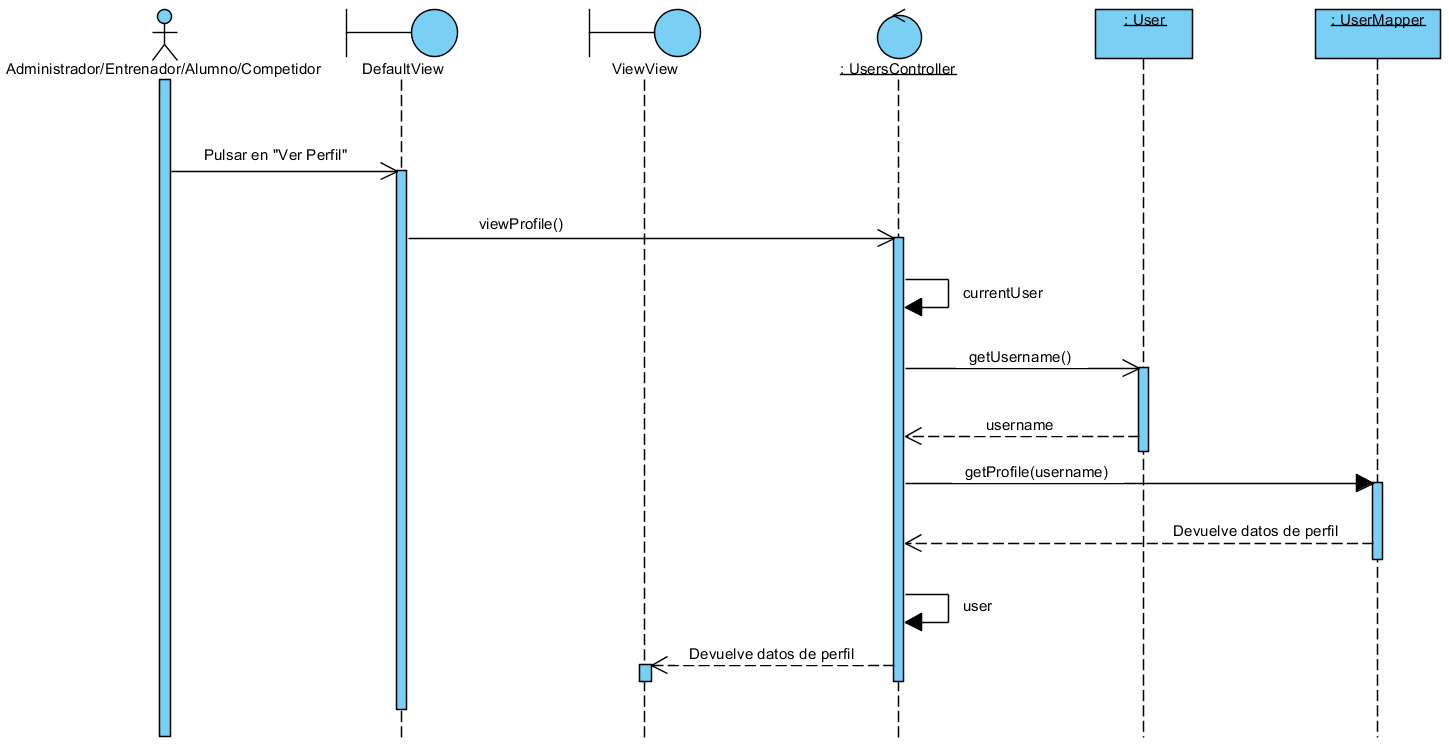
Listar Usuarios



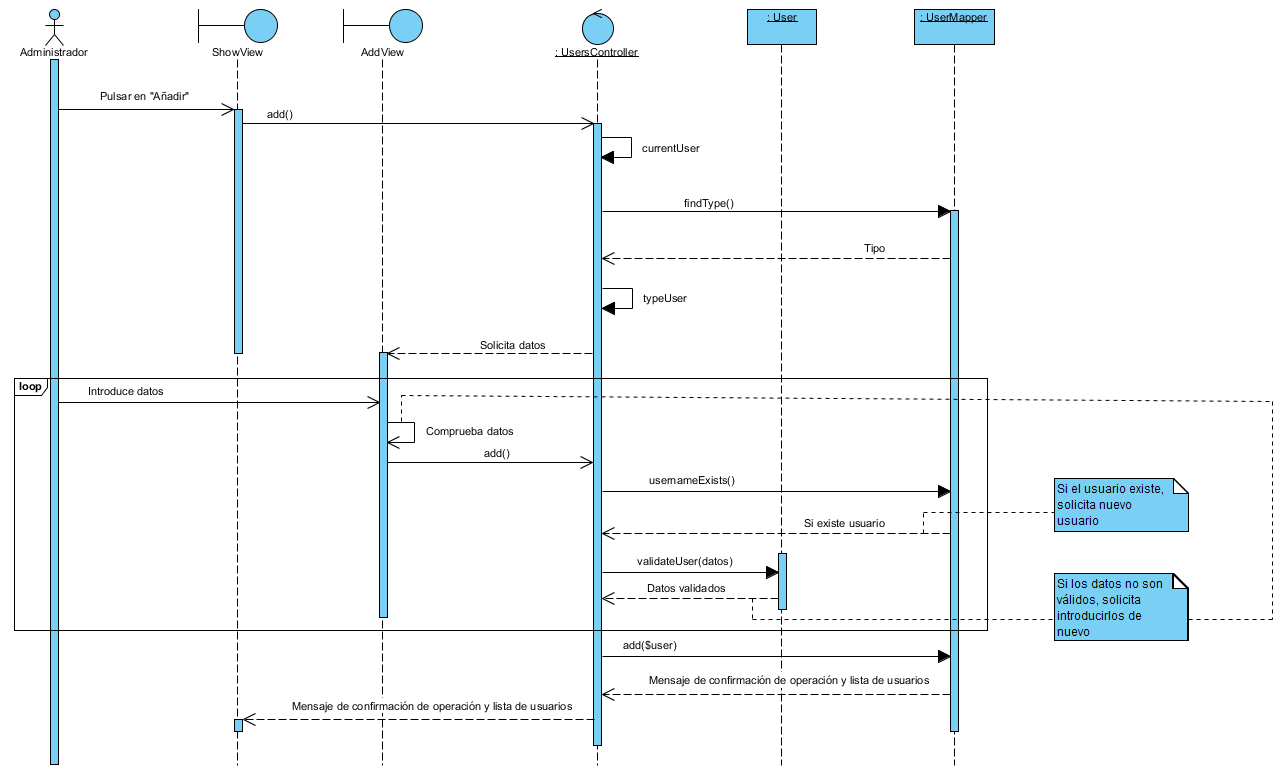
Ver Usuario



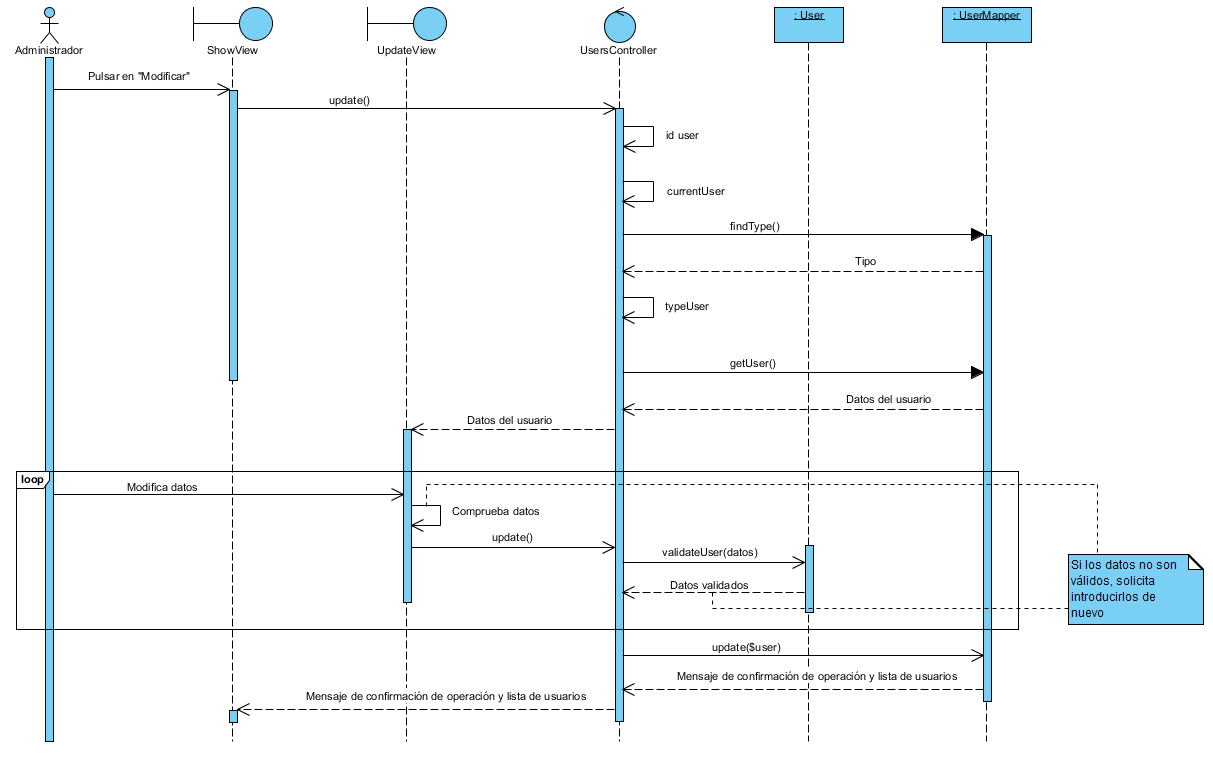
Ver Perfil



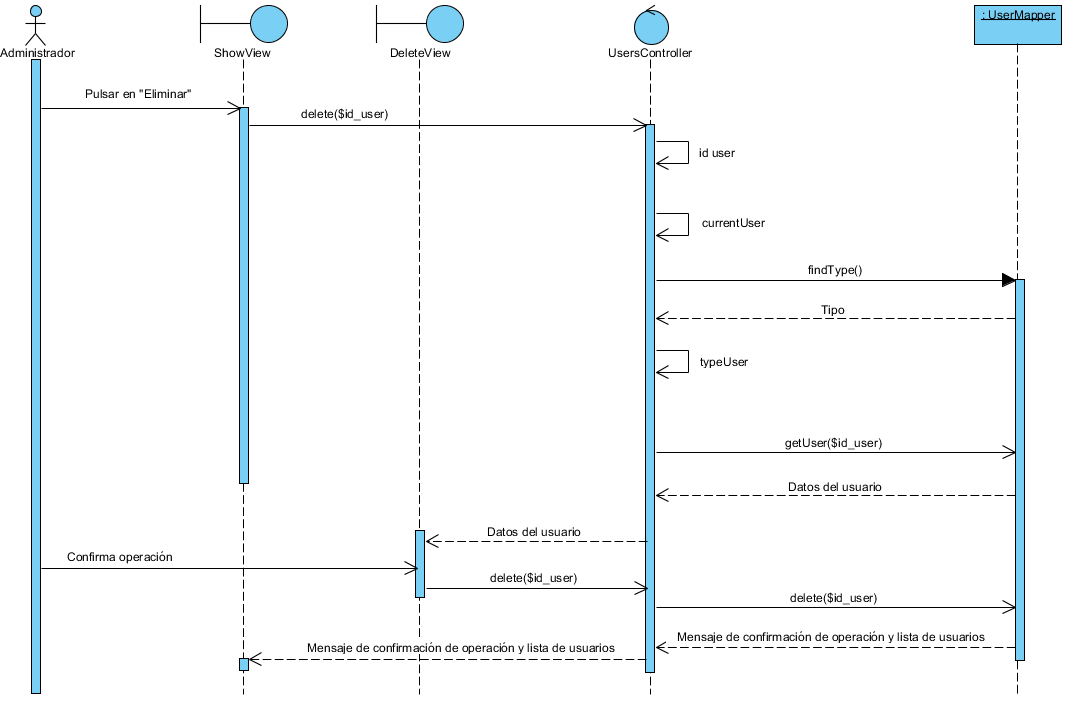
Añadir Usuario



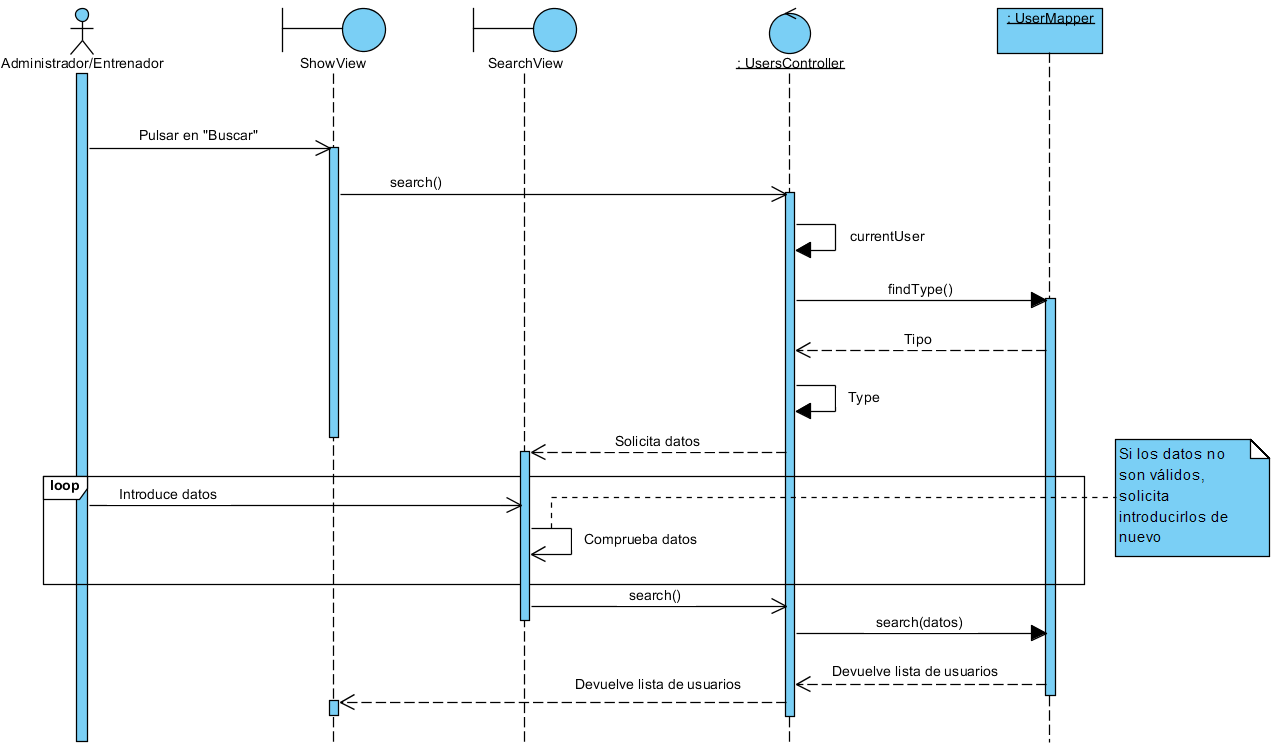
Modificar Usuario



Eliminar Usuario



