



TÉCNICO SUPERIOR EN DESARROLLO DE APLICACIONES WEB

Departamento de Informática



MANUAL TÉCNICO

Índice

1. Introducción.....	3
2. Arquitectura de la aplicación.....	4
2.1. Frontend.....	5
2.1.1. Tecnologías usadas.....	5
2.1.2. Entorno de desarrollo.....	5
2.2. Backend.....	5
2.2.1. Tecnologías usadas.....	5
2.2.2. Entorno de desarrollo.....	6
3. Documentación técnica.....	6
3.1. Análisis.....	6
3.2. Desarrollo (diagrama de secuencias,...).....	15
3.3. Pruebas realizadas.....	17
4. Proceso de despliegue.....	18
4.1. Requisitos de sistema.....	18
4.2. Carga del script SQL de EntreHobbies en el servidor.....	19
4.3. Subida de la aplicación al servidor del instituto.....	22
5. Propuestas de mejoras.....	25
5.1. Mejoras aplicadas durante el desarrollo.....	25
5.2. Mejoras propuestas para futuras versiones.....	26
6. Bibliografía.....	27

1. Introducción

EntreHobbies es una aplicación web desarrollada como proyecto final del **FP Superior de Desarrollo de Aplicaciones Web**, cuyo propósito es facilitar la organización y participación en eventos relacionados con aficiones compartidas.

La plataforma está diseñada principalmente para dos perfiles de usuario:

- **Colaborador** (usuario registrado): puede registrarse, iniciar sesión, explorar eventos disponibles, inscribirse en ellos, así como crear y gestionar sus propios eventos. Dispone de un panel personalizado donde puede visualizar sus participaciones, modificar eventos creados y editar su perfil.
- **Administrador**: cuenta con privilegios avanzados que le permiten acceder a toda la información de la aplicación. Puede visualizar estadísticas generales, visualizar los perfiles de los usuarios, revisar todos los eventos creados, consultar las categorías y subcategorías.

Además, **la aplicación contempla a los usuarios anónimos (no registrados)**, quienes pueden navegar por el sitio de forma limitada, consultar los eventos públicos disponibles y acceder a secciones informativas. Sin embargo, para participar en eventos o acceder a funcionalidades avanzadas, deberán registrarse e iniciar sesión.

EntreHobbies ha sido construida bajo una arquitectura multicapa siguiendo el patrón **Modelo-Vista-Controlador (MVC)**, utilizando **JSP y Servlets** para el backend, junto con tecnologías estándar como **HTML, CSS y JavaScript** en el frontend. El proyecto emplea **MySQL** como sistema gestor de bases de datos y se apoya en herramientas como **Apache Tomcat** y **Maven** para su despliegue y gestión de dependencias.

2. Arquitectura de la aplicación

La arquitectura de **EntreHobbies** sigue el patrón de diseño **Modelo–Vista–Controlador (MVC)**, esto nos permite separar la lógica de presentación, negocio y acceso a datos. Esto permite una mayor mantenibilidad, escalabilidad y reutilización del código.

Capas de la aplicación

Modelo(Beans): Se utilizan clases Java para el encapsulamiento de la lógica de negocio y mapear las entidades principales del sistema. Los **beans** principales de esta aplicación son **Usuario, Evento, Categoría, Subcategoría y Provincia**.

Vista: Compuesta por páginas **JSP** que permiten la interacción del usuario con el sistema. Se incluyen vistas específicas según el rol: anónimos, colaboradores y administradores.

Controlador: Implementado mediante **Servlets** que gestionan las peticiones del usuario, coordinan el flujo entre las vistas y los modelos, y devuelven la respuesta adecuada. (Ej: Login, UsuarioController, EventoController)

Tecnologías principales usadas

- Lenguaje de programación: Java (JDK 11)
- Servidor de aplicaciones: Apache Tomcat 9
- Sistema de gestión de bases de datos: MySQL 8
- Gestor de dependencias y construcción: Apache Maven
- Entorno de desarrollo: IntelliJ IDEA 2023.3.4

2.1. Frontend

2.1.1. Tecnologías usadas

- **HTML5** → para definir la estructura base de las páginas y formularios de la aplicación.
- **CSS3** → para aplicar estilos visuales y personalizar el diseño de los distintos componentes.
- **JavaScript(vanilla)** → scripts propios para gestión de filtros, búsqueda dinámica de eventos, y validación de formularios.
- **Bootstrap 5** → para un diseño responsive y componentes reutilizables como menús, botones, modales y alertas.
- **Bootstrap Table** → extensión de Bootstrap que facilita la creación de tablas interactivas con ordenación, búsqueda y paginación.
- **Chart.js** → librería utilizada para la generación de gráficas estadísticas en el panel del administrador.

2.1.2. Entorno de desarrollo

- **IntelliJ IDEA 2023.3.4**, ya que permite trabajar con archivos JSP, CSS y JavaScript de forma cómoda y eficiente.

2.2. Backend

2.2.1. Tecnologías usadas

- **Java (JDK 11)** → lenguaje de programación principal para el desarrollo de la lógica del servidor.
- **Servlets** → utilizados como controladores para gestionar las peticiones del usuario, acceder a la base de datos y redirigir a las vistas correspondientes.
- **Java Server Pages (JSP)** → empleados para generar dinámicamente las vistas, mostrar información al usuario y recuperar datos desde los controladores.

- **Apache Tomcat 9** → servidor de aplicaciones que permite ejecutar la aplicación web localmente o en producción.
- **Apache Maven** → herramienta para la gestión de dependencias, empaquetado del proyecto y automatización de tareas de compilación.

2.2.2. Entorno de desarrollo

- **IntelliJ IDEA 2023.3.4** → Utilizado para la edición y organización del código Java, configuración del servidor Tomcat, depuración y pruebas locales. Ofrece soporte completo para Maven y facilita la estructuración del proyecto en capas.
- **Apache Tomcat 9** → Integrado con IntelliJ para ejecutar y desplegar la aplicación localmente.
- **MySQL Workbench** → Herramientas auxiliares utilizadas para gestionar y visualizar la base de datos durante el desarrollo y la fase de pruebas.

3. Documentación técnica

3.1. Análisis

Diagrama de Gantt

Diagrama de Gantt utilizado para planificar y distribuir las tareas del proyecto, desde el análisis inicial hasta la fase de pruebas y entrega final.

Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: EntreHobbies



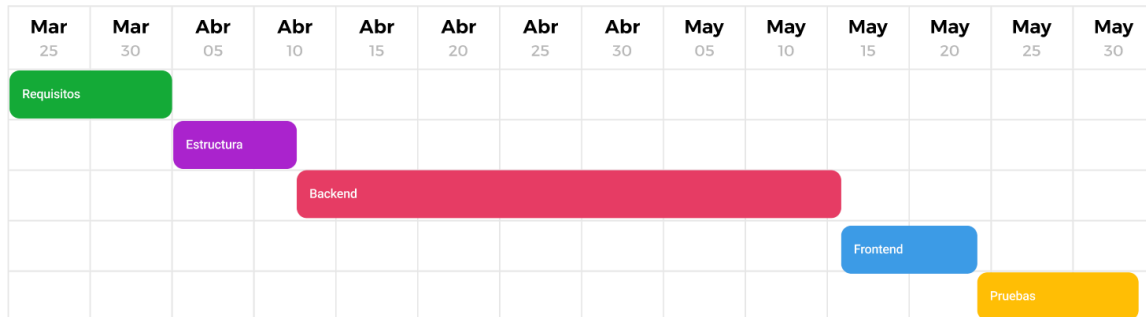
JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

30 MAYO 2025

Diagrama de Gantt

EntreHobbies



- VERDE: ANÁLISIS DE REQUISITOS
- PÚRPURA: DISEÑO Y ESTRUCTURA DEL PROYECTO
- ROJO: DESARROLLO BACKEND Y BBDD
- AZUL: DESARROLLO FRONTEND
- AMARILLO: PRUEBAS Y DOCUMENTACIÓN

Antes de iniciar el desarrollo de la aplicación **EntreHobbies**, se llevó a cabo una fase de análisis en la que se identificaron las funcionalidades necesarias para cumplir con los objetivos del sistema, así como los requisitos de calidad y rendimiento que debe satisfacer.

Los **requisitos funcionales** definen las acciones específicas que podrá realizar cada tipo de usuario (anónimo, colaborador o administrador), mientras que los **requisitos no funcionales** establecen criterios relacionados con la usabilidad, el rendimiento, la seguridad y la compatibilidad de la aplicación.

A continuación, se detallan ambos tipos de requisitos que han guiado el diseño y desarrollo del sistema.

Requisitos funcionales

Los requisitos funcionales se han clasificado según el tipo de usuario (anónimo, colaborador y administrador), ya que cada rol cuenta con un conjunto de funcionalidades específicas y diferenciadas.

RF - Usuario anónimo

- **RF1** - Podrá navegar por la aplicación de forma limitada, sin necesidad de registrarse.
- **RF2** - Podrá visualizar los eventos públicos disponibles.
- **RF3** - Podrá filtrar los eventos por categoría, subcategoría o provincia.
- **RF4** - Podrá acceder a páginas informativas como "Sobre nosotros" o "Política de privacidad".
- **RF5** - Podrá registrarse mediante un formulario con validación de datos.
- **RF6** - Si intenta participar en un evento o acceder a funciones restringidas, el sistema le mostrará un mensaje informativo indicando que debe estar registrado e iniciar sesión para realizar la acción.

RF - Usuario Colaborador

- **RF7** - Podrá iniciar sesión introduciendo sus credenciales (email o username y contraseña).
- **RF8** - Podrá visualizar los eventos públicos disponibles.
- **RF9** - Podrá inscribirse en eventos si hay plazas disponibles y aún no ha participado.
- **RF10** - Podrá cancelar su participación en un evento si este aún no ha comenzado.
- **RF11** - Podrá crear nuevos eventos, completando un formulario con todos los datos requeridos.
- **RF12** - Podrá gestionar sus propios eventos: modificarlos, eliminarlos o cancelarlos.
- **RF13** - Podrá actualizar la información de su perfil de usuario.
- **RF14** - Podrá cerrar sesión desde cualquier punto de la aplicación.

RF - Usuario administrador

- **RF15** - Podrá iniciar sesión con credenciales de administrador.
- **RF16** - Tendrá acceso a un panel de administración exclusivo (dashboard).
- **RF17** - Podrá visualizar todos los eventos registrados en el sistema.

- **RF18** - Podrá visualizar todos los usuarios junto con sus datos.
- **RF19** - Podrá visualizar las categorías y subcategorías disponibles.

Requisitos no funcionales

RNF1 - Interfaz intuitiva y accesible: La aplicación contará con un diseño claro, atractivo y fácil de usar, adaptado a distintos tamaños de pantalla mediante un diseño responsive. Se prioriza la experiencia de usuario.

RNF2 - Seguridad y privacidad: Se garantizará la protección de los datos personales mediante validaciones en formularios, sesiones seguras y control de acceso por roles. Solo los usuarios autorizados podrán acceder a funcionalidades restringidas.

RNF3 - Escalabilidad: La arquitectura de la aplicación está diseñada para permitir la incorporación de nuevas funcionalidades o el aumento de usuarios y eventos sin comprometer el rendimiento general.

RNF4 - Rendimiento y tiempo de respuesta: Las operaciones principales (como búsquedas o visualización de eventos) deben ejecutarse en un tiempo medio inferior a 3 segundos bajo condiciones normales de uso.

RNF5 - Compatibilidad multiplataforma: La aplicación será accesible desde los principales navegadores web (Google Chrome, Firefox) y se visualizará correctamente en distintos dispositivos (ordenadores, tablets y móviles).

También, durante la fase de análisis de **EntreHobbies**, se han identificado los actores principales del sistema, las funcionalidades disponibles según el tipo de usuario, así como la estructura general de las entidades y su relación con la base de datos.

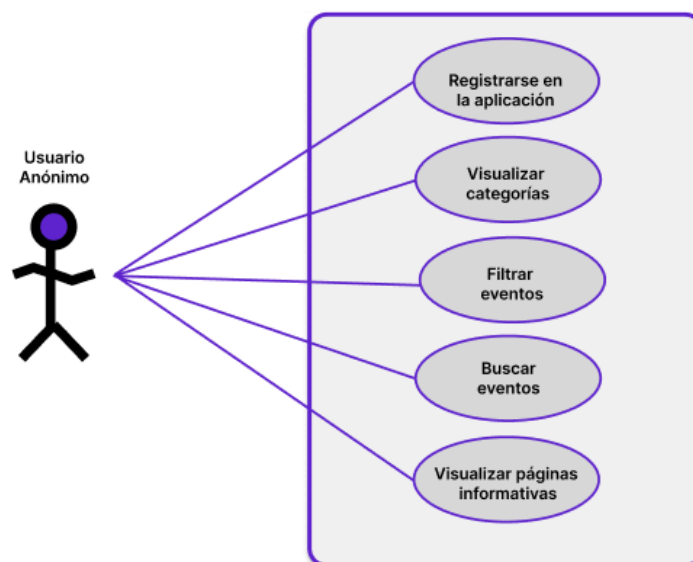
A continuación, se presentan los principales diagramas desarrollados para describir la lógica del sistema:

Diagrama de casos de uso

El diagrama de casos de uso refleja las acciones que pueden realizar los diferentes tipos de usuarios.

Este diagrama permite tener una visión general de las funcionalidades ofrecidas por la aplicación y el alcance de cada actor.

