*PROYECTO***:**

**CampestresCYL**



*Realizado por:*

***Los Showdowns***

* ***Asier Sanz Madrigal***
* ***Jorge Toribio Fernández***

|  |  |
| --- | --- |
| http://alba9.es/imagenes/logo_galileo_1.jpg | **Desarrollo de Aplicaciones Web (DAW)** |
| **Valladolid, a 22 de Enero de 2026** |

Contenido

[1. Introducción 2](#_Toc219982466)

[2. Análisis 2](#_Toc219982467)

[2.1. *Dataset* utilizado 2](#_Toc219982468)

[2.2. Requisitos funcionales 2](#_Toc219982469)

[2.3. Diagramas de casos de uso 2](#_Toc219982470)

[3. Diseño 3](#_Toc219982471)

[3.1. Modelo de base de datos 3](#_Toc219982472)

[3.2. Diseño de la interfaz 3](#_Toc219982473)

[4. Desarrollo 3](#_Toc219982474)

[4.1. *Stack* tecnológico 3](#_Toc219982475)

[4.2. Estructura de carpetas 3](#_Toc219982476)

[4.3. Descripción del funcionamiento 3](#_Toc219982477)

[5. Pruebas 3](#_Toc219982478)

[6. Despliegue 3](#_Toc219982479)

[7. Sostenibilidad y "Green Coding" 4](#_Toc219982480)

[8. Conclusiones 4](#_Toc219982481)

[8.1. Autoevaluación 4](#_Toc219982482)

[8.2. Líneas futuras 4](#_Toc219982483)

[9. Bibliografía 4](#_Toc219982484)

# Introducción

El proyecto constara con la ubicación e información de los diferentes campings que se pueden localizar en Castilla y León haciendo uso, del conjunto de datos “Registro de camping”.

El trabajo consta de una página web que visualizará los campamentos existentes en Castilla y León. La finalidad es dar a conocer la ubicación exacta de los campings, así como ofrecer una herramienta para poder guardar y mostrar su información para su uso futuro.

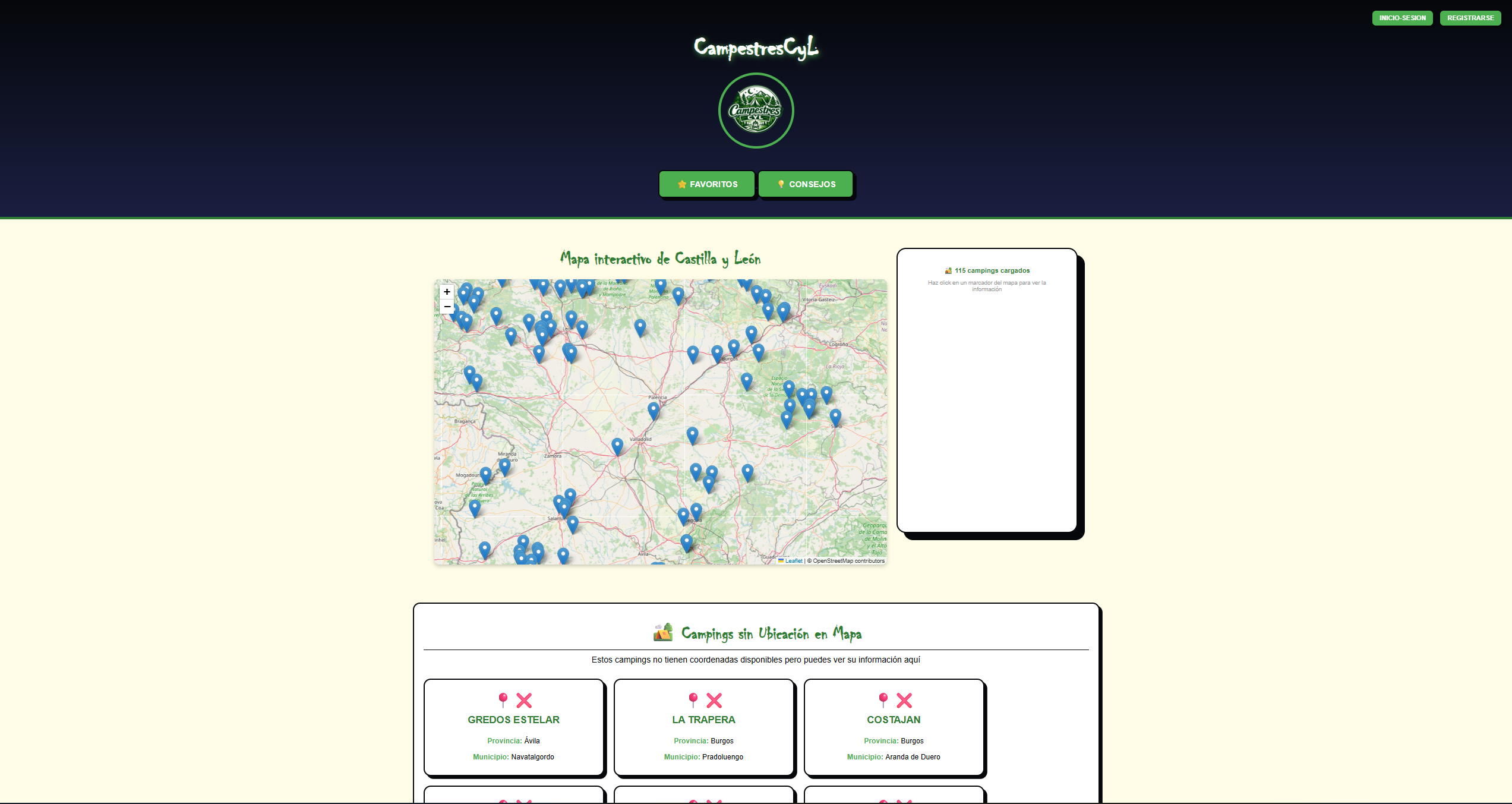


Figura 1. Ejemplo de leyenda numerada para una imagen o diagrama.

Tabla 1. Ejemplo de leyenda numerada para una tabla.

|  |  |  |
| --- | --- | --- |
| Columna 1 | Columna 2 | Columna 3 |
| Dato 1 | Dato 2 | Dato 3 |
| Mapa de castilla y león | Cuadro con información de los campings | Botón de registro / inicio de sesión |
| Sección de Favoritos | Consejos para campistas |  |

# Análisis

## *Registro de campings*

* **Nombre:** Registro de Turismo de Castilla y León.
* **URL:** [/api/explore/v2.1/catalog/datasets/registro-de-turismo-de-castilla-y-leon/records?limit=20&refine=establecimiento%3A%22Campings%22](https://analisis.datosabiertos.jcyl.es/api/explore/v2.1/catalog/datasets/registro-de-turismo-de-castilla-y-leon/records?limit=20&refine=establecimiento%3A%22Campings%22)
* **Campos utilizados:** Categoría, nombre, dirección, provincia, municipio, localidad, núcleo, teléfonos, email, web, posición, plazas