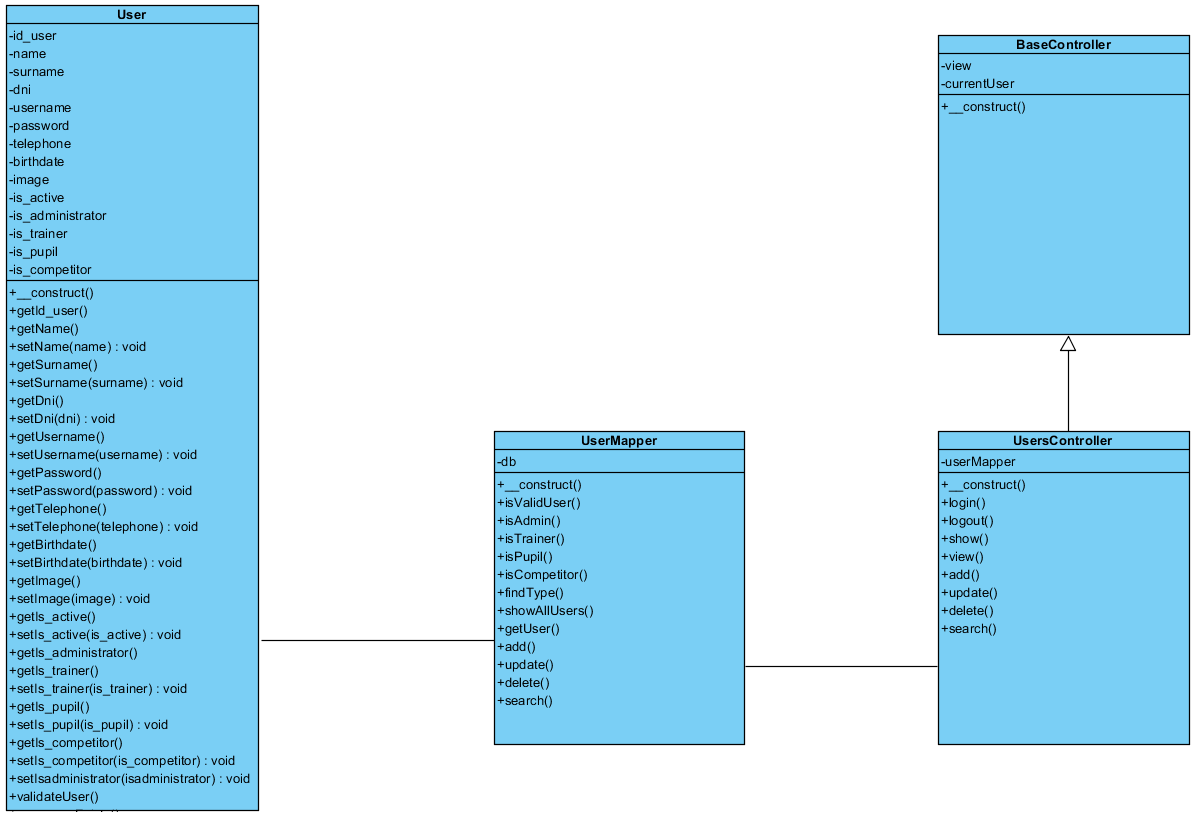
Buscar Usuarios



## Diseño Estático

Se presentan los diagramas de clases parciales de cada funcionalidad, así como una descripción de las responsabilidades de cada clase:

**Gestionar Usuarios**



User: Representa un usuario en la academia.