Usuario anónimo



Proyecto “Desarrollo de Aplicaciones Web”

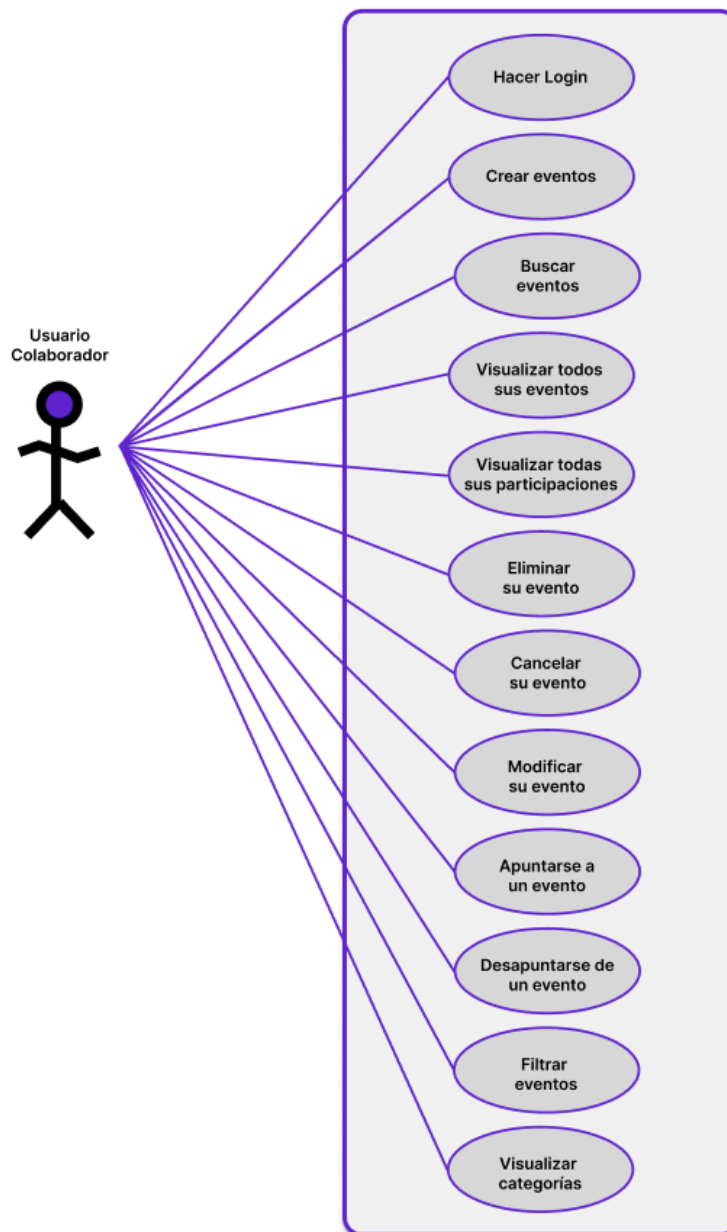
Título del Proyecto: EntreHobbies



JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

Usuario Colaborador



Proyecto “Desarrollo de Aplicaciones Web”