## Requisitos funcionales

|  |  |
| --- | --- |
| RF | Registro |
| Versión | 1.0 |
| Autores | Asier Sanz y Jorge Toribio |
| Descripción | El sistema debe permitir a un usuario registrarse en la página web. |
| Objetivos | Permitir a un usuario poder interactuar con las funciones de la página. |
| Precondición | El usuario debe haber entrado a la página. |
| Secuencia Normal | Rellena un formulario para la solicitud de registro. |
| Postcondición | El usuario puede iniciar sesión. |
| Excepción | Las credenciales del formulario pueden ser incorrectas. |
| Rendimiento | Normal |
| Importancia | Alta |
| Urgencia | Máxima |
| Estado | Operativo |
| Estabilidad | Alta |

|  |  |
| --- | --- |
| RF | Login |
| Versión | 1.0 |
| Autores | Asier Sanz y Jorge Toribio |
| Descripción | El sistema debe permitir a un usuario iniciar sesión en la página web. |
| Objetivos | Permitir a un usuario poder interactuar con las funciones de la página. |
| Precondición | El usuario debe haberse registrado. |
| Secuencia Normal | Rellena un formulario para la solicitud de login. |
| Postcondición | El usuario se ha logueado con éxito |
| Excepción | Las credenciales del formulario pueden ser incorrectas |
| Rendimiento | Normal |
| Importancia | Alta |
| Urgencia | Máxima |
| Estado | Operativo |
| Estabilidad | Alta |

|  |  |
| --- | --- |
| RF | Mostrar Mapa de Campings |
| Versión | 1.0 |
| Autores | Asier Sanz y Jorge Toribio |
| Descripción | El sistema debe permitir a un usuario visualizar el mapa de campings. |
| Objetivos | Incorporar de manera correcta la API de campings. |
| Precondición | Ninguna. |
| Secuencia Normal | El usuario al entrar podrá observar en un mapa los puntos donde se ubican los campings. |
| Postcondición | Poder interactuar con los puntos. |
| Excepción | La información de la API cambie con el paso del tiempo. |
| Rendimiento | Normal |
| Importancia | Alta |
| Urgencia | Máxima |
| Estado | Operativo |
| Estabilidad | Alta |

|  |  |
| --- | --- |
| RF | Desplegar información de camping |
| Versión | 1.0 |
| Autores | Asier Sanz y Jorge Toribio |
| Descripción | El sistema debe permitir a un usuario al dar click en uno de los puntos visualizar la información del camping. |
| Objetivos | Incorporar de manera correcta la API de campings. |
| Precondición | Visualizar el mapa de campings. |
| Secuencia Normal | El usuario al dar click en uno de los puntos podrá ver la información relacionado con el camping en cuestión. |
| Postcondición | Poder interactuar con la información para añadir a favoritos. |
| Excepción | La información de la API cambie con el paso del tiempo. |
| Rendimiento | Normal |
| Importancia | Alta |
| Urgencia | Máxima |
| Estado | Operativo |
| Estabilidad | Alta |

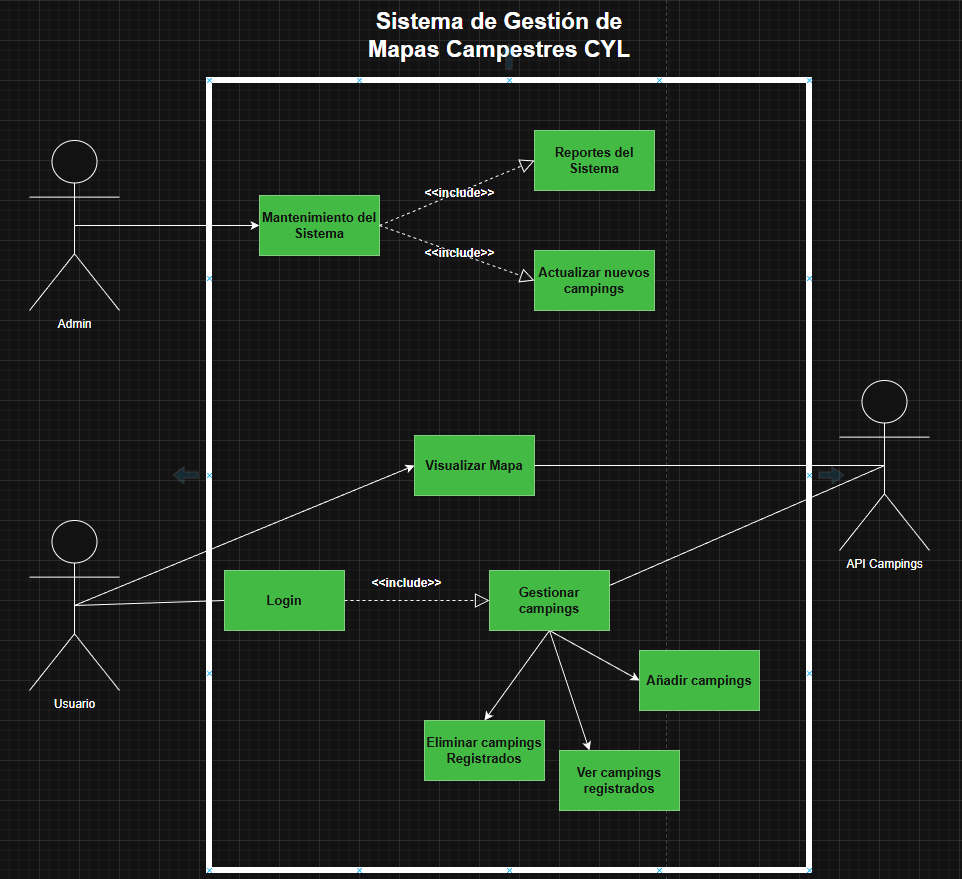
|  |  |
| --- | --- |
| RF | Mostrar Favoritos para usuario logueado |
| Versión | 1.0 |
| Autores | Asier Sanz y Jorge Toribio |
| Descripción | El sistema debe permitir a un usuario al seleccionar favoritos redirigirle a una sección donde visualice los campings registrados por favoritos. |
| Objetivos | Control de modelos y bases de datos. |
| Precondición | Un usuario debe de haber iniciado sesión. |
| Secuencia Normal | El usuario logueado podrá dar a la opción de redirigirse a la pestaña favoritos para ver el número de campings registrados. |
| Postcondición | De la pestaña favoritos poder eliminarlos de la lista. |
| Excepción | Error en la captura de controles y modelos. |
| Rendimiento | Normal |
| Importancia | Alta |
| Urgencia | Alta |
| Estado | Operativo |
| Estabilidad | Alta |

|  |  |
| --- | --- |
| RF | Agregar campings a favoritos |
| Versión | 1.0 |
| Autores | Asier Sanz y Jorge Toribio |
| Descripción | El sistema debe permitir a los usuarios logueados añadir campings a la sección de favoritos. |
| Objetivos | Control de modelos y bases de datos. |
| Precondición | El usuario debe haber iniciado sesión-. |
| Secuencia Normal | El usuario tras desplegar la información de uno de los campings, tendrá la opción de poder añadirlos a favoritos en ese mismo contenedor. |
| Postcondición | Podrá ver en la sección de favoritos su listado de campings actualizado. |
| Excepción | Mal manejo de controles y modelos. |
| Rendimiento | Normal |
| Importancia | Alta |
| Urgencia | Alta |
| Estado | Operativo |
| Estabilidad | Alta |

|  |  |
| --- | --- |
| RF | Eliminar campings de favoritos |
| Versión | 1.0 |
| Autores | Asier Sanz y Jorge Toribio |
| Descripción | El sistema debe permitir a los usuarios logueados eliminar campings de la sección de favoritos. |
| Objetivos | Control de modelos y bases de datos. |
| Precondición | El usuario debe haber iniciado sesión. |
| Secuencia Normal | El usuario tras desplegar el listado de los campings, tendrá la opción de poder eliminar de favoritos los campings seleccionados. |
| Postcondición | Tras eliminar los campings seleccionados podremos ver el nuevo listado tras recargar la sección. |
| Excepción | Mal manejo de controles y modelos. |
| Rendimiento | Normal |
| Importancia | Alta |
| Urgencia | Alta |
| Estado | Operativo |
| Estabilidad | Alta |