UserMapper: Interfaz de la base de datos para las entidades de usuario.

UsersController: Invoca peticiones al UserMapper para realizar login, logout y un CRUD (funciones básicas en la capa de persistencia: altas, bajas, modificaciones y consultas) de usuarios.

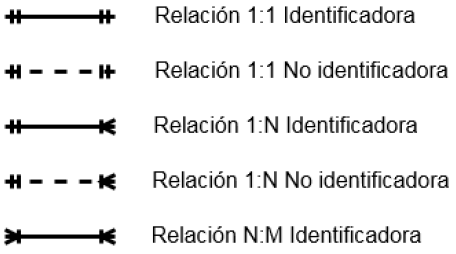
# Gestión de Datos e Información

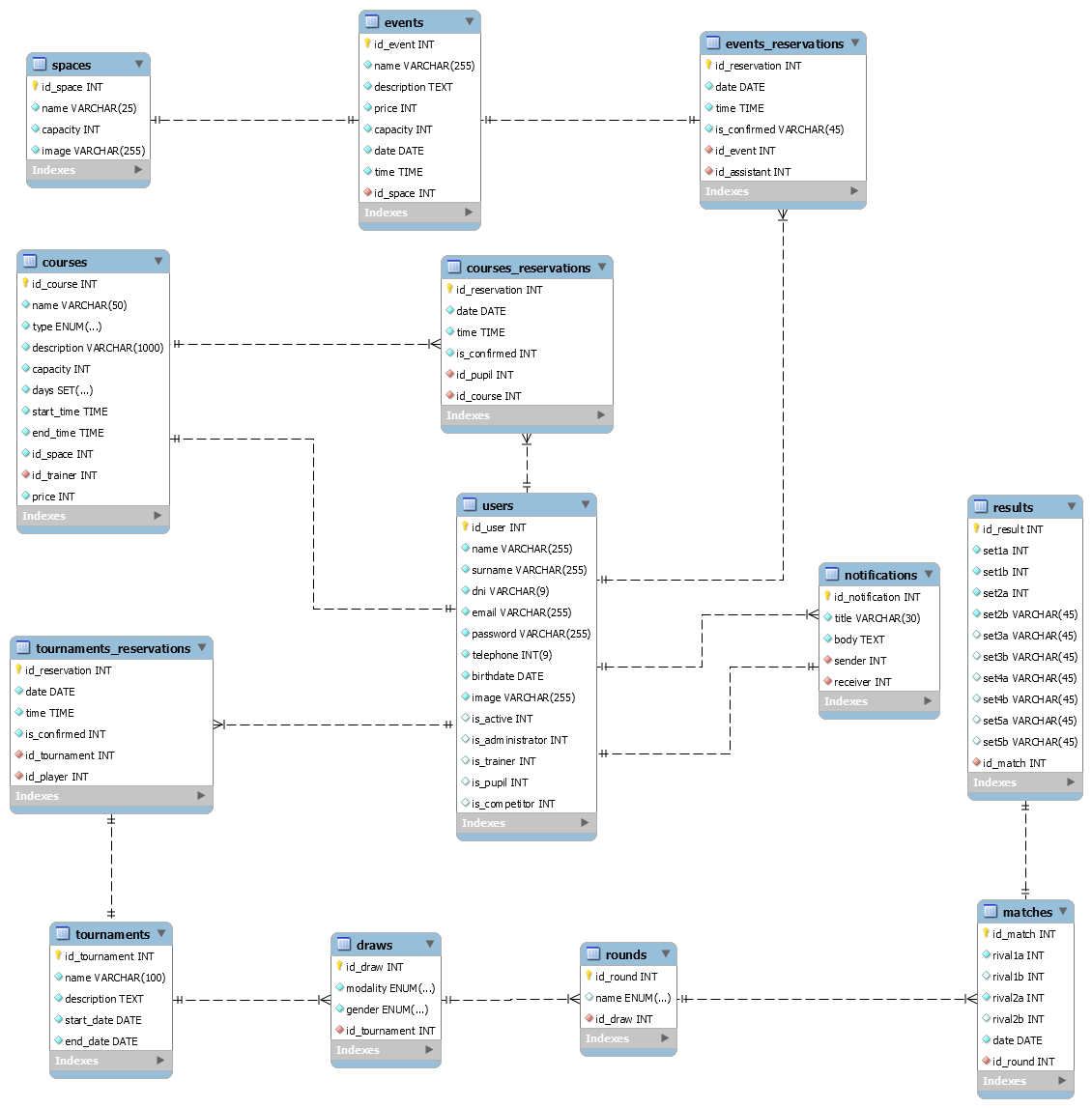
Para la gestión de datos, se ha empleado el sistema gestor de base de datos MySQL. En las siguientes páginas se muestra el diagrama entidad-relación de la base de datos y la estructura física de cada una de las tablas resultantes.

