# Agenda Escolar

ÍNDICE

1. Documento Descripcion del proyecto
   1. Ámbito y entorno
   2. Análisis de la realidad
   3. Solución y justificación de la solución
   4. Destinatarios
2. Objetivo del proyecto
3. Objetivo del proyecto en lengua extranjera
4. Requisitos
   1. Requisitos de aplicación
   2. Requisitos de la interfaz
      1. Pantalla Inicio
      2. Pantalla Secretaria
      3. Pantalla Formación Profesional
      4. Pantalla Usuarios
      5. Pantalla Agenda
   3. Requisitos funcionales (RF)
   4. Requisitos no funcionales (RNF)
   5. Lista de tareas
5. Planificación del proyecto
   1. Metodologia de trabajo
   2. Diagrama de Gantt
   3. Presupuesto
   4. Precio técnico y de venta
   5. Análisis de Riesgo
6. Análisis y diseño de la base de datos
   1. Esquema Entidad-Relación
   2. Casos de Uso
   3. Esquema Relacional
   4. Script de la base de datos
   5. Diagrama de Clases
7. Implementación y pruebas
   1. Pruebas Funcionales
   2. Manual de Instalación
   3. Manual de Usuario
   4. Resultados obtenidos y conclusiones
   5. Agradecimientos
8. Anexos
   1. Presupuesto
   2. Contrato
   3. Documento de pruebas funcionales
   4. Manual de usuario

# Documento Descripcion del proyecto

## Ámbito y entorno

Este proyecto está orientado a la creación de un web para un centro educativo, en el cual habra una agenda virtual para alumnos, profesores y padres en la cual podre crear actividades o eventos fuera y dentro del horario escolar. Además, de poder gestionar usuarios y formaciones que imparte el centro.

## Análisis de la realidad

Actualmente, en la mayoría de las webs de centros educativos no disponen de un agenda virtual en la que el propio usuario pueda crear eventos y unirse a ellos.

El objetivo es desarrollar una aplicación web para la creación de eventos un centro educativo. Son muchas las actividades y eventos que se realizan en un centro educativo (cenas empresa, charlas, jornadas, lotería, evaluaciones, cursos piee...) por ello se requiere una aplicación que permita su gestión centralizada.

## Solución y justificación de la solución

La aplicación permitirá dar de alta, baja, modificación de eventos dirigidos a 3 roles diferentes: alumnos, profesores y padres en un calendario. Permitirá añadir noticias sobre eventos. Permitirá inscribirse en el evento lo que conlleva el recibir avisos y notificaciones relacionados con ese evento. Como es obvio permitirá también el darse de baja del evento.

## Destinatarios

Los usuarios destinatarios del proyecto serán todos los alumnos, profesores y padres del AMPA del centro educativo. Cada usuario tendrá su correspondiente nombre de usuario, contraseña y rol. El usuario deberá autenticarse para poder acceder a la aplicación. Los roles se definen a continuación: - Rol de administrador que será el encargado de dar de alta o eliminar a cualquiera de los 3 roles. - Rol de alumno, el cual podrá crear su propio evento. - Rol de profesor, el cual podrá crear su propio evento. - Rol de padre, el cual

podrá crear su propio evento.

# Objetivo del proyecto

El objetivo del proyecto es desarrollar una aplicación web que gestione las actividades y eventos de un centro educativo mediante un calendario. El usuario estará informado de las ultimas noticias referiras al centro. La aplicación está orientada a ser una extensión o agregación de una página web ya creada y estructurada.

El acceso a la agenda sera solamente para alumnos matriculados en el centro, profesores del centro y padres inscritos en el AMPA.

# Objetivo del proyecto en ingles

The objective of the project is to develop a web application that manages the activities and events of an educational centre by means of a calendar. The user will be informed of the latest news concerning the centre. The application is oriented to be an extension or aggregation of an already created and structured web page.

The access to the agenda will be only for students enrolled in the centre, teachers of the centre and parents registered in the AMPA.

# Requisitos

## Requisitos de aplicación

Se trabajará con el entorno de PhpStorm, el framework de Laravel, los lenguajes de programación PHP y JavaScript, y el lenguaje de marcas HTML con Bootstrap.

## Requisitos de la interfaz

El cliente interactuara con la aplicación a través de las opciones del menu.

### Pantalla Inicio

Se mostrará información y descripción del centro.

### Pantalla Secretaria

Se mostrará un formulario de contacto y ubicación del centro.

### Pantalla Formación Profesional

Se mostrará una lista con todos los grados que imparte el centro.

### Pantalla Usuarios

Se mostrará una lista con todos los usuarios registrados en el centro. Esta opción es solo para el administrador.

### Pantalla Agenda

Se mostrará un calendario donde el usuario deberá elegir la fecha en la que crear el evento. Esta opción solo se mostrará para los usuarios logeados.