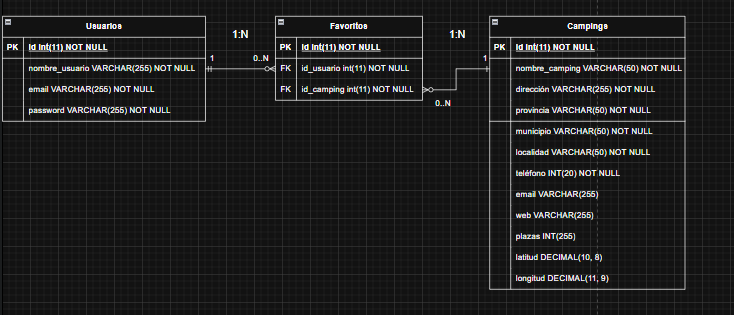
## Diagramas de casos de uso

Ejemplo: <https://drawio-app.com/blog/uml-use-case-diagrams-with-draw-io/>



# Diseño

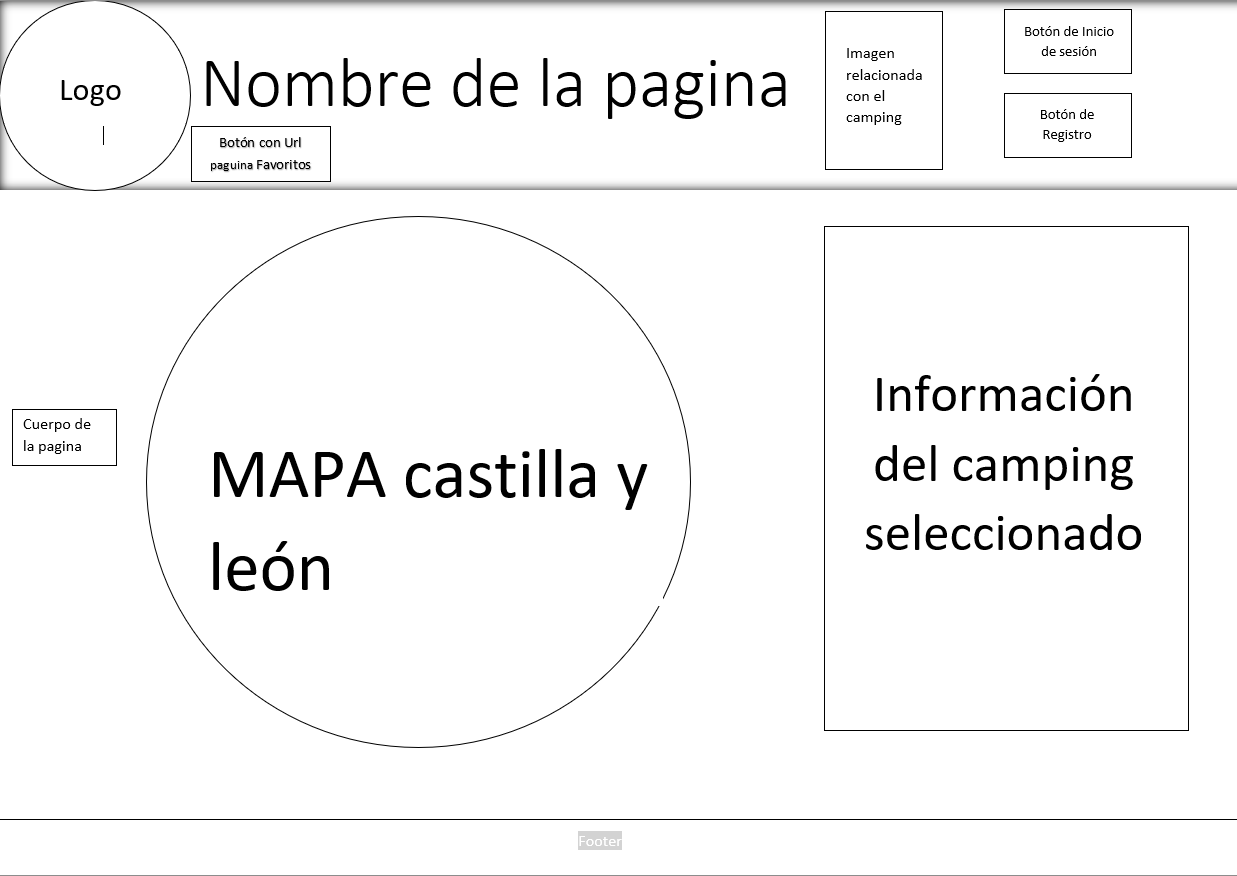
## Modelo de base de datos

* **Diagrama Entidad-Relación:**
* **Justificación:**
* **Tabla usuarios:** Almacena la información de los usuarios registrados en el sistema. Cada usuario se identifica de forma única.
* **Tabla favoritos:** Implementa una relación muchos a muchos (N:M) entre usuarios y campings, permitiendo que cada usuario pueda marcar múltiples campings como favoritos.
* **Tabla Campings:** Es la tabla principal del sistema, almacenando toda la información detallada de los campings de Castilla y León obtenida desde la API de Datos Abiertos de la Junta de Castilla y León.

## Diseño de la interfaz

* + **Prototipo** del *layout* principal para las vistas *desktop* y *mobile*.

Layout vista Desktop

* + 

Version grafica desktop versión

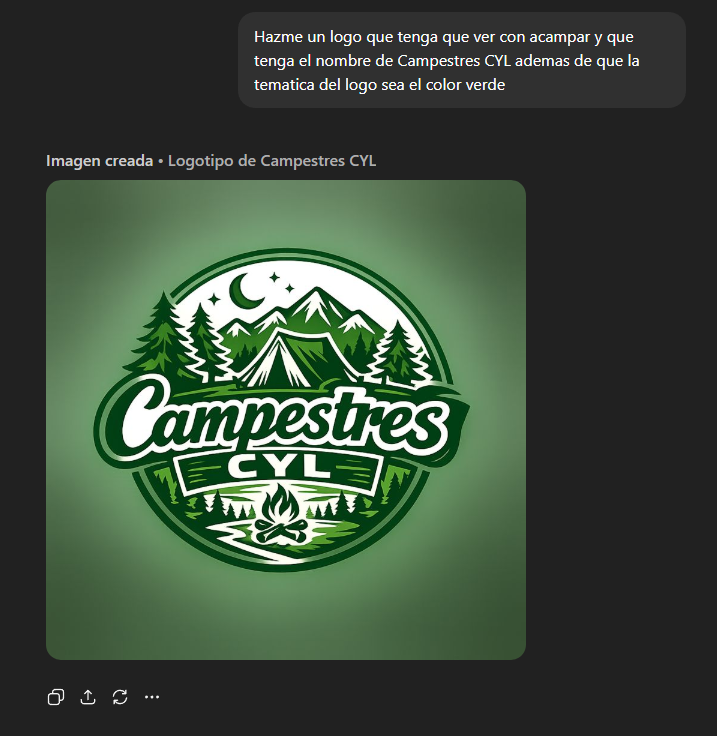


Layout vista mobile Version grafica mobile versión

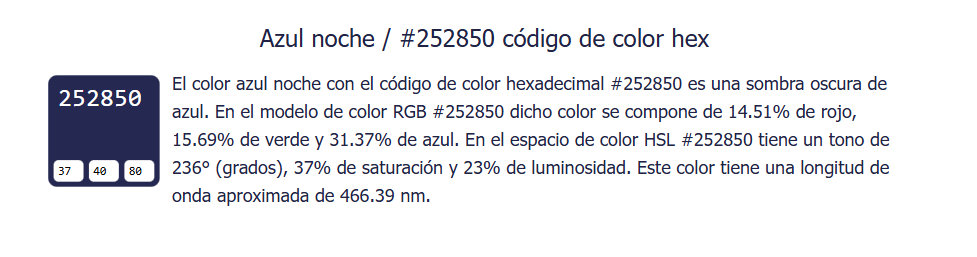


* + Breve **guía de estilos** general que incluya la paleta de colores utilizada y la selección de fuentes.