## Diagrama Entidad-Relación

A continuación, se puede ver el diagrama entidad-relación en el que se representan las entidades más relevantes del sistema de información, así como sus interrelaciones y propiedades.

Para el diseño del diagrama se ha empleado la herramienta MySQL Workbench, y debe tenerse en cuenta la siguiente simbología:





## Estructura Física de las Tablas

Abreviaturas:

* TD: Tipo de dato
* CV: Clave primaria
* NN: No nulo
* AI: Auto incremental
* VD: Valor por defecto

Tabla users

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Columna | Descripción | TD | CP | NN | AI | VD |
| id\_user | Identificador de un usuario | INT | X | X | X |  |
| name | Nombre de un usuario | VARCHAR(255) |  | X |  |  |
| surname | Apellidos de un usuario | VARCHAR(255) |  | X |  |  |
| dni | Dni de un usuario | INT(9) |  | X |  |  |
| email | Correo electrónica de un usuario | VARCHAR(255) |  | X |  |  |
| password | Contraseña de un usuario | VARCHAR(255) |  | X |  |  |
| telephone | Teléfono de un usuario | INT(9) |  | X |  |  |
| birthdate | Fecha de nacimiento de un usuario | DATE |  | X |  |  |
| image | Imagen de perfil de un usuario | VARCHAR(255) |  | X |  |  |
| is\_active | Estado de la cuenta de un usuario | INT |  |  |  | 1 |
| is\_administrator | Rol de un usuario | INT |  |  |  | NULL |
| is\_trainer | Rol de un usuario | INT |  |  |  | NULL |
| is\_pupil | Rol de un usuario | INT |  |  |  | NULL |
| is\_competitor | Rol de un usuario | INT |  |  |  | NULL |

Tabla courses

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Columna | Descripción | TD | CP | NN | AI | VD |
| id\_course | Identificador de un curso | INT | X | X | X |  |
| name | Nombre de un curso | VARCHAR(50) |  | X |  |  |
| type | Tipo de un curso | ENUM('Children', 'Adults') |  | X |  |  |
| description | Descripción de un curso | VARCHAR(1000) |  | X |  |  |
| capacity | Capacidad de un curso | INT |  | X |  |  |
| days | Días en los que se imparte un curso | SET('Sunday', 'Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday') |  | X |  |  |
| start\_time | Hora de inicio un curso | TIME |  | X |  |  |
| end\_time | Hora de finalización un curso | TIME |  | X |  |  |
| id\_space | Identificador del espacio en el que se imparte un curso | INT |  |  |  |  |
| id\_trainer | Identificador del entrenador que dirige un curso | INT |  |  |  |  |
| price | Precio de un curso | INT |  |  |  |  |

Tabla courses\_reservations

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Columna | Descripción | TD | CP | NN | AI | VD |
| id\_reservation | Identificador de la reserva de un curso | INT | X | X | X |  |
| date | Fecha de la reserva de un curso | DATE |  | X |  |  |
| time | Hora de la reserva de un curso | TIME |  | X |  |  |
| is\_confirmed | Estado de la reserva de un curso | INT |  | X |  |  |
| id\_pupil | Identificador del alumno que hace la reserva de un curso | INT |  | X |  |  |
| id\_course | Identificador del curso de una reserva | INT |  | X |  |  |

Tabla spaces

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Columna | Descripción | TD | CP | NN | AI | VD |
| id\_space | Identificador de un espacio | INT | X | X | X |  |
| name | Nombre de un espacio | VARCHAR(25) |  | X |  |  |
| capacity | Capacidad de un espacio | INT |  | X |  |  |
| image | Imagen de un espacio | VARCHAR(255) |  | X |  |  |

Tabla events

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Columna | Descripción | TD | CP | NN | AI | VD |
| id\_event | Identificador de un evento | INT | X | X | X |  |
| name | Nombre de un evento | VARCHAR(255) |  | X |  |  |
| description | Descripción de un evento | TEXT |  | X |  |  |
| price | Precio de un evento | INT |  | X |  |  |
| capacity | Capacidad de un evento | INT |  | X |  |  |
| date | Fecha de un evento | DATE |  | X |  |  |
| time | Hora de un evento | TIME |  | X |  |  |
| id\_space | Identificador del espacio en el que se lleva a cabo un evento | INT |  | X |  |  |