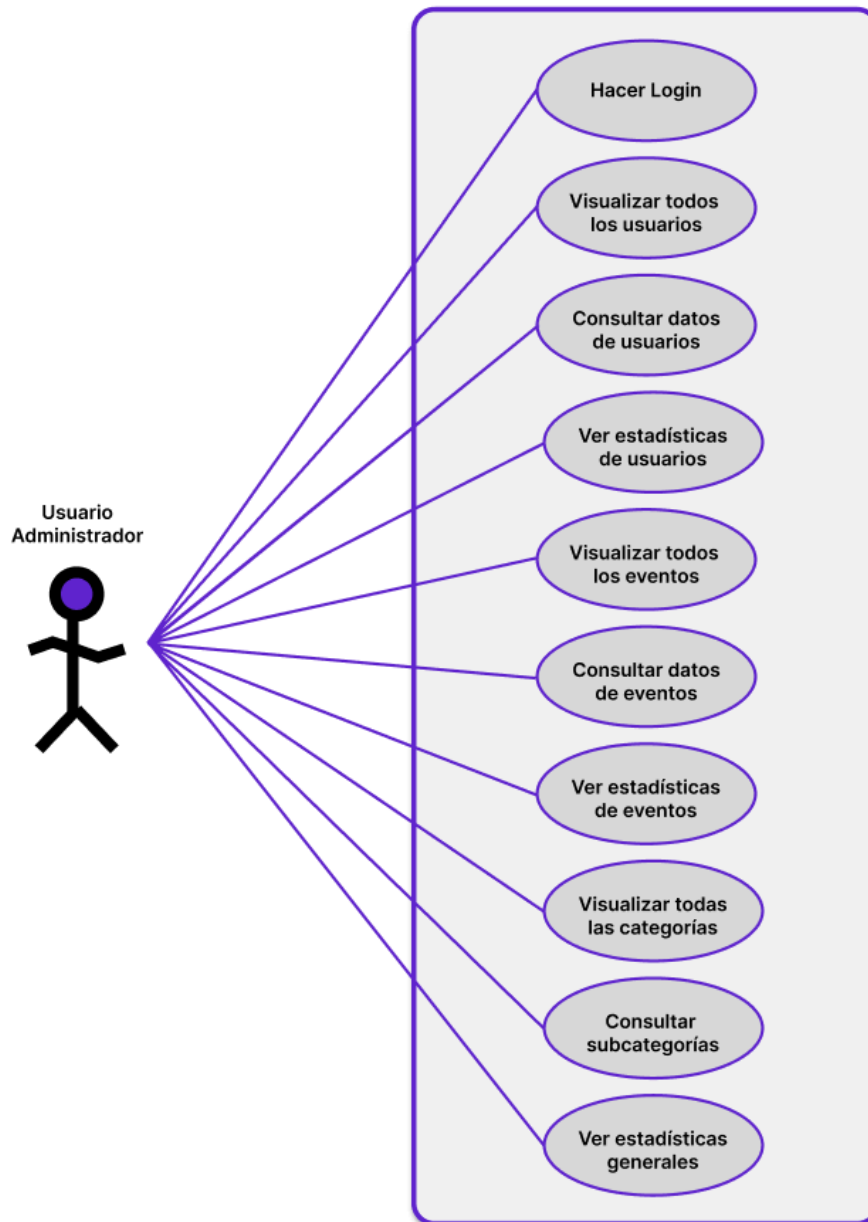
Título del Proyecto: EntreHobbies



JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

Usuario Administrador



Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: EntreHobbies

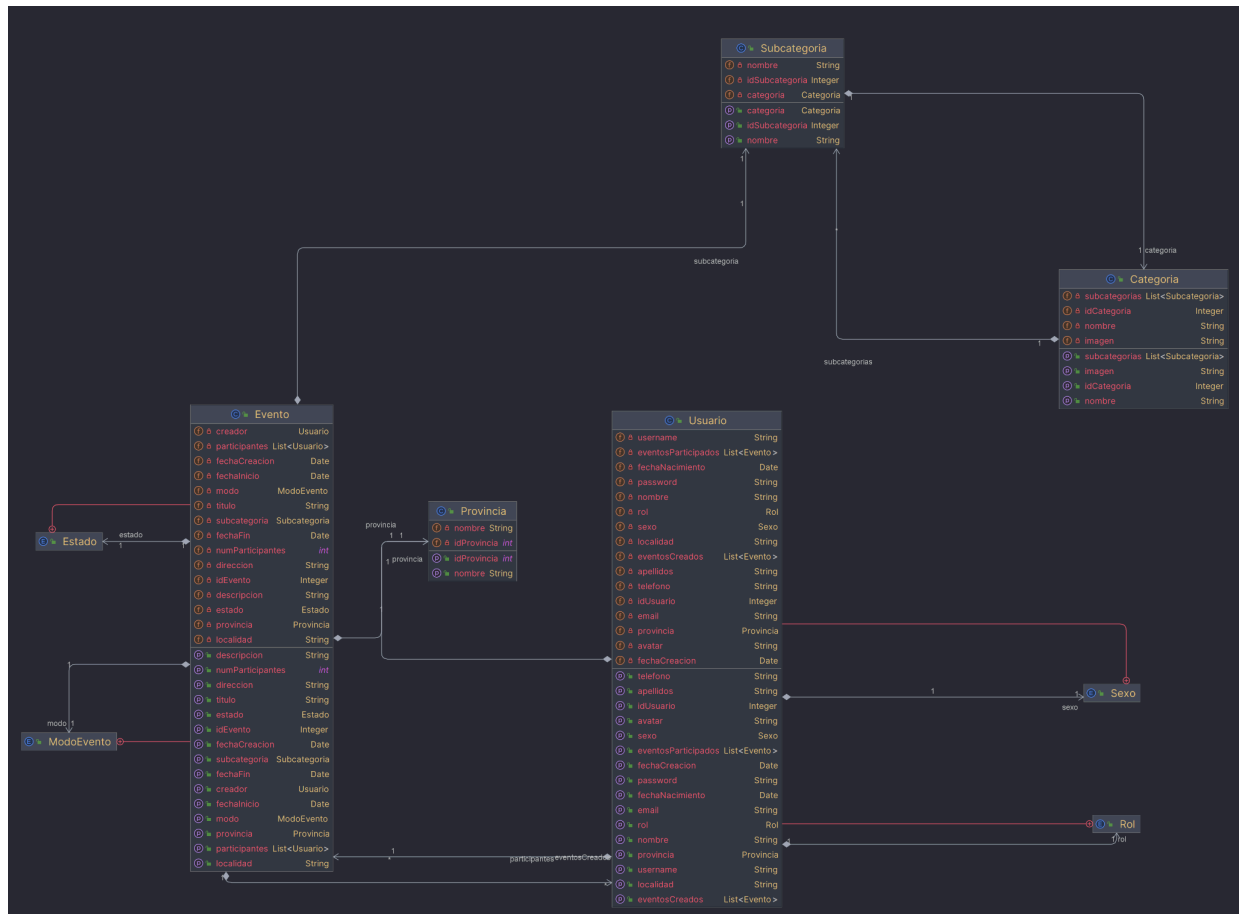


JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

Diagrama de clases

En este diagrama se representan las principales clases del sistema, así como sus atributos, métodos y relaciones entre ellas. Algunas de las clases más relevantes son:



Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: EntreHobbies

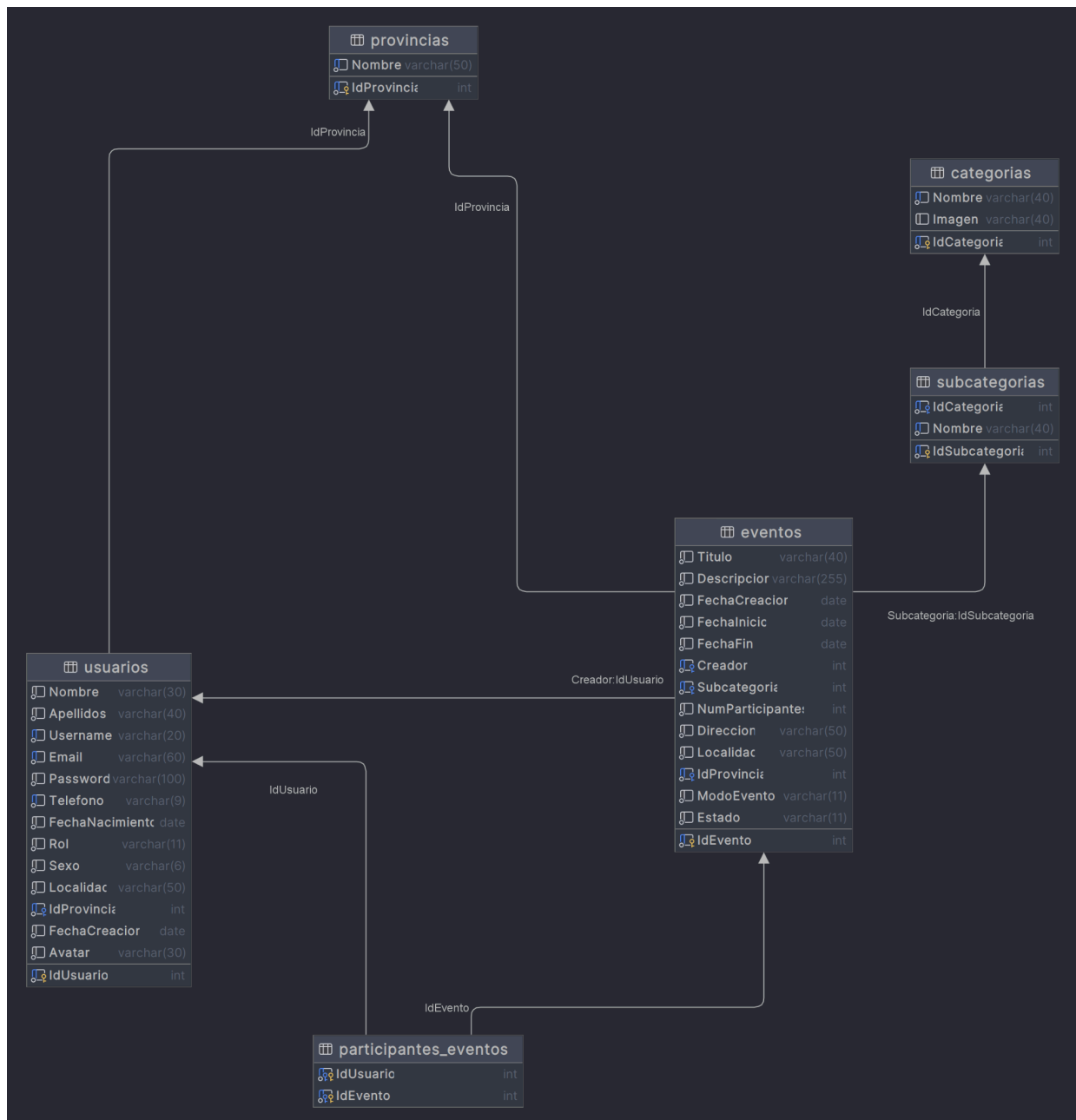


JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

Diagrama entidad-relación(ER)

Se ha diseñado un modelo relacional normalizado que define la estructura de la base de datos.