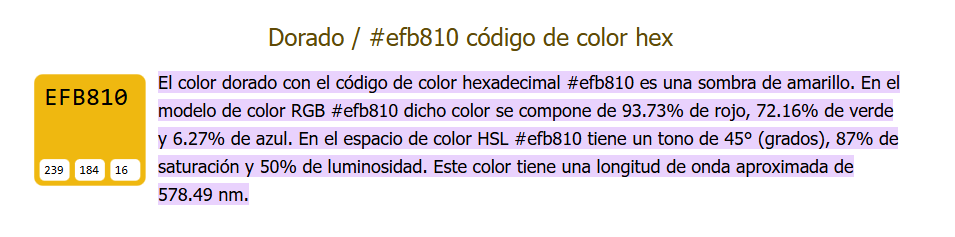
Para la creación del logo nos hemos dado a la labor de crearlo con la herramienta de chat gpt para ello le hemos pedido atraves de la siguiente orden que nos cree el logo con el siguiente contexto “***Hazme un logo que tenga que ver con acampar y que tenga el nombre de Campestres CYL ademas de que la temática del logo sea el color verde***” una vez pasado a chat gpt el siguiente contexto nos ha proporcionado el logo que hemos usado en el proyecto.



Para los colores del fondo del Header y Footer, he usado el color de la noche que representa los mejores momentos en donde el estar de camping es lo mejor para ello he usado el siguiente código hex para llevar a cabo dicho color en el proyecto.



Para los colores de la caja en la que se encuentra la información he usado un color el cual se usa para representar las estrellas de la noche que al igual que en el proyecto es lo más destacado a encontrar.



**Tipografia utilizada**

\*Título Principal

Fuente de Word “Chiller”.

**Texto: Chiller/80.**

\*Inicio de sesión Login/Registro/Favoritos

Fuente de Word “Calibri cuerpo”.

**Texto: Calibri/16.**

\*Información del camping

\*Titulo (nombre del camping).

Fuente “Segoe UI Emoji”.

**Texto: Segoe UI Emoji / 18.**

\*Títulos (apartados de información sobre el camping).

Fuente “Times New Roman”

**Text: Times New Roman / 11.**

Información de los apartados.

**Fuente “Times New Roman/ 8.**

# Desarrollo

## *Stack* tecnológico

* **Entorno:**
  + **Front-End:** Javascript ECMAScript 2025, HTML5 y CSS3. Framework utilizado de Javascript Leaflet.
  + **Bases de Datos:** MySQL, phpmyAdmin 5.2.1.
  + **Back-End:** PHP 8.2.12 nativo y Composer 2.9.2.
  + **Despliegue:** XAMPP
* **Software:**
  + **Visual Studio Code 1.107.1.**
  + **XAMPP v3.3.0**
  + **Git version 2.52.0.windows.1**

## Estructura de carpeta

El proyecto sigue una arquitectura MVC (Modelo-Vista-Controlador) con separación clara entre el código fuente y los recursos públicos accesibles desde el navegador.

**Directorio Raíz**

los-showdowns/

├──. env # Variables de entorno (credenciales BD)

├──. gitignore # Archivos excluidos del control de versiones

├── composer.json # Dependencias de PHP y configuración de autoload

├── composer.lock # Versiones exactas de dependencias instaladas

├── cron.php # Script de sincronización automática con API externa

├── docs/ # Documentación del proyecto

├── public/ # Directorio público (Document Root)

├── routes/ # Definición de rutas de la aplicación

├── src/ # Código fuente (backend)

└── vendor/ # Dependencias de Composer (generado automáticamente)

/public/ Directorio Público

Nuestro front-controller, único directorio accesible desde el navegador web y contiene recursos estáticos (imágenes, etc.). Nuestro frontend en pocas palabras.

**public/**

├── .htaccess # Configuración de Apache (caché, compresión, seguridad)

├── index.php # Front Controller (punto de entrada único)

├── css/ # Hojas de estilo

│ ├── CampestresCyL.css

│ ├── Favoritos.css

│ ├── Registro.css

│ ├── Inicio-Sesión.css

│ └── 404.css

├── js/ # Scripts JavaScript (con defer y lazy loading)

│ ├── CampestresCyL.js # Lógica del mapa y campings

│ ├── Favoritos.js # Gestión de favoritos

│ ├── Registro.js # Validación de registro

│ ├── Inicio-Sesión.js # Validación de login

│ └── 404.js # Animaciones página de error

└── images/ # Recursos gráficos

└── CamPiMG/

├── Logo CampestresCyL.jpeg

├── Campestre.png

└── mapa\_provincias.png

* **index.php:** Front-controller que recibe todas las peticiones y las enruta mediante respuestas GET, redirige dichas respuestas mediante arrays whitelist.
* **.htaccess:** Configura caché HTTP (1 mes para CSS/JS/imágenes), comprensión y bloquea acceso a archivos sensibles.

**/src/**

src/

├── config/ # Configuración del sistema

│ ├── db.php # Clase Database (conexión PDO a MySQL)

│ └── database\_campestrecyl.sql # Script de creación de BD

├── controllers/ # Controladores (lógica de negocio)

│ ├── ApiController.php # Gestión de API externa y endpoints REST

│ ├── AuthController.php # Autenticación (login/registro)

│ └── WebController.php # Renderizado de vistas

├── models/ # Modelos (acceso a datos)

│ ├── Usuario.php # CRUD de usuarios y autenticación

│ ├── Campings.php # CRUD de campings

│ ├── Favoritos.php # Gestión de favoritos (relación N:M)

│ └── Imagenes.php # Gestión de imágenes

└── views/ # Vistas (HTML + PHP)

├── CampestresCyL.php # Página principal con mapa

├── Favoritos.php # Lista de favoritos del usuario

├── Registro.php # Formulario de registro

├── Inicio-Sesión.php # Formulario de login

├── Consejos.php # Página de consejos

└── 404.php

La carpeta config contiene ficheros que gestiona la conexión PDO con MySQL y la importación de los datos a MySQL.

La carpeta controllers contiene todos los controladores que gestionan la lógica principal del backend, trabajando con lógica y ejecutando funciones de los modelos, permite redirigir los datos para lo que se necesite.

La carpeta models contienen los modelos que permite trabajar con datos de MySQL

La carpeta views son las vistas que se generan a partir de nuestro único front-controller, gestionado por controladores.

**/routes/**

routes/

├── api.php # Rutas de API REST (JSON)

└── web.php # Rutas de vistas (HTML)

Define las rutas disponibles y mapea los controladores

**/docs/**

docs/

├── diagramas/

│ ├── Diagrama\_CasoUso\_CampestresCYL.drawio

│ └── Diagrama\_Entidad\_Relacion\_CampestresCYL.drawio

└── documentación/

├── Diseño de interfaz.docx

├── Guía de estilos.docx

└── DAW-PI-Documentos para alumnado/

Documentación técnica, diagramas y guías del proyecto

/vendor/

Librerías de terceros instaladas por Composer, generado automáticamente permite la funcionalidad del autoloader en la web.

## Descripción del funcionamiento

**Breve manual de usuario.**

**Acceso y Navegación**

El usuario puede acceder a la web y localizar información detallada de los distintos campings de Castilla y León a través del mapa interactivo. Al hacer clic sobre cualquier marcador del mapa, se despliega la información completa del camping seleccionado (nombre, ubicación, contacto, plazas disponibles, etc.).

**Funcionalidades sin Registro**

* Visualización del mapa interactivo con todos los campings de las 9 provincias.
* Consulta de información detallada de cada camping.
* Acceso a la sección de consejos para campistas.
* Visualización de campings sin coordenadas en la sección inferior.