Tabla events\_reservations

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Columna | Descripción | TD | CP | NN | AI | VD |
| id\_reservation | Identificador de la reserva de un evento | INT | X | X | X |  |
| date | Fecha de la reserva de un evento | DATE | X |  |  |  |
| time | Hora de la reserva de un evento | TIME | X |  |  |  |
| is\_confirmed | Estado de la reserva de un evento | VARCHAR(45) | X |  |  |  |
| id\_event | Identificador del evento de una reserva | INT |  |  |  |  |
| id\_assistant | Identificador del usuario que hace la reserva de un curso | INT |  |  |  |  |

Tabla tournaments

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Columna | Descripción | TD | CP | NN | AI | VD |
| id\_tournament | Identificador de un torneo | INT | X | X | X |  |
| name | Nombre de un torneo | VARCHAR(100) |  | X |  |  |
| description | Descripción de un torneo | TEXT |  | X |  |  |
| start\_date | Fecha de inicio de un torneo | DATE |  | X |  |  |
| end\_date | Fecha en la que finaliza un torneo | DATE |  | X |  |  |

Tabla tournaments\_reservations

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Columna | Descripción | TD | CP | NN | AI | VD |
| id\_reservation | Identificador de la reserva de un torneo | INT | X | X | X |  |
| date | Fecha de la reserva de un torneo | DATE |  | X |  |  |
| time | Hora de la reserva de un torneo | TIME |  | X |  |  |
| is\_confirmed | Estado de la reserva de un torneo | INT |  | X |  |  |
| id\_tournament | Identificador del torneo de una reserva | INT |  | X |  |  |
| id\_player | Identificador del competidor que hace la reserva de un curso | INT |  | X |  |  |

Tabla draws

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Columna | Descripción | TD | CP | NN | AI | VD |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

Tabla rounds

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Columna | Descripción | TD | CP | NN | AI | VD |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

Tabla matches

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Columna | Descripción | TD | CP | NN | AI | VD |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

Tabla results

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Columna | Descripción | TD | CP | NN | AI | VD |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

Tabla notifications

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Columna | Descripción | TD | CP | NN | AI | VD |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

# Pruebas Llevadas a Cabo

Se detallan las pruebas de caja negra y caja blanca realizadas para cada uno de los requisitos funcionales definidos en los apartados anteriores.

Para cada tipo de prueba se presentan dos tablas, una con su definición y otra con su comprobación y valoración de los resultados obtenidos.

## Gestionar Usuario

**Listar Usuarios**

Caja Negra: No se han realizado pruebas de caja negra ya que no se realiza ninguna inserción al sistema que deba ser comprobada.

Caja Blanca: Comprobamos la correcta ejecución de la funcionalidad.

Definición

|  |  |  |
| --- | --- | --- |
| Campo | Válido | Inválido |
| Todos | 1-Visualización de los usuarios del sistema en la vista correspondiente. | 2-No se visualizan los usuarios del sistema en la vista correspondiente. |

Comprobación

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Campo | Valor | Prueba | Resultado Esperado | Resultado Obtenido |
| Todos | Visualización de los datos | 1 | Visualización de los datos | Visualización de los datos |

**Ver Usuario**

Caja Negra: No se han realizado pruebas de caja negra ya que no se realiza ninguna inserción al sistema que deba ser comprobada.

Caja Blanca: Comprobamos la correcta ejecución de la funcionalidad.

Definición

|  |  |  |
| --- | --- | --- |
| Campo | Válido | Inválido |
| Todos | 1-Visualización de los datos del usuario solicitado en la vista correspondiente. | 2-No se visualizan los datos del usuario solicitado en la vista correspondiente. |

Comprobación

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Campo | Valor | Prueba | Resultado Esperado | Resultado Obtenido |
| Todos | Visualización de los datos | 1 | Visualización de los datos | Visualización de los datos |

**Añadir Usuario**

Caja Negra: Se comprueban las salidas generadas en respuesta a las entradas seleccionadas.

Definición

|  |  |  |
| --- | --- | --- |
| Campo | Válido | Inválido |
| name | 1-Varchar alfabético | 2-Vacío  3-Combinación de caracteres no alfabéticos |
| surname | 4-Varchar alfabético | 5-Vacío  6-Combinación de caracteres no alfabéticos  7-Una sola palabra |
| dni | 8-Varchar con combinación de caracteres: 34273074S | 9-Vacío  10-Combinación de caracteres no especificada como válida |
| username | 11-Varchar con combinación de caracteres: email@email.com | 12-Vacío  13- Combinación de caracteres no especificada como válida |
| password | 14-Varcghar de 5 o más caracteres | 15-Vacío  16-Varchar de menos de 5 caracteres |
| telephone | 17-Int con formato: [967]87654321 | 18-Vacío  19-Combinación de caracteres con formato no válido |
| birthdate | 20-Date con formato:1990-06-11 | 21-Vacío  22-Combinación de caracteres con formato no válido |
| image | 23-Varchar con extensión gif, jpeg, jpg, png  24- Tamaño inferior a 5MB | 25-Vacío  26-Varchar con extensión no aceptada como válida  27-Tamaño superior a 5MB |
| is\_active | 28-Int 1  29-Int 0 | 30-Otro int |
| is\_administrator | 31-Int 1  32-Int 0 | 33-Otro int |
| is\_trainer | 34-Int 1  35-Int 0 | 36-Otro int |
| is\_pupil | 37-Int 1  38-Int 0 | 39-Otro int |
| is\_competitor | 40-Int 1  41-Int 0 | 42-Otro int |