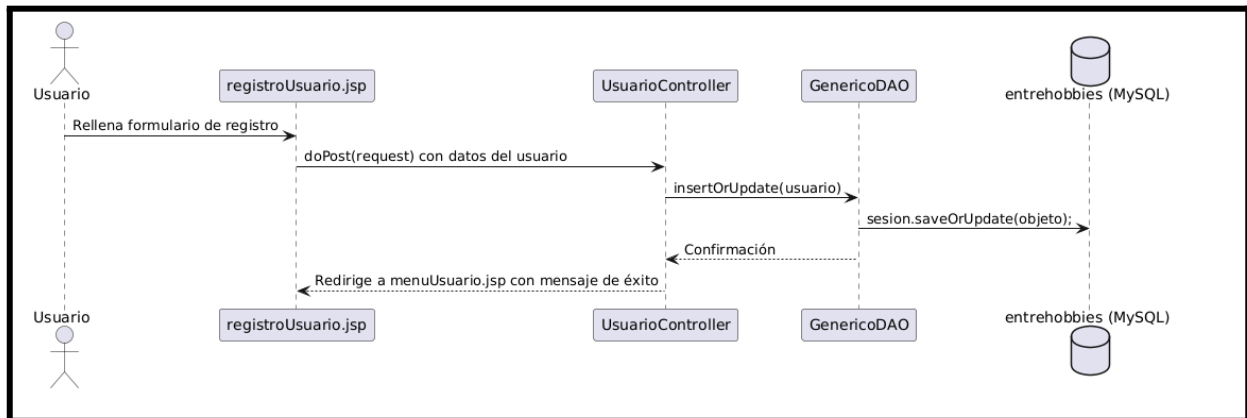


3.2. Desarrollo (diagrama de secuencias,...)

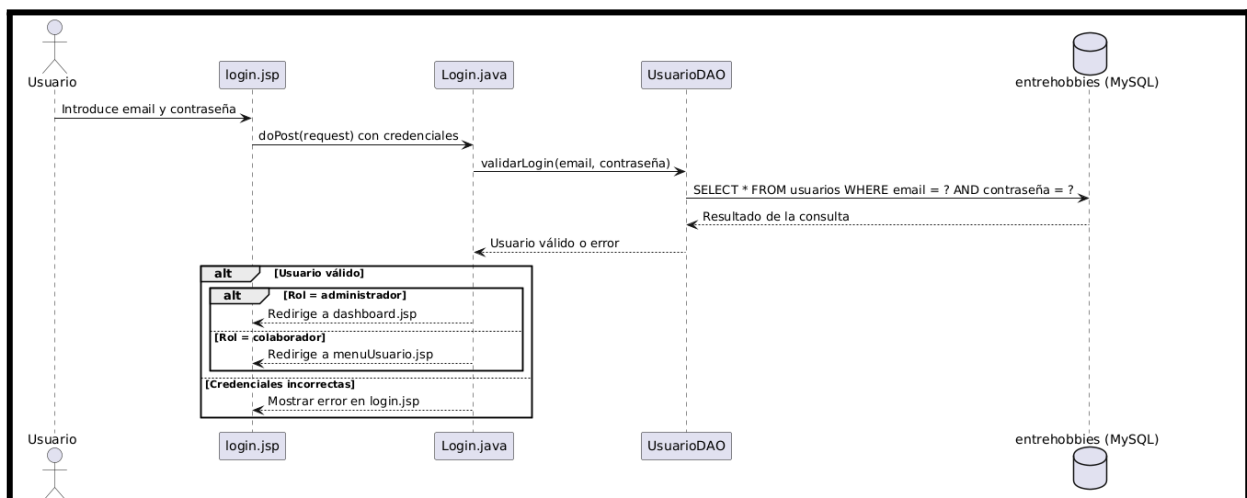
Durante la fase de desarrollo de la aplicación **EntreHobbies**, se han definido distintos diagramas que permiten representar de forma visual el comportamiento interno del sistema. Estos diagramas ayudan a comprender cómo interactúan los distintos componentes de la aplicación ante las acciones realizadas por el usuario.

A continuación, se incluyen los diagramas más relevantes desarrollados en esta fase.