## Requisitos funcionales (RF)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| RF1: Identificación del usuario administrador Prioridad | | | |  | | | |
| RF1.1 El administrador podrá crear, editar y borrar usuarios | | | Importante |
| RF1.2 El administrador podrá crear, editar y borrar eventos del calendario | | | Importante |
| RF1.3 El administrador podrá crear, editar y borrar cursos | | | Esencial |
| RF2: Identificación del usuario registrado Prioridad | |  | | | | | |
| RF2.1 El usuario podrá crear eventos en la agenda | Esencial |
| RF2.2 El usuario podrá unirse a uno o varios eventos | Opcional |
| RF3: Requisitos agenda Prioridad | | | | | |  | |
| RF3.1 El usuario podrá indicar el titulo, descripción y fecha en la que se realizará el evento | | | | | Esencial |
| RF4: Requisitos usuario NO registrado Prioridad | | | | | | | |
| RF4.1 El usuario sola podra ver las paginas de Inicio, Secretaria, Cursos y Login | | | | | | | Importante |
| RF4.2 El usuario que NO sea alumno matriculado, profesor del centro o padre del AMPA no podra logearse | | | | | | | Importnte |
| RF4.3 El usuario que NO sea alumno matriculado, profesor del centro o padre del AMPA no podra ver la agenda | | | | | | | Importnte |

* 1. Requisitos no funcionales (RF)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| RNF1: Accesibilidad Prioridad | | | | | | | | | |
| RNF1.1 El sistema debe resultar fácil de utilizar y de fácil navegación para cualquier tipo de usuario, independientemente de sus habilidades, características o conocimientos personales | | | | | | | | | Esencial |
| RNF2: Idioma Prioridad | |  | | | | | | | |
| RNF2.1 La aplicación utilizará el idioma español | Esencial |
| RNF3: Usabilidad en distintos navegadores Prioridad | | | | | |  | | | |
| RNF3.1 Deberá ser compatible con las últimas versiones de los distintos navegadores | | | Esencial | | |
| RNF4: Interfaz adaptable Prioridad | | | | | | |  | | |
| RNF4.1 El diseño de la interfaz de usuario debe adaptarse a cualquier medida de pantalla | | | | Esencial | | |
| RNF5: Almacenamiento Prioridad | | | | | | | |  | |
| RNF5.1 Todos los datos de la aplicación serán almacenados en una base de datos relacional | | | | | Importante | | |

|  |  |  |  |
| --- | --- | --- | --- |
| RNF6: Acceso óptimo | Prioridad | | |
| RNF6.1 El sistema debe acceder de forma rápida a toda la información que quiera consultar el usuario | | Importante | |
| RNF7: Codificación UTF-8 | Prioridad | |  |
| RNF7.1 la aplicación estará codificada en UTF-8 para poder representar cualquier carácter UNICODE | Esencial | |

* 1. Lista de tareas

|  |  |  |
| --- | --- | --- |
| Código | Descrición | Duración en horas |
| T1 | Especificación de los casos de uso | 26 |
| T1.1 | Funcionalidad del administrador para gestionar la web | 5 |
| T1.2 | Caso de uso del administrador para gestionar usuarios | 8 |
| T1.3 | Caso de uso del administrador para gestionar cursos | 5 |
| T1.4 | Caso de uso del administrador para gestionar eventos | 8 |
| T2 | Análisis y diseño de la base de datos | 6 |
| T2.1 | Esquema Entidad-Relación | 1 |
| T2.2 | Casos de Uso | 1 |
| T2.3 | Esquema Relacional | 1 |
| T2.4 | Script de la base de datos | 1 |
| T2.5 | Diagrama de Clases | 1 |
| T2.6 | inserción de datos | 1 |
| T3 | Diseño interfaz gráfica | 16 |
| T3.1 | Crear interfaz gráfica pantalla Inicio | 2 |
| T3.2 | Crear interfaz gráfica pantalla Secretaria | 2 |
| T3.3 | Crear interfaz gráfica pantalla Cursos | 3 |
| T3.4 | Crear interfaz gráfica pantalla Usuarios | 3 |
| T3.5 | Crear interfaz gráfica pantalla Agenda | 5 |
| T3.6 | Crear interfaz gráfica pantalla Registro | 2 |
| T4 | Implementación de la aplicación | 28 |
| T4.1 | Desarrollo método para crear, editar, listar y borrar usuarios | 8 |
| T4.2 | Desarrollo método para crear, editar, listar y borrar cursos | 8 |
| T4.3 | Desarrollo método para crear, editar, listar y borrar eventos | 8 |
| T4.4 | Desarrollo método para recibir correos desde el formulario de contactos | 4 |
| T5 | Realización de pruebas | 5 |
| T5.1 | Realizar pruebas individuales por rol de usuarios | 3 |
| T5.2 | Realizar pruebas completas de la aplicacion desplegada | 2 |
| Total de horas |  | 81 |

# Planificación del proyecto

## Metodologia de trabajo

La metodologia elegida para este proyecto es la metodología tradicional.Se ha hecho énfasis en la planificación total de todo el trabajo a realizar y una vez que está todo detallado, comienza el ciclo de desarrollo del proyecto.