Comprobación

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Campo | Valor | Prueba | Resultado Esperado | Resultado Obtenido |
| name | Brais | 1 | Válido | Válido |
| Vacío | 2 | Error | Error |
| Brais1 | 3 | Error | Error |
| surname | Domínguez Álvarez | 4 | Válido | Válido |
| Vacío | 5 | Error | Error |
| Domínguez Álva123 | 6 | Error | Error |
| Domínguez | 7 | Error | Error |
| dni | 34273074S | 8 | Válido | Válido |
| Vacío | 9 | Error | Error |
| abcd1234Q | 10 | Error | Error |
| username | braisda@gmail.com | 11 | Error | Válido |
| Vacío | 12 | Válido | Error |
| brais.com | 13 | Error | Error |
| password | admin | 14 | Error | Válido |
|  | Vacío | 15 | Válido | Error |
| 1234 | 16 | Error | Error |
| telephone | 666555444 | 17 | Error | Válido |
| Vacío | 18 | Válido | Error |
| 666abcqwe | 19 | Error | Error |
| birthdate | 1991-06-10 | 20 | Error | Válido |
| Vacío | 21 | Válido | Error |
| 1991-aa-bb | 22 | Error | Error |
| image | profile.png | 23 | Válido | Válido |
| Tamaño inferior a 5MB | 24 | Válido | Válido |
| Vacío | 25 | Error | Error |
| profile.txt | 26 | Error | Error |
| Tamaño superior a 5MB | 27 | Error | Error |
| is\_active | 1 | 28 | Válido | Válido |
| 0 | 29 | Válido | Válido |
| 2 | 30 | Error | Error |
| is\_administrator | 1 | 31 | Válido | Válido |
| 0 | 32 | Válido | Válido |
| 2 | 33 | Error | Error |
| is\_trainer | 1 | 34 | Válido | Válido |
| 0 | 35 | Válido | Válido |
| 2 | 36 | Error | Error |
| is\_pupil | 1 | 37 | Válido | Válido |
| 0 | 38 | Válido | Válido |
| 2 | 39 | Error | Error |
| is\_competitor | 1 | 40 | Válido | Válido |
| 0 | 41 | Válido | Válido |
| 2 | 42 | Error | Error |

Caja Blanca: Comprobamos la correcta ejecución de la funcionalidad.

Definición

|  |  |  |
| --- | --- | --- |
| Campo | Válido | Inválido |
| Todos | 1-Tras introducir los datos, éstos deben estar almacenados en la base de datos. | 2-Tras introducir los datos, éstos no están almacenados en la base de datos. |
| dni | 3- DNI con código de control correcto. | 4-DNI con código de control incorrecto. |
| username | 5-El nombre de usuario debe ser único. | 6-No pueden existir dos usuarios con el mismo nombre de usuario. |

Comprobaciones

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Campo | Valor | Prueba | Resultado Esperado | Resultado Obtenido |
| Todos | Inserción de datos válida | 1 | Datos almacenados | Datos almacenados |
| dni | 34273074S | 3 | Válido | Válido |
| 12345678A | 4 | Error | Error |
| username | braisda@gmail.com | 5 | Válido | Válido |
| braisda@gmail.com | 6 | Error | Error |

**Modificar Usuario**

Caja Negra: Se comprueban las salidas generadas en respuesta a las entradas seleccionadas.

Definición

|  |  |  |
| --- | --- | --- |
| Campo | Válido | Inválido |
| name | 1-Varchar alfabético | 2-Vacío  3-Combinación de caracteres no alfabéticos |
| surname | 4-Varchar alfabético | 5-Vacío  6-Combinación de caracteres no alfabéticos  7-Una sola palabra |
| dni | 8-Varchar con combinación de caracteres: 34273074S | 9-Vacío  10-Combinación de caracteres no especificada como válida |
| username | 11-Varchar con combinación de caracteres: email@email.com | 12-Vacío  13- Combinación de caracteres no especificada como válida |
| password | 14-Varcghar de 5 o más caracteres | 15-Vacío  16-Varchar de menos de 5 caracteres |
| telephone | 17-Int con formato: [967]87654321 | 18-Vacío  19-Combinación de caracteres con formato no válido |
| birthdate | 20-Date con formato:1990-06-11 | 21-Vacío  22-Combinación de caracteres con formato no válido |
| image | 23-Varchar con extensión gif, jpeg, jpg, png  24- Tamaño inferior a 5MB | 25-Vacío  26-Varchar con extensión no aceptada como válida  27-Tamaño superior a 5MB |
| is\_active | 28-Int 1  29-Int 0 | 30-Otro int |
| is\_administrator | 31-Int 1  32-Int 0 | 33-Otro int |
| is\_trainer | 34-Int 1  35-Int 0 | 36-Otro int |
| is\_pupil | 37-Int 1  38-Int 0 | 39-Otro int |
| is\_competitor | 40-Int 1  41-Int 0 | 42-Otro int |