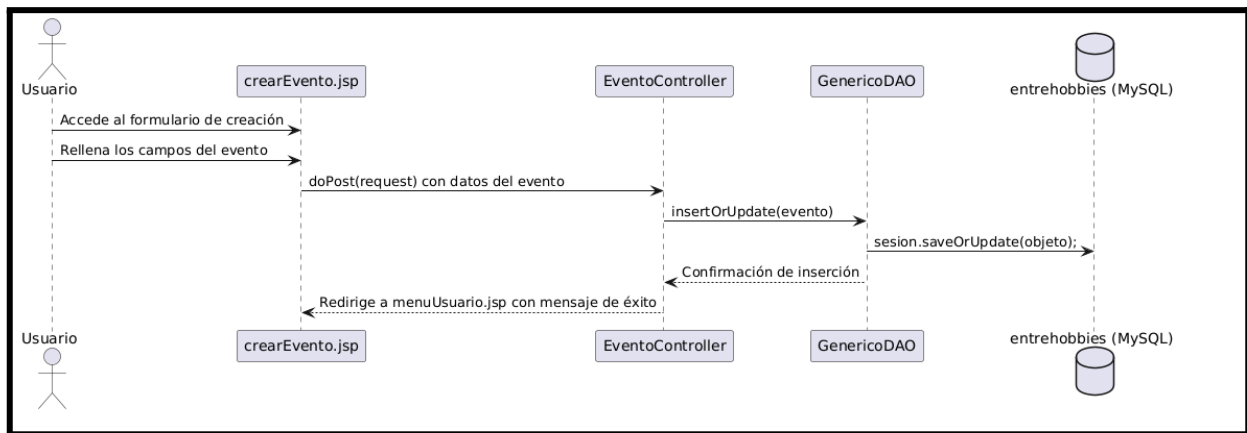
Registro de usuario



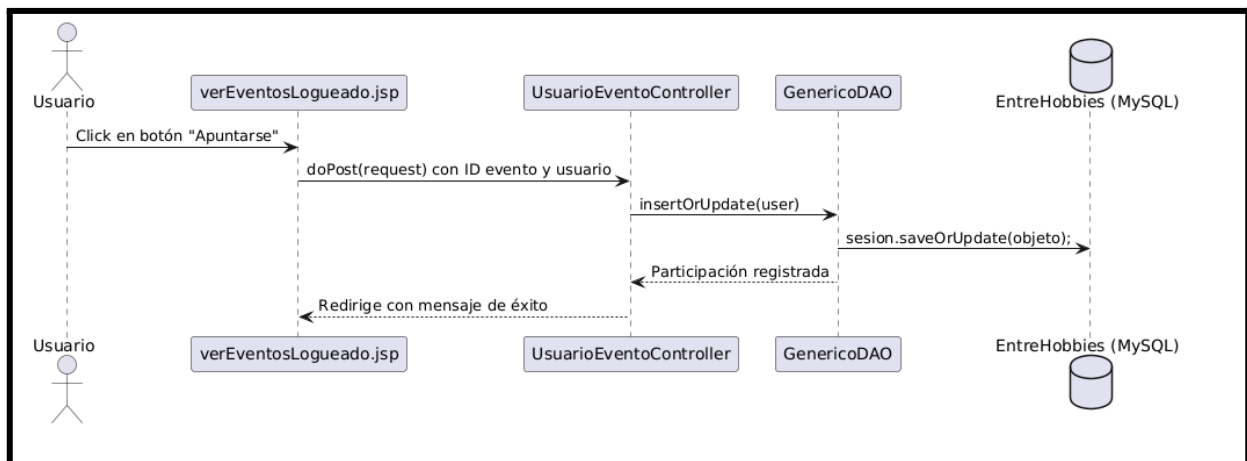
Inicio de sesión



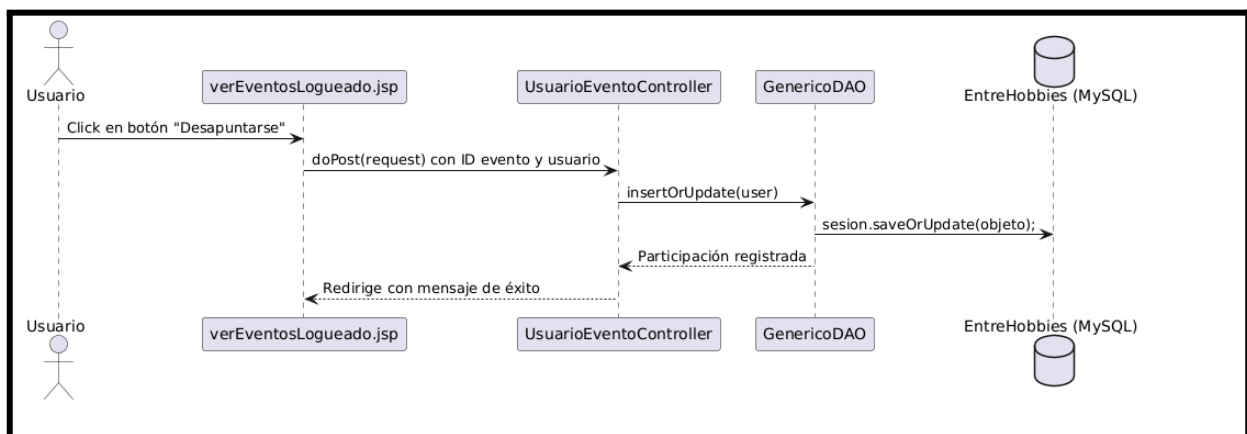
Crear evento



Apuntarse a un evento



Desapuntarse de un evento



3.3. Pruebas realizadas

Con el objetivo de asegurar la fiabilidad y estabilidad de la aplicación **EntreHobbies**, se llevaron a cabo pruebas funcionales manuales centradas en las operaciones más relevantes para los usuarios registrados (**colaboradores**). Estas pruebas se ejecutaron en el entorno de desarrollo local, utilizando datos simulados representativos y analizando los resultados obtenidos para verificar el comportamiento del sistema.

Se validaron funcionalidades como el inicio de sesión, el registro de nuevos usuarios, la participación en eventos y la creación de actividades, observando el comportamiento esperado del sistema y la correcta persistencia de datos en la base de datos.

A continuación, se describen las pruebas más significativas realizadas durante el desarrollo:

Prueba	Componente	Datos de entrada	Resultado esperado	Resultado Obtenido
Inicio de sesión	Login.java	Email: usuario@correo.com Contraseña: 1234	Redirección a menuUsuario.jsp o dashboard.jsp	Redirección a login.jsp junto al mensaje de error "credenciales incorrectas"
Registro de usuario	UsuarioController	Nombre: Alberto Apellidos: Sánchez Macías Email: nuevo@correo.com Contraseña: 123456 Repetir contraseña: 123456 Teléfono: 666777888 Provincia: Badajoz Fecha de nacimiento: 1994-03-26 Género: Hombre Acepta condiciones: Sí	Usuario insertado y redirección a menuUsuario.jsp junto mensaje de éxito "Te has registrado correctamente"	Usuario insertado y redirección a menuUsuario.jsp junto mensaje de éxito "Te has registrado correctamente"

Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: EntreHobbies



JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

Participación en evento	UsuarioEventoController	ID Usuario: 5 ID Evento: 12	Registro en tabla participaciones sin duplicación	Registro en tabla participaciones sin duplicación
Creación de evento	EventoController	Título: Torneo de ajedrez Descripción: Torneo amistoso para jugadores de todos los niveles. Número máximo de participantes: 12 Categoría: Juegos de mesa Subcategoría: Ajedrez Dirección: Calle Benito N° 25 Localidad: Mérida Provincia: Badajoz Fecha inicio: 2025-06-15 Fecha fin: 2025-05-16	Evento guardado y visible en gestionEventos.jsp	Evento guardado y visible en gestionEventos.jsp

4. Proceso de despliegue

A continuación, se detallan las herramientas y versiones utilizadas durante el desarrollo, así como aquellas necesarias para el despliegue y correcto funcionamiento de la aplicación EntreHobbies:

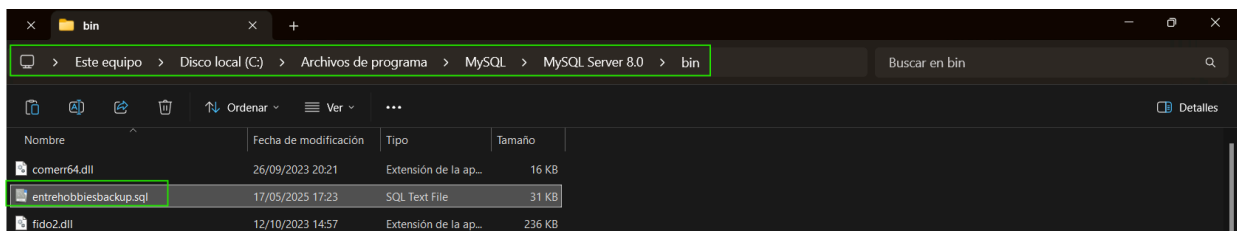
4.1. Requisitos de sistema

Herramienta	Versión	Enlace de descarga
Java Development Kit	JDK 11	Enlace descarga JDK 11
Apache Tomcat	9.0	Enlace descarga Apache 9.0
MySQL Server	8	Enlace descarga MySQL Server 8
IntelliJ IDEA	2023.3.4 o superior	Enlace descarga IntelliJ IDEA
Navegador web	Chrome, Firefox	Enlace descarga Chrome

4.2. Carga del script SQL de EntreHobbies en el servidor

Paso 1

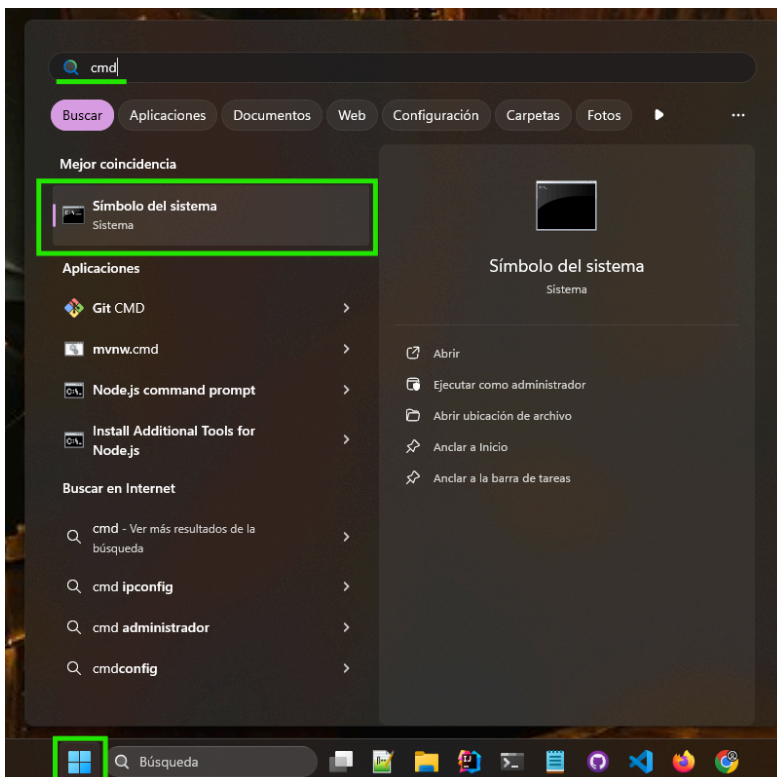
Copiaremos el archivo con nombre “**entrehobbiesbackup.sql**” en el directorio bin de nuestro MySQL server.