## Diagrama de Gantt

* 1. Planificación temporal de tareas

Dado el tiempo después de la jornada de FCT se estima una hora diaria de lunes a jueves, dos horas los viernes y de tres horas sábado y domingo.

|  |  |
| --- | --- |
| Dias | Horas |
| Lunes - Viernes | 6 |
| Sábado - Domingo | 6 |
| Semanal | 12 |

## Presupuesto

|  |  |
| --- | --- |
| Costes fijos mensuales | Precio € |
| Salario | 384€  (8€/hora) |
| Cuota de autónomos | 50€ |
| Fibra y móvil | 36€ |
| Luz | 50€ |
| Material de oficina | 20€ |
| Total costes fijos/mes | 535€ |

Total de los costes en la finalización de los 3 meses: 1.605€

|  |  |
| --- | --- |
| Costes virtuales Precio € | |
| Dominio + Hosting | 65,45€ |

Total de los costes en la finalización de los 3 meses: 130.90€

|  |  |
| --- | --- |
| Coste Total | Precio € |
| Costes Fijos | 1.605€ |
| Costes Variables | 130.90€ |
| Precio total sin IVA | 1.735,90€ |
| Precio total con IVA | 2.100,44€ |

## Precio técnico y de venta

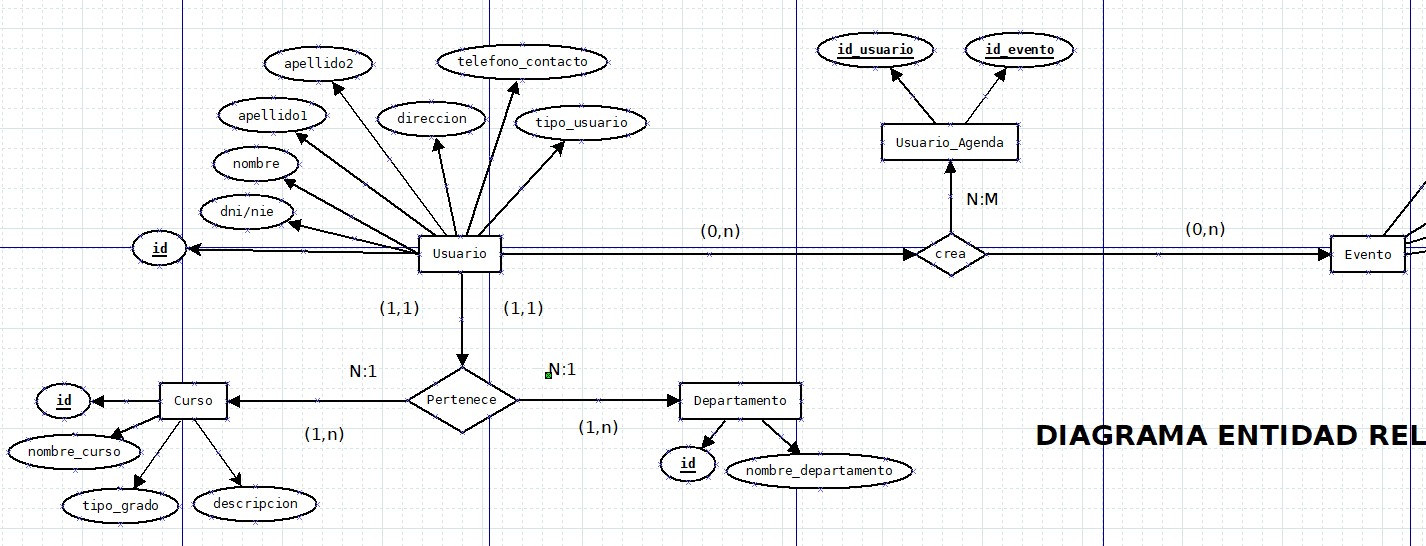
|  |  |  |  |
| --- | --- | --- | --- |
| Precio técnico Precio € | | |  |
| Calculo | | Coste total/unidad |
| Resultado | | 1.735,90/1 =  1.735,90€ |
| Precio técnico total sin IVA | | 1.735,90€ |
| Precio total con IVA | | 2.100,44€ |
| Precio Venta Precio € | | | |
| Calculo | Precio técnico X 1 +(margen 0.85)) | | |
| Resultado | 1.735,90 X (1 + 0.85) = 3.211.41€ | | |
| Precio venta total sin IVA | 3.211.41€ | | |
| Precio total con IVA | 3.885.80€ | | |