**Registro e Inicio de Sesión**

Para guardar campings en favoritos, el usuario debe registrarse o iniciar sesión. Una vez autenticado, el usuario obtiene acceso completo a todas las funcionalidades presentes y futuras de la web.

**Funcionalidades con Sesión Iniciada**

* Agregar a Favoritos, visible en la información de cada camping.
* Gestionar Favoritos: Acceso a la página de favoritos donde se visualizan todos los campings guardados.
* Eliminar Favoritos: Opción de quitar campings de la lista de favoritos con un solo clic.
* Notificaciones: Feedback visual inmediato de todas las acciones realizadas.

**Cierre de Sesión**

El usuario puede cerrar sesión en cualquier momento mediante el botón ubicado en la esquina superior derecha, volviendo al modo de navegación sin registro.

# Pruebas

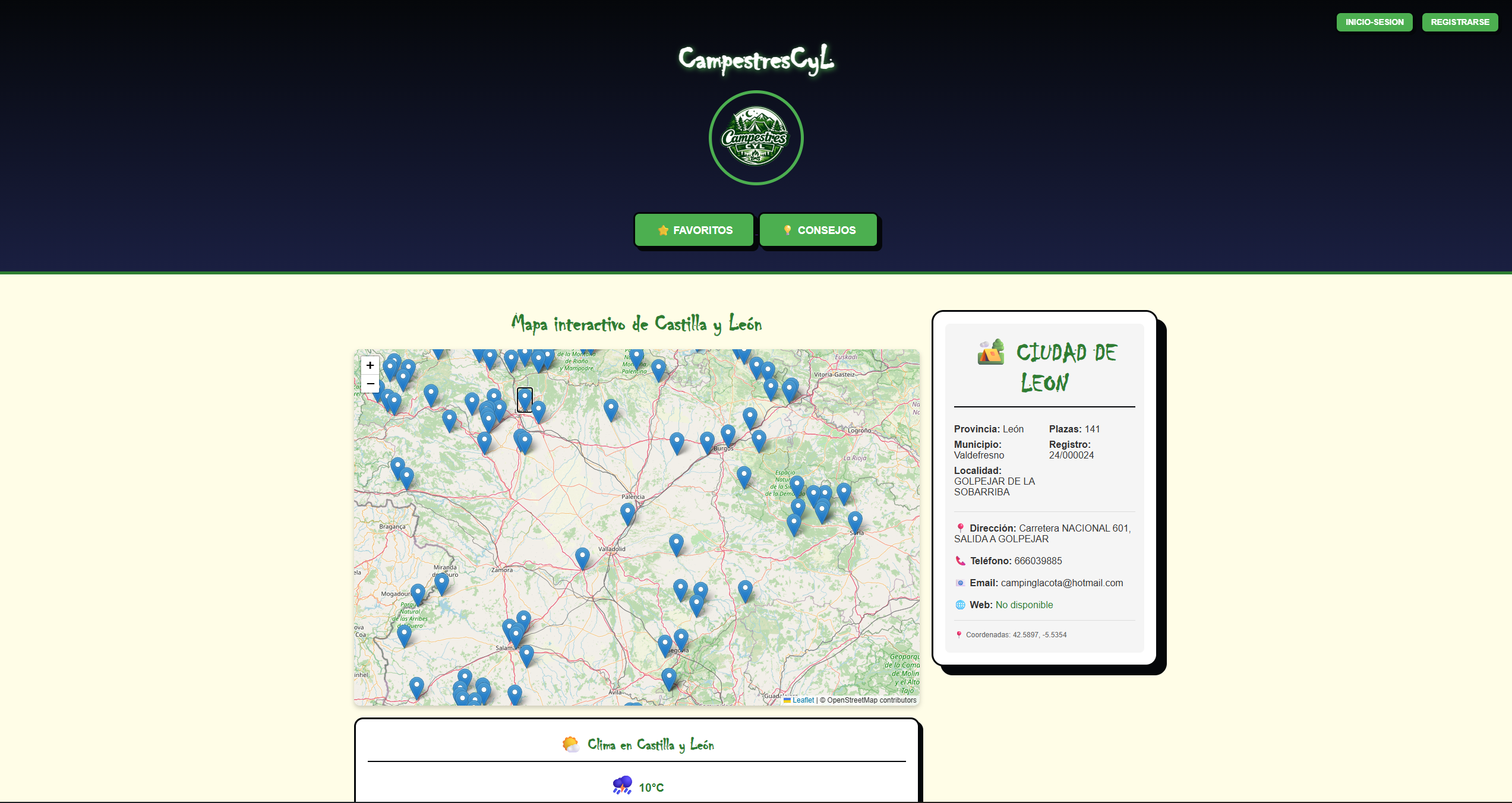
Documentar una breve prueba de **usabilidad** en la que se encomiende realizar al menos una tarea a un usuario y un análisis automático de **accesibilidad.** Ej. Con WAVE [1].

En esta prueba de usabilidad hemos usado a un usuario de Pruebas (Manuel Cardaba), vamos a valorar que este pueda ver la información de los campings, que pueda registrarse además de iniciar sesión con su cuenta recién registrada y que pueda guardar su camping favorito en el apartado de favoritos.

**Usabilidad**

**Información de los campings**

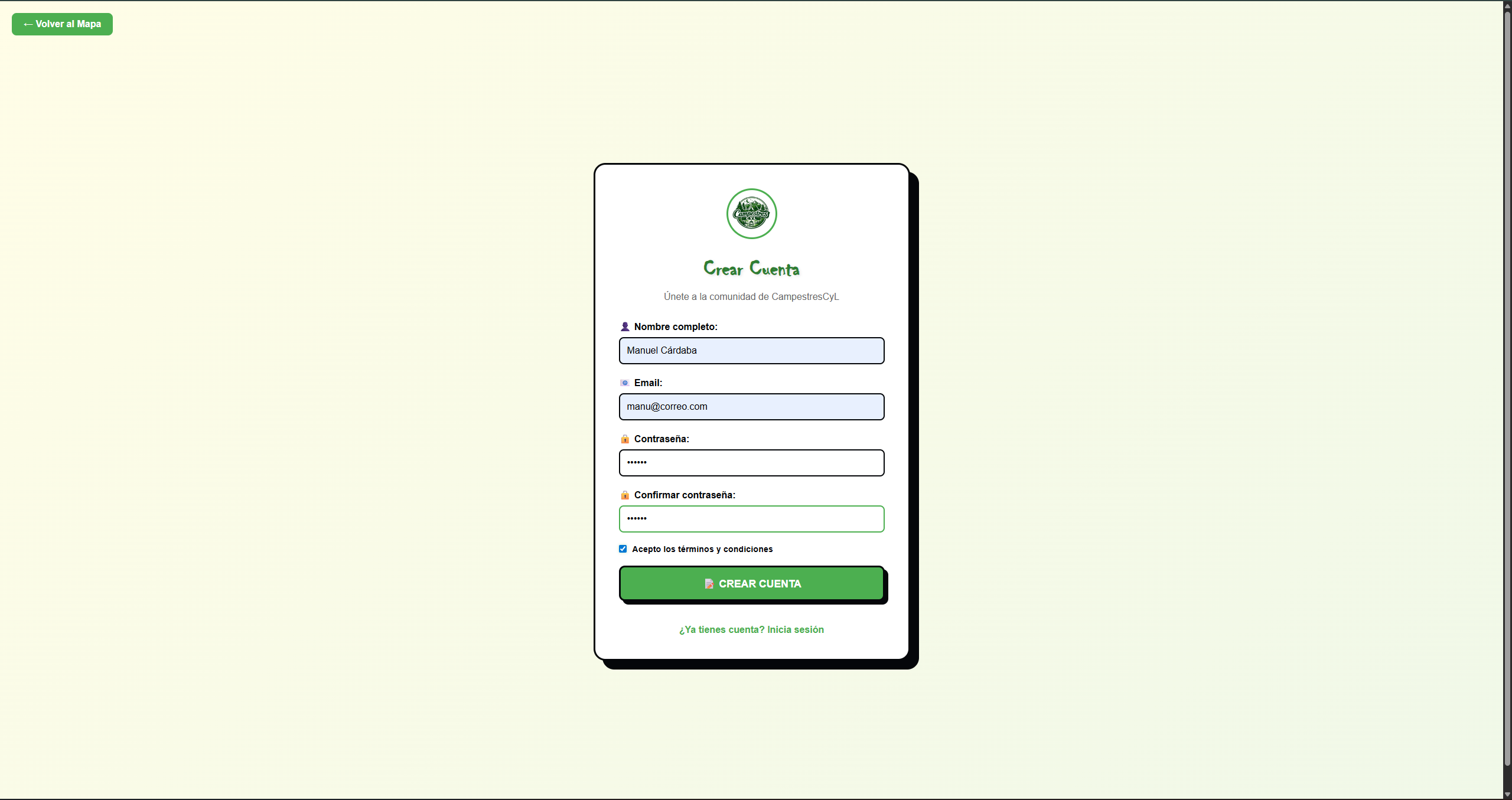
El usuario a la hora de localizar el camping que más le pudiera interesar este ha podido acceder al mapa de manera correcta pudiendo hacer zoom, y pudiendo ver la información de cada camping de manera correcta