Ruta de guardado → C:\Program Files\MySQL\MySQL Server 8.0\bin

Paso 2

Accedemos a la terminal de windows (cmd).



Paso 3

Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: EntreHobbies



JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

Una vez abierta la terminal nos situaremos en la ruta del directorio “bin” de MySQL Server.

```
Símbolo del sistema
Microsoft Windows [Versión 10.0.22631.5335]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\saul_>cd..

C:\Users>cd..

C:\>cd Program Files\MySQL\MySQL Server 8.0\bin
C:\Program Files\MySQL\MySQL Server 8.0\bin>
```

Comando 1 → cd

Comando 2 → cd Program Files\MySQL\MySQL Server 8.0\bin

Paso 4

En este paso se **carga el script SQL** con la estructura y datos de la base de datos **EntreHobbies** en el servidor MySQL del instituto, utilizando un comando de importación desde la consola.

```
Símbolo del sistema
Microsoft Windows [Versión 10.0.22631.5335]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\saul_>cd..

C:\Users>cd..

C:\>cd Program Files\MySQL\MySQL Server 8.0\bin
C:\Program Files\MySQL\MySQL Server 8.0\bin>MySQL -h informatica.iesalbarregas.com -u sanchez -p --port 7004 < entrehobbiesbackup.sql
```

Comando → MySQL -h informatica.iesalbarregas.com -u sanchez -p --port 7004 < entrehobbiesbackup.sql

Introducimos la contraseña.

```
Símbolo del sistema
Microsoft Windows [Versión 10.0.22631.5335]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\saul_>cd..

C:\Users>cd..

C:\>cd Program Files\MySQL\MySQL Server 8.0\bin
C:\Program Files\MySQL\MySQL Server 8.0\bin>MySQL -h informatica.iesalbarregas.com -u sanchez -p --port 7004 < entrehobbiesbackup.sql
Enter password: *****
C:\Program Files\MySQL\MySQL Server 8.0\bin>
```

Contraseña → 76267833V

Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: EntreHobbies



JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

Paso 5

Nos conectamos a la base de datos del instituto con mismo usuario y contraseña.

```
Símbolo del sistema
Microsoft Windows [Versión 10.0.22631.5335]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\saul_>cd..

C:\Users>cd..

C:\>cd Program Files\MySQL\MySQL Server 8.0\bin

C:\Program Files\MySQL\MySQL Server 8.0\bin>MySQL -h informatica.iesalbarregas.com -u sánchez -p --port 7004 < entrehobbiesbackup.sql
Enter password: *****

C:\Program Files\MySQL\MySQL Server 8.0\bin>MySQL -h informatica.iesalbarregas.com -u sánchez -p --port 7004
```

Comando → MySQL -h informatica.iesalbarregas.com -u sánchez -p --port 7004

Paso 6

Comprobamos que esté nuestra base de datos correctamente añadida al servidor.

```
Símbolo del sistema - MySQL
mysql> SHOW databases;
+-----+
| Database |
+-----+
| aviary   |
| entrehobbies |
| gestocarb |
| information_schema |
| mysql   |
| performance_schema |
| tiendaVirtual |
+-----+
7 rows in set (0.02 sec)
```

Comando → SHOW databases;

Paso 7

Procedemos a crear el usuario con el que atacaremos a la base de datos “entrehobbies”.

```
Símbolo del sistema - MySQL
mysql> SHOW databases;
+-----+
| Database |
+-----+
| aviary   |
| entrehobbies |
| gestocarb |
| information_schema |
| mysql   |
| performance_schema |
| tiendaVirtual |
+-----+
7 rows in set (0.02 sec)

mysql> USE entrehobbies;
Database changed
mysql> CREATE USER 'sanchezadminhobbies'@'localhost' IDENTIFIED BY 'AdminHobby*2025+';
```

Comando 1 → USE entrehobbies;

Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: EntreHobbies



JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

Comando 2 → CREATE USER 'sanchezadminhobbies'@'localhost' IDENTIFIED BY 'AdminHobby*2025+';

Le damos todos los permisos sobre la base de datos “entrehobbies” al usuario creado.

```
mysql> CREATE USER 'sanchezadminhobbies'@'localhost' IDENTIFIED BY 'AdminHobby*2025+';
Query OK, 0 rows affected (0.04 sec)

mysql> GRANT ALL ON entrehobbies.* TO 'sanchezadminhobbies'@'localhost';
Query OK, 0 rows affected (0.10 sec)
```

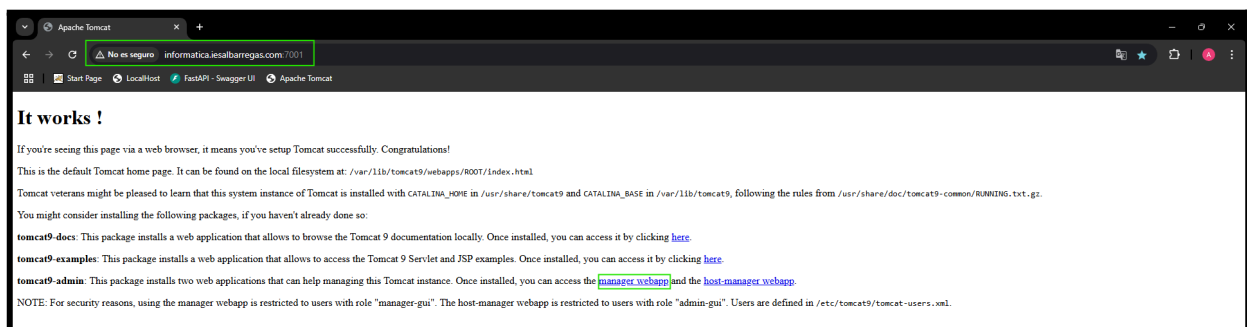
Comando → GRANT ALL ON entrehobbies.* TO 'sanchezadminhobbies'@'localhost';

En este punto, la base de datos ya se encuentra correctamente alojada en el servidor, junto con el usuario configurado para gestionar y establecer conexión desde la aplicación.

4.3. Subida de la aplicación al servidor del instituto

Paso 1

Nos dirigimos a la siguiente dirección <http://informatica.iesalbarregas.com:7001/> y seleccionamos manager web app.



Nos pedirá las credenciales de acceso.

Usuario → tomcat

Contraseña → tomcat

Paso 2

Dentro del apartado “Archivo WAR a desplegar” seleccionamos la opción de “Seleccionar archivo”.

Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: EntreHobbies



JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

Apache Tomcat Manager

Mensaje: FALLO - Se ha especificado una trayectoria inválida de contexto [null]

Gestor

Lista Aplicaciones	Ayuda HTML de Gestor	Ayuda de Gestor	Estado de Servidor
--------------------	----------------------	-----------------	--------------------

Ruta	Versión	Nombre a Mostrar	Ejecutándose	Sesiones	Comandos
/	Ninguno especificado		true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar > 30 minutos
/TiendaWeb	Ninguno especificado		true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar > 30 minutos
/dt	Ninguno especificado		true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar > 30 minutos
/host-manager	Ninguno especificado	Tomcat Host Manager Application	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar > 30 minutos
/manager	Ninguno especificado	Tomcat Manager Application	true	2	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar > 30 minutos

Desplegar

Desplegar directorio o archivo WAR localizado en servidor

Trayectoria de Contexto (opcional):
Version (for parallel deployment):
URL de archivo de Configuración XML:
URL de WAR o Directorio:

Desplegar

Archivo WAR a desplegar

Selecciona archivo WAR a cargar Seleccionar archivo Ningún archivo seleccionado

Desplegar

Paso 3

Seleccionamos el archivo “.war” que se ha proporcionado (“EntreHobbies.war”).