* 1. Análisis de riesgo

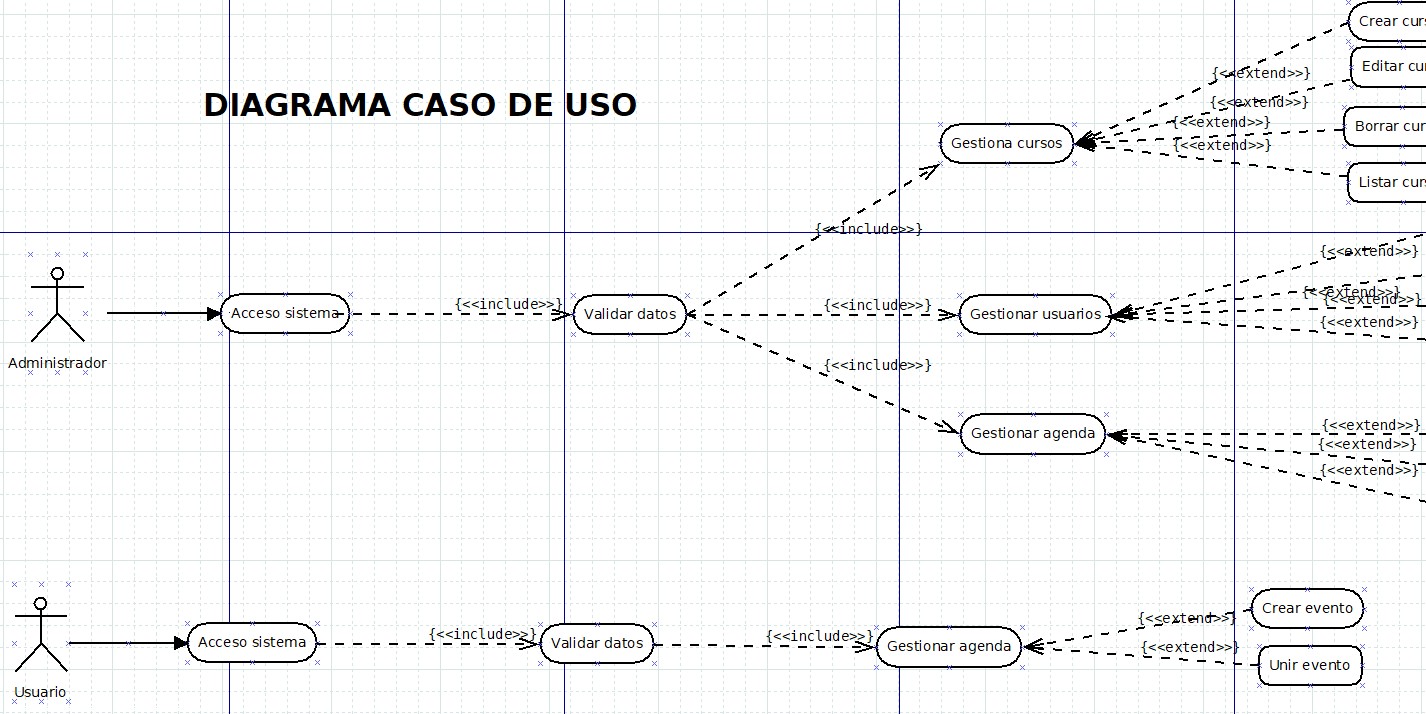
|  |  |  |
| --- | --- | --- |
| Riesgo | Medida | Nivel de peligrosidad |
| Rotura de equipo | Tener otro equipo de emergencia y guardar todo en un disco externo | Grave |
| Perdida de información | Tener una copia de seguridad | Grave |
| Fallo en los requisitos | Realizar una reunión con el cliente para acordar los requisitos | Grave |
| Incompatibilidad de las librerías utilizadas | Utilizar las últimas versiones de las librerías | Moderado |
| Circunstancias personales | Tratar de recuperar las horasen algún momento | Leve |

# Análisis y diseño de la base de datos

## Esquema Entidad-Relación

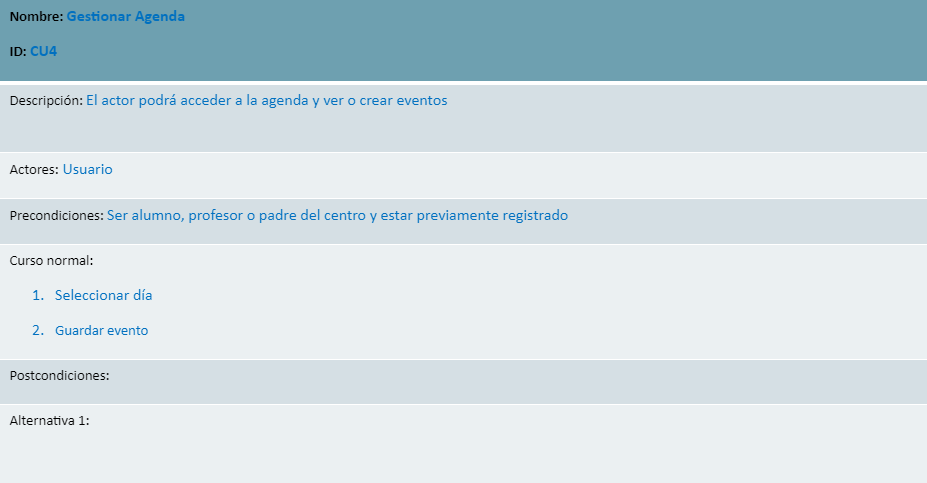


* 1. Casos de Uso

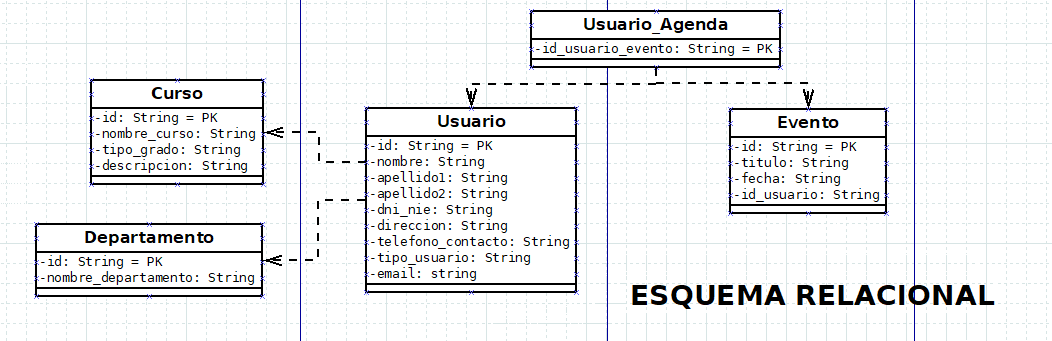








## Esquema Relacional



* 1. Script de la base de datos

CREATE TABLE `cursos` (

`id` bigint UNSIGNED NOT NULL,

`familia\_profesional` varchar(100) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`nombre\_curso` varchar(100) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`tipo\_grado` varchar(255) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`descripcion` varchar(500) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