**Registro / Inicio de sesión**

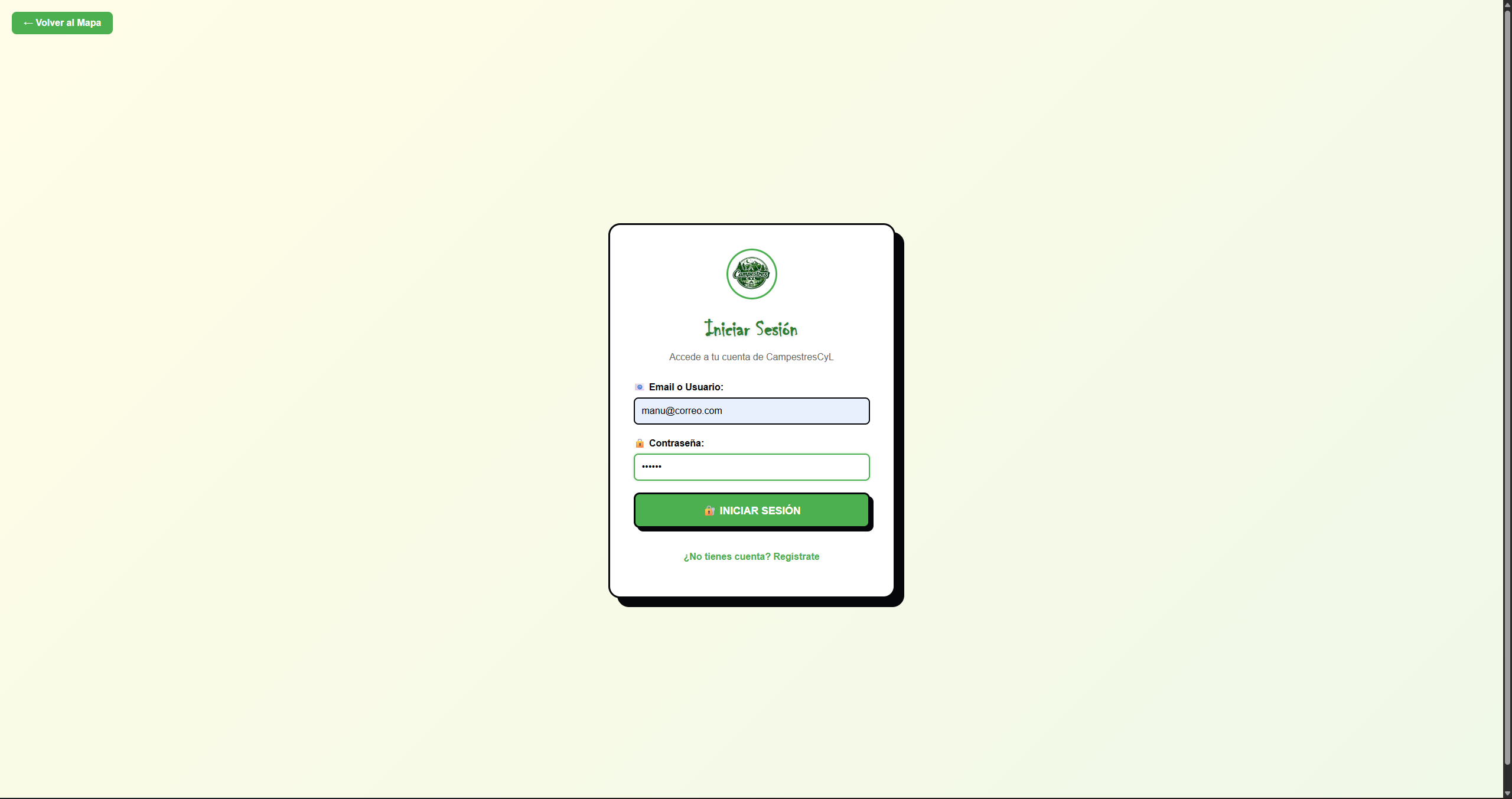
**Registro**

EL usuario ha podido registrase sin ninguna complicación en ninguna de las secciones.