Comprobación

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Campo | Valor | Prueba | Resultado Esperado | Resultado Obtenido |
| name | Brais | 1 | Válido | Válido |
| Vacío | 2 | Error | Error |
| Brais1 | 3 | Error | Error |
| surname | Domínguez Álvarez | 4 | Válido | Válido |
| Vacío | 5 | Error | Error |
| Domínguez Álva123 | 6 | Error | Error |
| Domínguez | 7 | Error | Error |
| dni | 34273074S | 8 | Válido | Válido |
| Vacío | 9 | Error | Error |
| abcd1234Q | 10 | Error | Error |
| username | braisda@gmail.com | 11 | Error | Válido |
| Vacío | 12 | Válido | Error |
| brais.com | 13 | Error | Error |
| password | admin | 14 | Error | Válido |
|  | Vacío | 15 | Válido | Error |
| 1234 | 16 | Error | Error |
| telephone | 666555444 | 17 | Error | Válido |
| Vacío | 18 | Válido | Error |
| 666abcqwe | 19 | Error | Error |
| birthdate | 1991-06-10 | 20 | Error | Válido |
| Vacío | 21 | Válido | Error |
| 1991-aa-bb | 22 | Error | Error |
| image | profile.png | 23 | Válido | Válido |
| Tamaño inferior a 5MB | 24 | Válido | Válido |
| Vacío | 25 | Error | Error |
| profile.txt | 26 | Error | Error |
| Tamaño superior a 5MB | 27 | Error | Error |
| is\_active | 1 | 28 | Válido | Válido |
| 0 | 29 | Válido | Válido |
| 2 | 30 | Error | Error |
| is\_administrator | 1 | 31 | Válido | Válido |
| 0 | 32 | Válido | Válido |
| 2 | 33 | Error | Error |
| is\_trainer | 1 | 34 | Válido | Válido |
| 0 | 35 | Válido | Válido |
| 2 | 36 | Error | Error |
| is\_pupil | 1 | 37 | Válido | Válido |
| 0 | 38 | Válido | Válido |
| 2 | 39 | Error | Error |
| is\_competitor | 1 | 40 | Válido | Válido |
| 0 | 41 | Válido | Válido |
| 2 | 42 | Error | Error |

Caja Blanca: Comprobamos la correcta ejecución de la funcionalidad.

Definición

|  |  |  |
| --- | --- | --- |
| Campo | Válido | Inválido |
| Todos | 1-Tras introducir los datos, éstos deben estar almacenados en la base de datos. | 2-Tras introducir los datos, éstos no están almacenados en la base de datos. |
| dni | 3- DNI con código de control correcto. | 4-DNI con código de control incorrecto. |
| username | 5-El nombre de usuario debe ser único. | 6-No pueden existir dos usuarios con el mismo nombre de usuario. |
| username | 7-Debe redirigirse a la página de login si se modifica el email del usuario en sesión | 8-No se redirige a la página de login si se modifica el email del usuario en sesión |

Comprobaciones

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Campo | Valor | Prueba | Resultado Esperado | Resultado Obtenido |
| Todos | Inserción de datos válida | 1 | Datos almacenados | Datos almacenados |
| dni | 34273074S | 3 | Válido | Válido |
| 12345678A | 4 | Error | Error |
| username | braisda@gmail.com | 5 | Válido | Válido |
| braisda@gmail.com | 6 | Error | Error |
| username | brais@gmail.com | 7 | Redirección a login | Redirección a login |

Eliminar Usuario

Caja Negra: No se han realizado pruebas de caja negra ya que no se realiza ninguna inserción al sistema que deba ser comprobada.

Caja Blanca: Comprobamos la correcta ejecución de la funcionalidad.

Definición

|  |  |  |
| --- | --- | --- |
| Campo | Válido | Inválido |
| Todos | 1-Se elimina el usuario seleccionado del sistema. | 2-No se elimina el usuario seleccionado del sistema. |

Comprobación

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Campo | Valor | Prueba | Resultado Esperado | Resultado Obtenido |
| Todos | Se elimina un usuario | 1 | Eliminación del usuario | Eliminación del usuario |

Buscar Usuario

# Manual de Usuario

# Principales Aportaciones

Una vez finalizado el proyecto, con el propósito de satisfacer los objetivos descritos en el punto 2, la aplicación desarrollada, ha permitido realizar las siguientes aportaciones:

Se ha conseguido informatizar la gestión de una academia de tenis, permitiendo:

* Gestionar el alta y la baja de clientes en la academia, así como disponer de información detallada de cada uno de ellos, pudiendo modificar dicha información si fuera necesario (Objetivo 1).
* Gestionar las instalaciones de la academia, pudiendo añadir y eliminar espacios o modificar los ya existentes, así como asignar dichos espacios a una actividad en concreto (Objetivo 2).
* Gestionar eficazmente los cursos, eventos y torneos de la academia, junto con sus cuadros y partidos. Los responsables de la academia podrán ahora crear, eliminar y modificar estas actividades, conociendo en todo momento los usuarios que las realizan y qué espacios ocupan (Objetivos 3, 4 y 5).
* Crear un sistema de reservas que permita a los clientes de la academia anotarse a las actividades que deseen, tanto cursos como eventos o torneos, pudiendo también cancelar las reservas cuando lo deseen. Del mismo modo, la aplicación permite a los responsables de la academia gestionar estas reservas pudiendo aceptar o denegar aquellas que estimen oportunas, teniendo un registro detallado de todas las reservas de la academia (Objetivos 6, 7, 8, 11, 12 y 13).
* Obtener en todo momento, la información acerca de los servicios ofrecidos por la academia y sus características, a sus usuarios (Objetivo 10).
* Poner en contacto a todos los usuarios de la aplicación por medio de un sistema de notificaciones, que permitirá una comunicación inmediata de los clientes con los responsables de la academia (Objetivo 9 y 14).

Otra aportación relevante, es que todo lo anteriormente descrito se realizará por medio de una aplicación informática que provee un entorno seguro, cuidando la privacidad de sus usuarios y gestionando los errores que se puedan producir con su interacción (Objetivos 15 y 16).

Se ha conseguido crear una herramienta de uso sencillo, que permite, de manera segura, reducir costes y tiempo a la academia a la hora de gestionar sus recursos y clientes, mejorar los servicios que les ofrecen y facilitar también a estos últimos la práctica del tenis, cumpliendo así con el objetivo general que se pretendía satisfacer con este proyecto, favorecer la profesionalización del deporte.

# Conclusiones

# Vías de Trabajo Futuro

# Referencias

# Anexos