CREATE TABLE `eventos` (

`id` bigint UNSIGNED NOT NULL,

`title` varchar(255) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`descripcion` text COLLATE utf8mb4\_unicode\_ci NOT NULL,

`start` datetime NOT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

CREATE TABLE `users` (

`id` bigint UNSIGNED NOT NULL,

`name` varchar(255) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`apellido1` varchar(255) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`apellido2` varchar(255) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`dni` varchar(255) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`direccion` varchar(255) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`telefono\_contacto` varchar(255) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`tipo\_usuario` varchar(255) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`email` varchar(255) COLLATE utf8mb4\_unicode\_ci NOT NULL,

`email\_verified\_at` timestamp NULL DEFAULT NULL,

`password` varchar(255) COLLATE utf8mb4\_unicode\_ci NOT NULL,

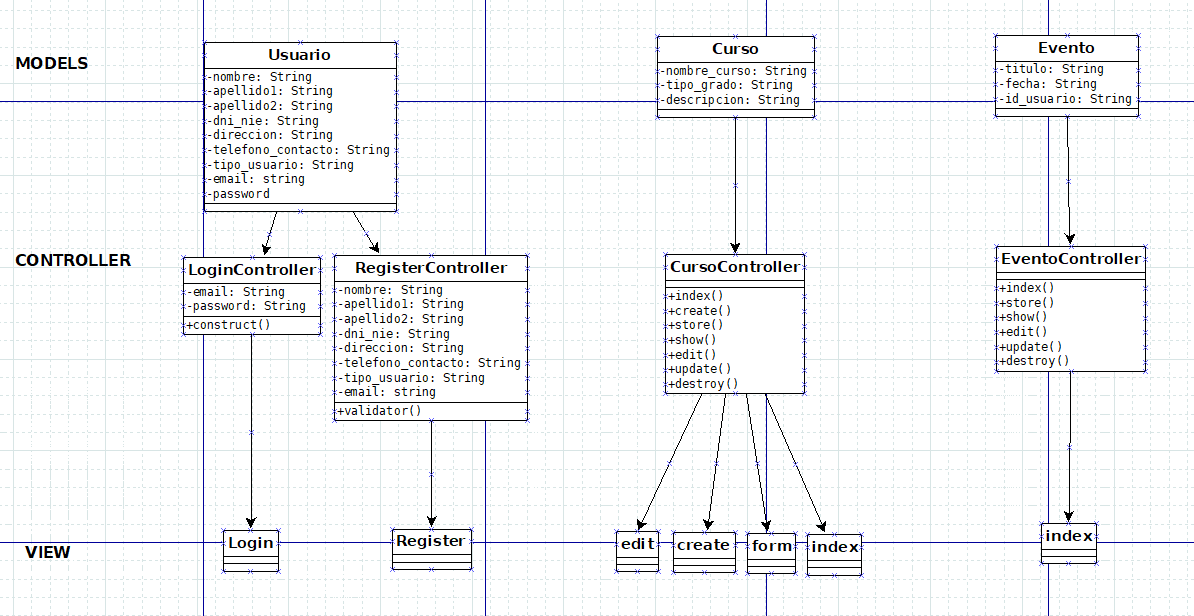
`remember\_token` varchar(100) COLLATE utf8mb4\_unicode\_ci DEFAULT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL

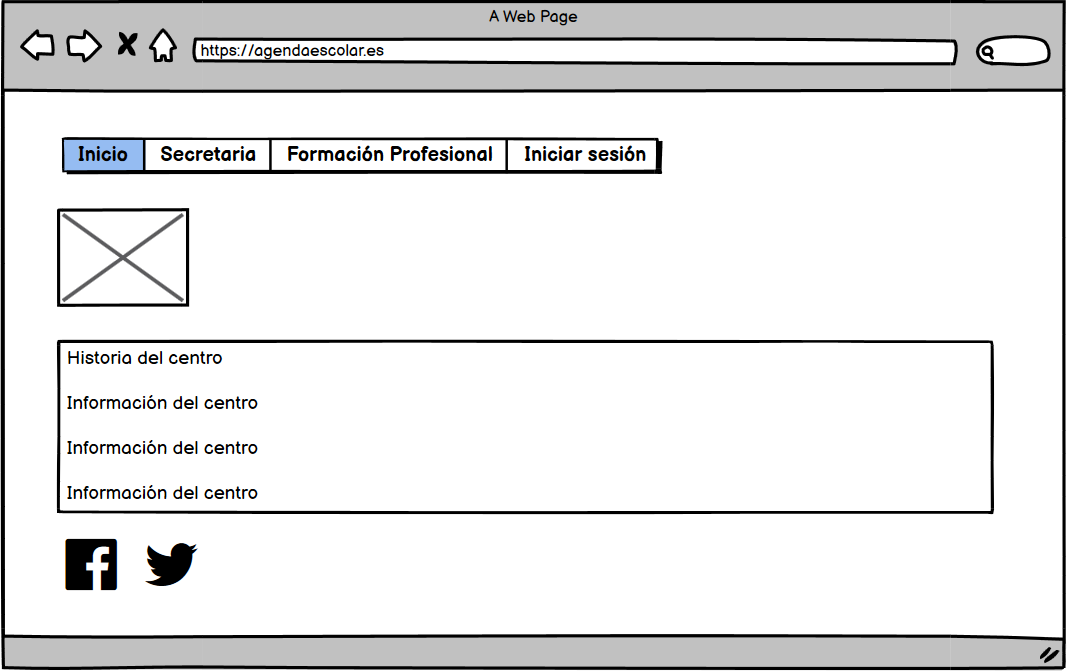
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