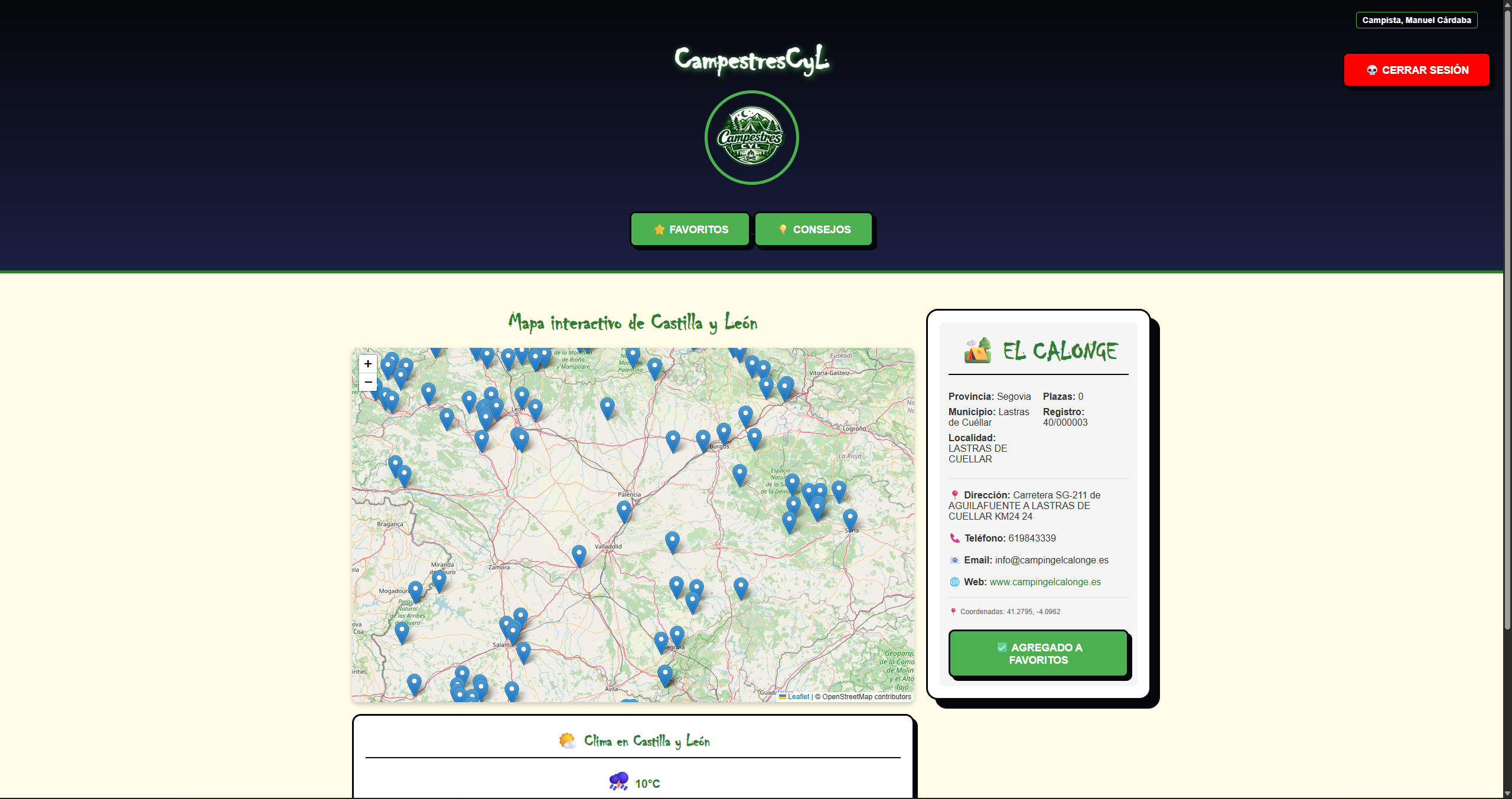
**Inicio**

El usuario ha podido usar la información que a registrado antes sin ninguna complicación y pudiendo acceder a las caracteristicas a las que solo puede acceder mediante el inicio de sesión la cual seria el poder registrar los campings a favoritos.

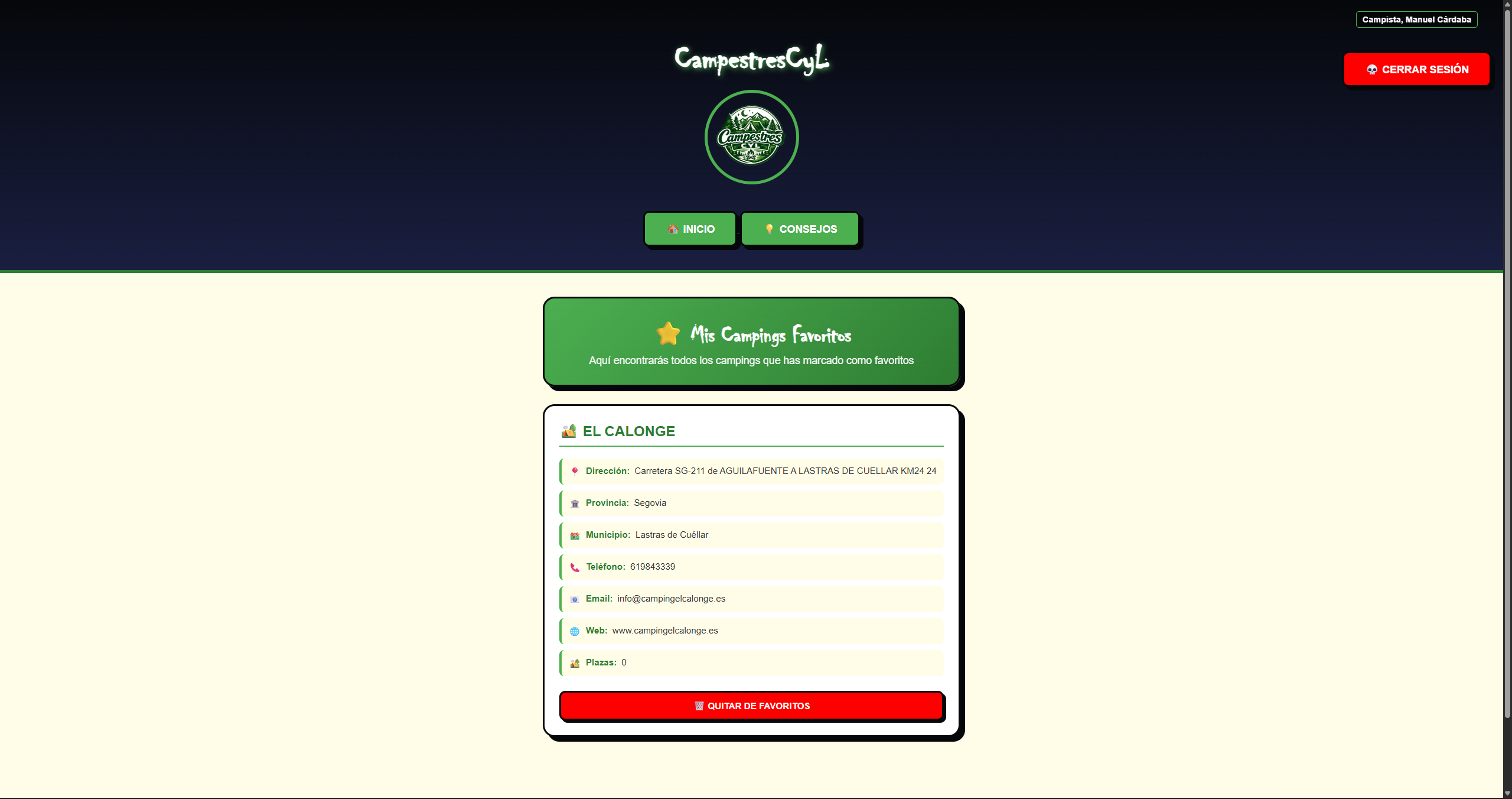


Favoritos

Al usuario se le han podido desbloquear la característica de inicio de sesión la cual es la de poder añadir a favoritos los diferentes campos como se ve en la información del camping este ha podido guardarse a favoritos si ningún problema.