Abrir

DAW2 > AA-TFC > WAR-ENTREHOBBIES

Organizar Nueva carpeta

Nombre	Fecha de modificación	Tipo
EntreHobbies.war	16/05/2025 13:49	Archivo WAR

Nombre: EntreHobbies.war

Todos los archivos (*.*)

Abrir Cancelar

Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: EntreHobbies



JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

Paso 4

Comprobamos que se ha seleccionado correctamente el archivo war indicado y hacemos clic en “Desplegar”.

Ruta	Versión	Nombre a Mostrar	Ejecutándose	Sesiones	Comandos
/	Ninguno especificado		true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/ProyectoTFG-1.0	Ninguno especificado		true	1	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/TiendaWeb	Ninguno especificado		true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/idc	Ninguno especificado		true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/host-manager	Ninguno especificado	Tomcat Host Manager Application	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/manager	Ninguno especificado	Tomcat Manager Application	true	2	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos

Desplegar
Desplegar directorio o archivo WAR localizado en servidor

Trayectoria de Contexto (opcional):
Version (for parallel deployment):
URL de archivo de Configuración XML:
URL de WAR o Directorio:

Archivo WAR a desplegar
Seleccione archivo WAR a cargar: [EntreHobbies.war](#)

Paso 5

El servidor nos notificará en el apartado de “Mensaje” que se ha subido correctamente la aplicación (OK). Y ya podremos ver la aplicación en la lista de “Aplicaciones”.

Mensaje:

Ruta	Versión	Nombre a Mostrar	Ejecutándose	Sesiones	Comandos
/	Ninguno especificado		true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/EntreHobbies	Ninguno especificado		true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/ProyectoTFG-1.0	Ninguno especificado		true	1	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/TiendaWeb	Ninguno especificado		true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/idc	Ninguno especificado		true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/host-manager	Ninguno especificado	Tomcat Host Manager Application	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/manager	Ninguno especificado	Tomcat Manager Application	true	2	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos

En este punto ya se puede acceder a la aplicación correctamente. Se puede hacer clicando el nombre de la aplicación o accediendo directamente a la siguiente dirección: <http://informatica.iesalbarregas.com:7001/EntreHobbies/>

5. Propuestas de mejoras

5.1. Mejoras aplicadas durante el desarrollo

Durante el proceso de desarrollo de **EntreHobbies**, se han identificado y aplicado diversas mejoras con respecto al diseño inicial del proyecto. Estas mejoras surgieron de la experiencia práctica de implementación, la detección de necesidades no previstas en la fase de análisis, y la optimización de la experiencia de usuario. A continuación, se detallan las más relevantes.

- **Visualización de estadísticas generales sobre la aplicación (Usuario Administrador).** Ahora el usuario administrador puede consultar estadísticas generales de la aplicación. Entre las estadísticas disponibles se incluyen: número de eventos por categoría, distribución de usuarios por sexo, y provincias con mayor volumen de eventos creados.
- **Visualización de estadísticas de usuarios (Usuario Administrador).** Como pueden ser: usuario con más eventos creados, usuario con más participaciones, total de usuarios registrados y número de usuarios registrados por mes.
- **Visualización de estadísticas de eventos (Usuario Administrador).** Se pueden ver estadísticas como: eventos totales en la aplicación, evento con más participaciones, evento más reciente creado y número de eventos creados por mes.
- **Automatización de actualización diaria de los eventos.** La aplicación todos los días a las 00:00 realizará dos actualizaciones: los eventos que empiecen ese día pasarán a estar “En curso” y los eventos que finalizan ese día pasarán a “Finalizado”. Esta automatización garantiza que la información mostrada en la aplicación esté siempre actualizada sin intervención manual.
- **Envío de emails a usuarios colaboradores.** La aplicación ahora realiza envíos automáticos a los usuarios cuando se realizan las siguientes funcionalidades: Registrarse (Mensaje de bienvenida), modificar un evento con

participantes (Notificación a participantes de evento modificado) y cancelar un evento con participantes (Notificación a participantes de evento cancelado).

5.2. Mejoras propuestas para futuras versiones

Además de las funcionalidades ya implementadas, se han identificado una serie de mejoras que podrían incorporarse en versiones futuras de **EntreHobbies**. Estas propuestas tienen como objetivo ampliar las capacidades de la plataforma, mejorar la experiencia de usuario y optimizar la gestión de eventos y usuarios.

- **Gestión de usuarios (Usuario Administrador).** Que el administrador pueda modificar o eliminar un usuario en la aplicación.
- **Gestión de eventos (Usuario Administrador).** Que el administrador pueda modificar o eliminar un evento en la aplicación.
- **Gestión de categorías(Usuario Administrador).** Que el administrador pueda modificar o una categoría para poder añadir nuevas subcategorías.
- **Creación de nuevas categorías y subcategorías (Usuario Administrador).** Que el administrador pueda crear nuevas categorías junto con sus subcategorías.
- **Iniciar/Finalizar un evento (Usuario Colaborador).** Actualmente, el estado de los eventos se actualiza de forma automática según la fecha establecida. Sin embargo, se propone incorporar una funcionalidad que permita al usuario colaborador iniciar o finalizar sus eventos manualmente desde su panel de gestión. Esta mejora ofrecería mayor flexibilidad en la administración de los eventos, permitiendo adaptarse a cambios de última hora o situaciones imprevistas.

6. Bibliografía

Documentación oficial de tecnologías empleadas

- Java 11 - <https://docs.oracle.com/en/java/javase/11/>
- Apache Tomcat 9 - <https://tomcat.apache.org/tomcat-9.0-doc/index.html>
- MySQL 8 - <https://dev.mysql.com/doc/refman/8.0/en/>
- IntelliJ IDEA - <https://www.jetbrains.com/idea/resources/>

Librerías de Frontend

- Bootstrap 5 - <https://getbootstrap.com/docs/5.3/getting-started/introduction/>
- Bootstrap Table - <https://bootstrap-table.com/docs/getting-started/introduction/>
- Chart.js - <https://www.chartjs.org/docs/latest/getting-started/>
- HTML, CSS y JavaScript - <https://manz.dev/>

API y clases utilizadas

- Java EE 7 API - <https://docs.oracle.com/javaee/7/api/>
- Jakarta Mail API - <https://javadoc.io/doc/jakarta.mail/jakarta.mail-api/latest/jakarta.mail/module-summary.html>
- Interface ServletRequestListener - <https://docs.oracle.com/javaee/7/api/javax/servlet/ServletRequestListener.html>

Creación diagramas

- PlantUML - <https://plantuml.com/es/starting>
- Figma - <https://www.figma.com/>

Foros y consultas frecuentes

- stackoverflow - <https://stackoverflow.com/questions>