## Diagrama de Clases

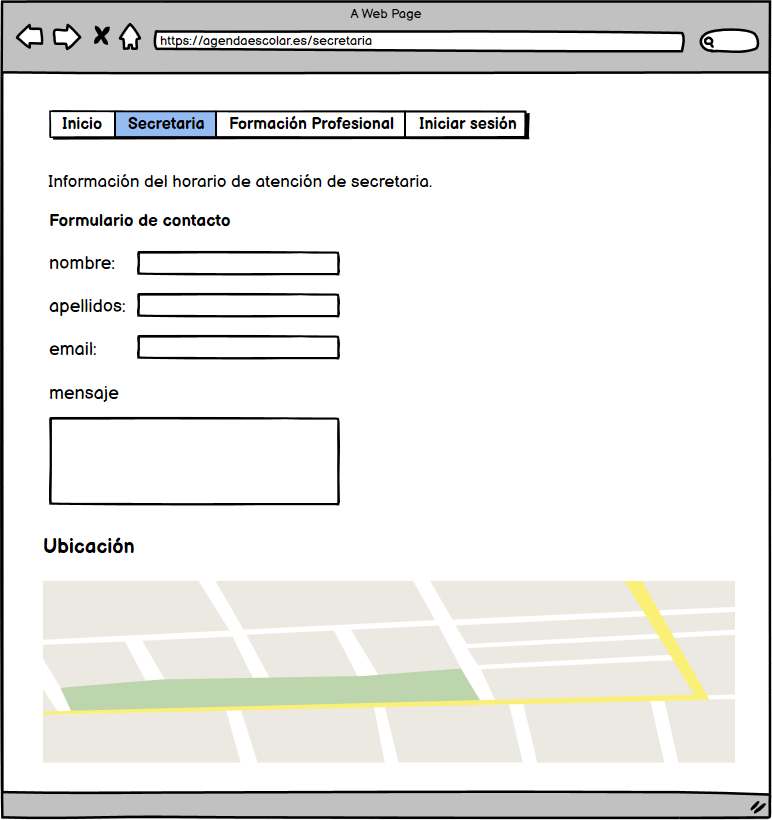


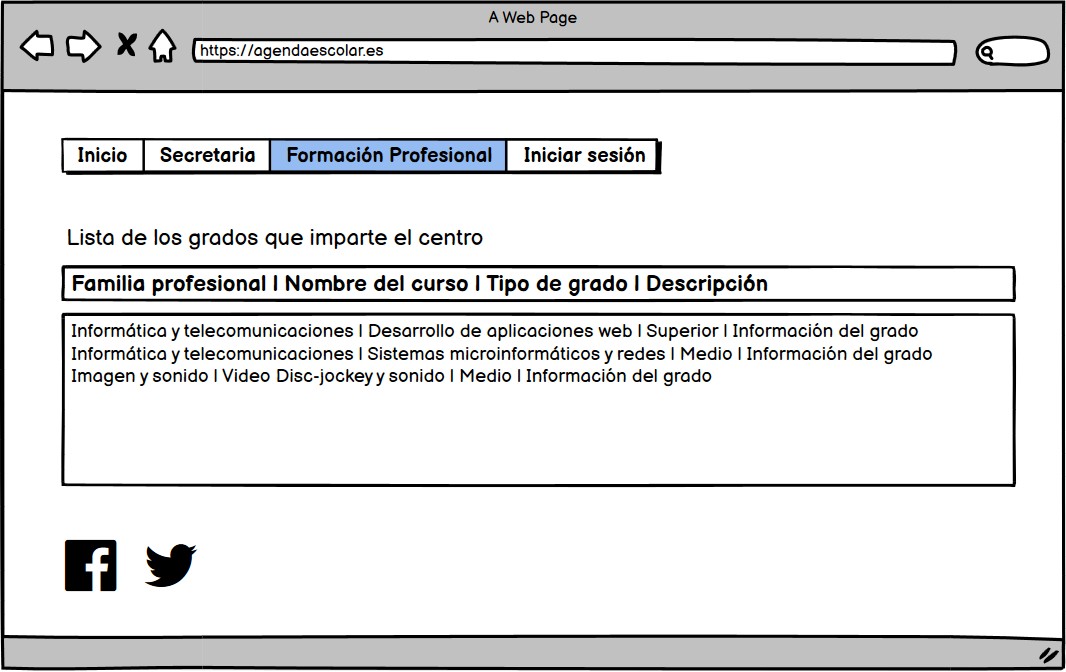
* 1. Diseño de interfaces

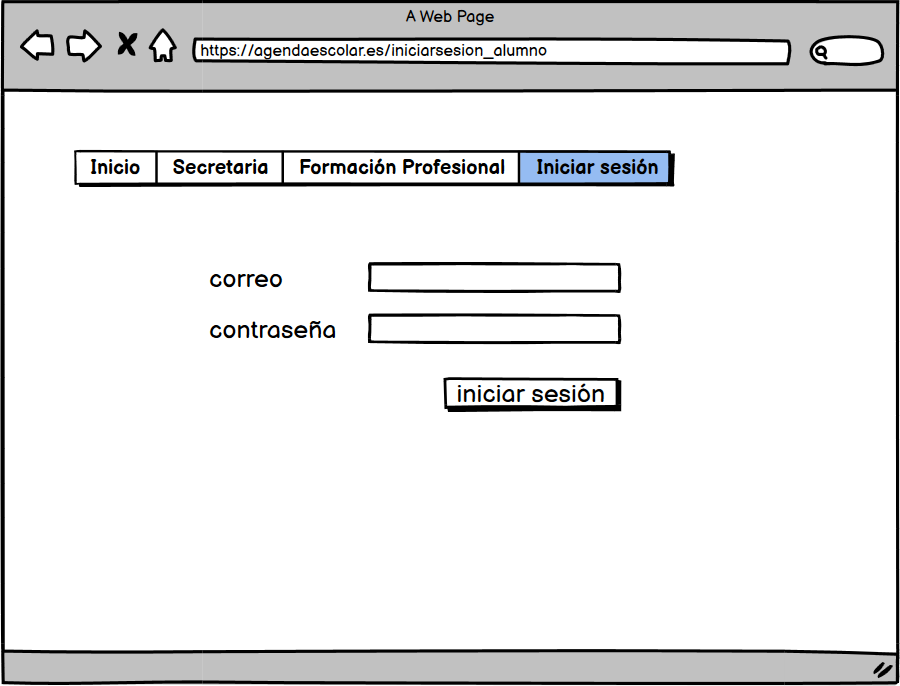
### Interfaz para usuario no registrado

Iniiciio mostrara el logo e información del centro. Tambien se mostraran los botones de redes sociales.

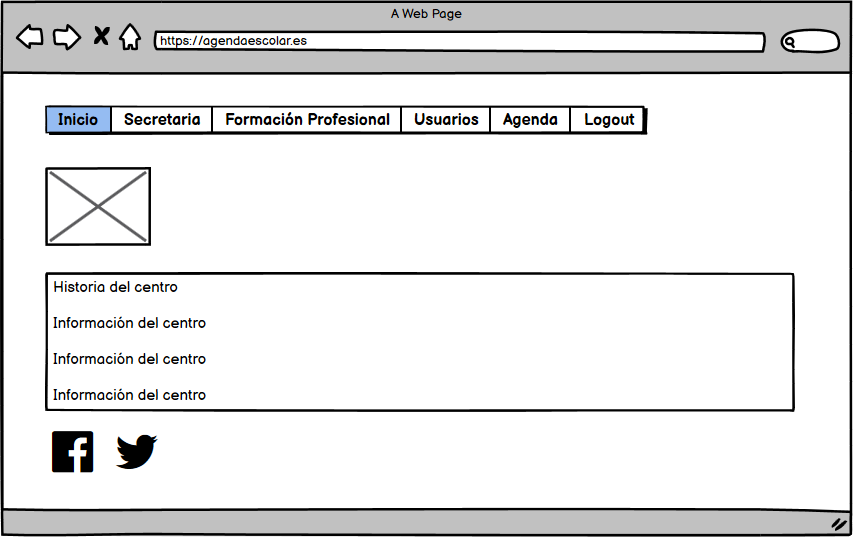
Secretariia mostrara el horario de atención de secretaria, un formulario de contacto y la ubicación del centro.