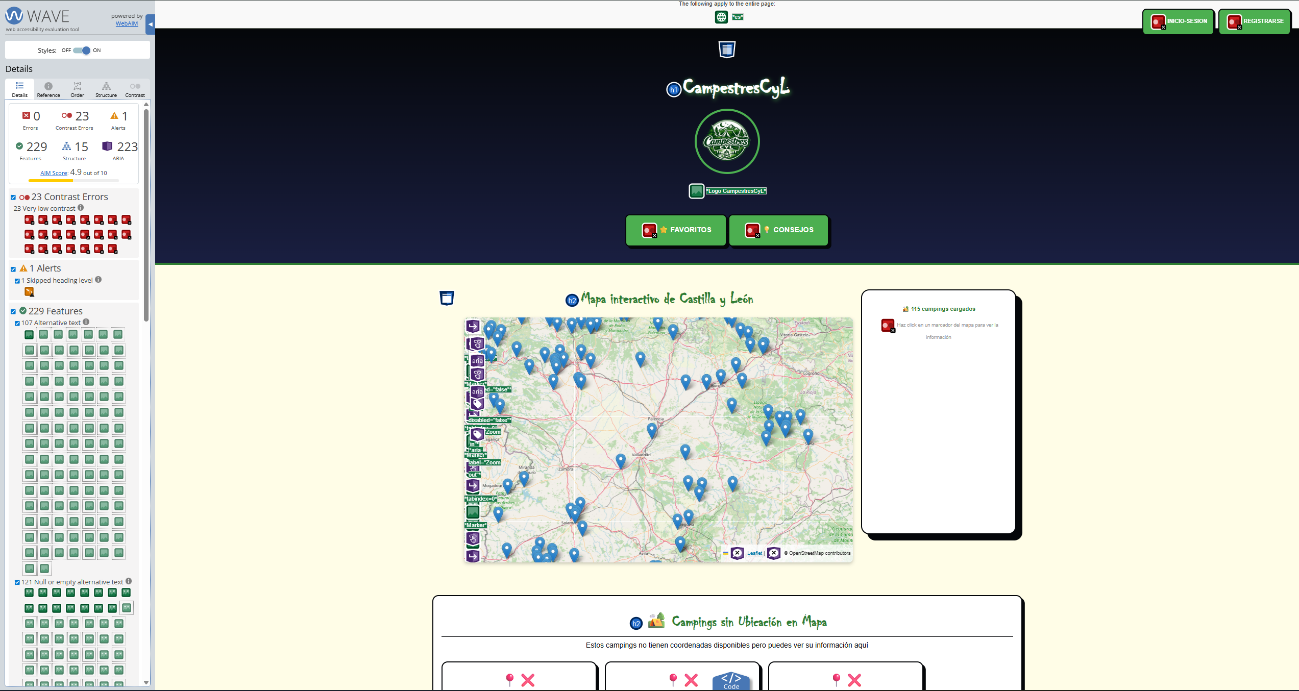


Como se puede observar la información del camping anterior se ha podido guardar de manera correcta en el apartado de favoritos.



**Accesibilidad**

El análisis automático de accesibilidad similar al proporcionado por WAVE muestra que la página presenta un uso de encabezados y posibles elementos interactivos sin etiquetas accesibles; también se identifican numerosas características positivas como la presencia de atributos alternativos, elementos estructurales reconocibles y componentes interactivos detectables, por lo que sería recomendable mejorar el contraste ademas de que abría que solucionar problemas con el header ya que aun faltarían opciones futuras a implementar.



# Despliegue

* **Instrucciones de instalación en local:**

1. Requisitos previos: XAMPP(PHP 7.4+, MySql, Apache), Git y Composer.
2. Una vez instalado se debe descargar el repositorio de GitHub en la URL proporcionada o utilizando el comando **git clone <enlace>** en el directorio htdocs, ubicado dentro del directorio de XAMPP.
3. Una vez tengamos instalado todo, utilizamos el comando **composer install** para instalarnos el autoloader generado en la carpeta vendor.
4. Ahora tenemos que acceder al panel de control de xampp que lo podemos encontrar dentro del repositorio de xampp, activamos apache para el despliegue de la web y mysql para la gestión con la base de datos y nos apuntamos el puerto de Apache para integrarlo posteriormente en nuestra URL.
5. Configurar bases de datos: edita el fichero db.php ubicado en src/config con tus credenciales para luego crear la base de datos,
6. Una vez instalado todo, tenemos que importar correctamente la base de datos en phpmyadmin así que nos dirigimos a phpmyadmin con la url [http://localhost:<puerto>/phpmyadmin](http://localhost:%3cpuerto%3e/phpmyadmin). Una vez accedido al panel de control de phpmyadmin, importamos el fichero ubicado en src/config/database\_campestrecyl.sql
7. Abre el panel de control de XAMPP e inicia Apache y MySQL, guarda los puertos o configúralos si es necesario.
8. Finalmente, para cargar los datos de la API y gestionar cuando quieres recibir los datos puedes configurar el archivo cron.php para determinar las veces que se llamará a la API para insertar o actualizar datos. También si no quieres configurar el cron puedes llamar desde la terminal con php cron.php desde el repositorio para llamar a la API de campings.
9. Ahora podemos acceder a la aplicación con <http://localhost/los-showdowns/public/index.php>.

* **URLs:**
  + Del repositorio de Git: <https://github.com/Qu0zer/los-showdowns>

# Sostenibilidad y "Green Coding"

Justificar las decisiones tomadas para reducir el impacto ambiental:

* **Estrategia de Caché:**

El proyecto implementa un sistema de caché en tres niveles: primero, un script CRON sincroniza diariamente (o manualmente mediante terminal) los datos de la API externa de Castilla y León hacia una base de datos MySQL local, evitando que los usuarios consulten directamente la API externa.

El archivo .htaccess configura caché HTTP en el navegador para recursos estáticos (CSS/JS e imágenes durante 1 mes y favicon durante 1 año).

La arquitectura backend garantiza que el frontend siempre lea de la base de datos mediante action=cargarCampings, eliminando completamente las llamadas directas desde JavaScript a API externas.

* **Optimización de recursos:**

El proyecto ha implementado lazy loading en todas las imágenes mediante el atributo loading=”lazy” y carga diferida de scripts Javascript con el atributo defer, lo que permite que las imágenes solo se descarguen cuando el usuario las va a visualizar y que los scripts se descarguen en paralelo sin bloquear el renderizo del HTML.

* **Reflexión:**

El sistema de caché implementado supone un ahorro energético del 99% comparada con consultas directas constantes a la API externa, así evitamos que por ejemplo, 1000 usuarios estén simultáneamente llamando a la API. Con CRON los usuarios leen las consultas de mysql y la petición a la API se delega al manejo realizado por cron.

# Conclusiones

## Autoevaluación

* **Dificultades encontradas:**

A lo largo de la realización de este proyecto nos hemos encontrado con varios problemas:

**Problemas en el manejo de Git y Github:** Manejar Github es simple, siempre y cuando estes solo trabajando. Trabajando en equipo resulta más difícil organizar quién realiza los cambios y nos hubiera gustado aprender más cómo organizar los Branch por separada antes de haber empezado con el proyecto.

**Manejo de datos utilizando la API pública:** Hemos tenido que cambiar varias veces el flujo de cómo se transporta los datos a lo largo de la realización del proyecto. Nos ha costado bastante progresar de manera fluida por culpa de ello, la gestión de datos ha sido complicada y hasta cierto punto estresante.

* **Valoración de la metodología.**

Durante la realización de la aplicación hemos tenido que cambiar constantemente nuestros roles de quién realiza las partes frontend y backend y no ha habido cierta responsabilidad directa en general.

Sin embargo, hemos tenido una buena coordinación y delegado correctamente ciertos aspectos del trabajo quitando la parte de la aplicación. Nunca ha habido descontento por parte de los dos miembros y se ha podido enfrentar y solucionar los problemas imprevistos.

## Líneas futuras

* **Propuestas de futuro.**

**Añadir más información de la zona**: como bares, restaurantes, puntos de autocaravana, es decir, haber introducido más información sobre la database del registro de turismo para ofrecer mayor accesibilidad a dicha información, y no enfocarnos exclusivamente en solamente los campings.

**Sistema de valoraciones:** Una buena forma de comercializarlo sería haber integrado un sistema de valoración en el que el usuario sería capaz de ofrecer una puntuación a los campings. Esto permitiría por un lado que el usuario registrado tuviera mayor interacción con la aplicación, por otro lado, permite a los usuarios no registrados y en general de un feedback general del tipo de campings que podrían añadir a favoritos.

# Bibliografía

[1] WebAIM, «WAVE web accessibility evaluation tool,» 2026. [En línea]. Available: <https://wave.webaim.org/>. [Último acceso: enero 2026].

[2] Junta de Castilla y León, «Registro de Turismo de Castilla y León,» 2026. [En línea]. Available: <https://analisis.datosabiertos.jcyl.es/>. [Último acceso: enero 2026].

[3] Leaflet, «Leaflet — an open-source JavaScript library for interactive maps,» 2026. [En línea]. Available: <https://leafletjs.com/>. [Último acceso: enero 2026].

[4] draw.io, «UML use case diagrams and ER diagrams with draw.io,» 2026. [En línea]. Available: <https://drawio-app.com/>. [Último acceso: enero 2026].

[5] Windsurf, «Kiro AI — AI-powered IDE and development assistant,» 2026. [En línea]. Available: https://codeium.com/windsurf. [Último acceso: febrero 2026].