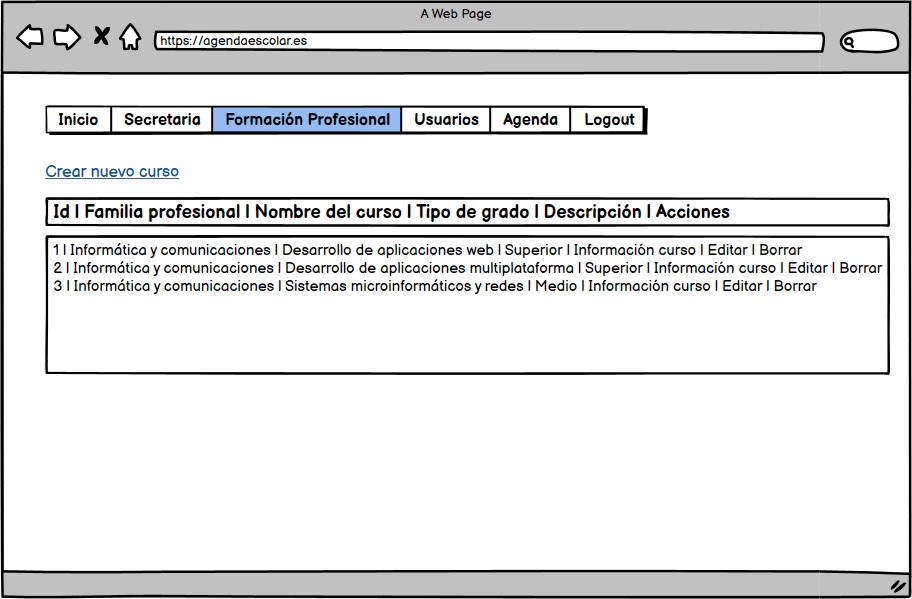
Formaciión Profesiionall mostrara una lista con toda la información sobre todos los grados que imparte el centro.

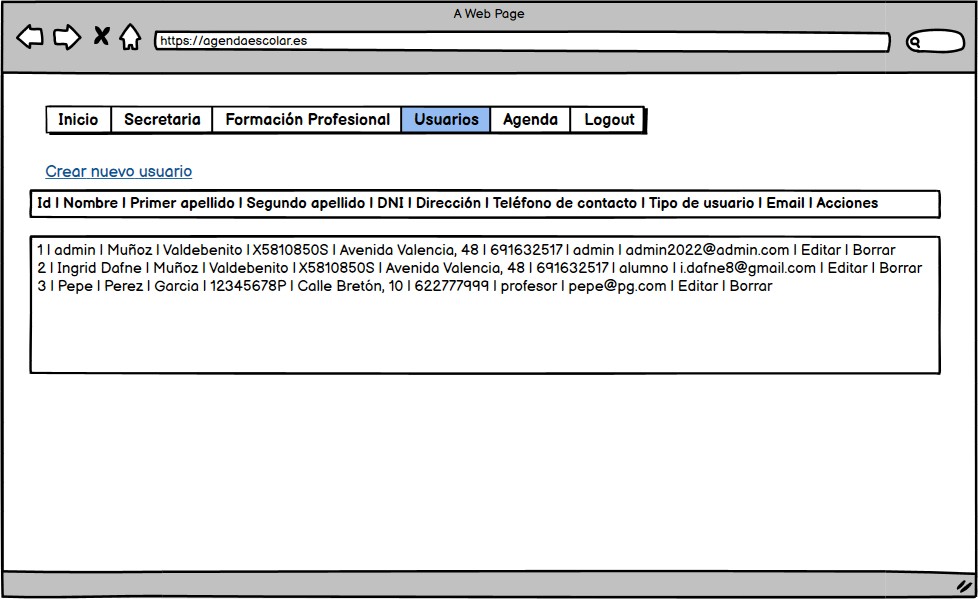
Iniiciiar Sesiión mostrara un inicio de sesión solo para los usuarios del centro.

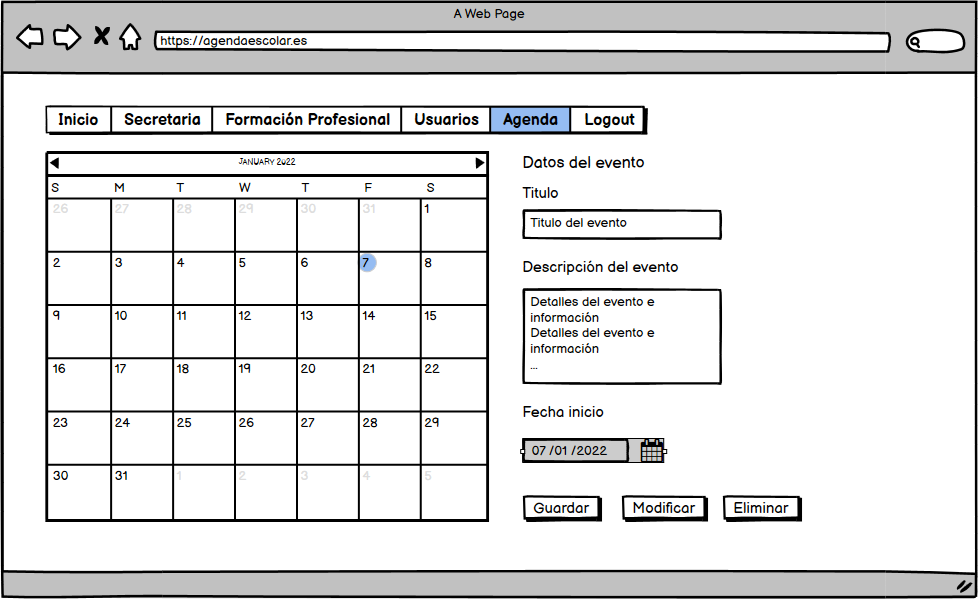
### Interfaz para usuario registrado Admin

Iniiciio mostrara el logo e información del centro. Tambien se mostraran los botones de redes sociales.

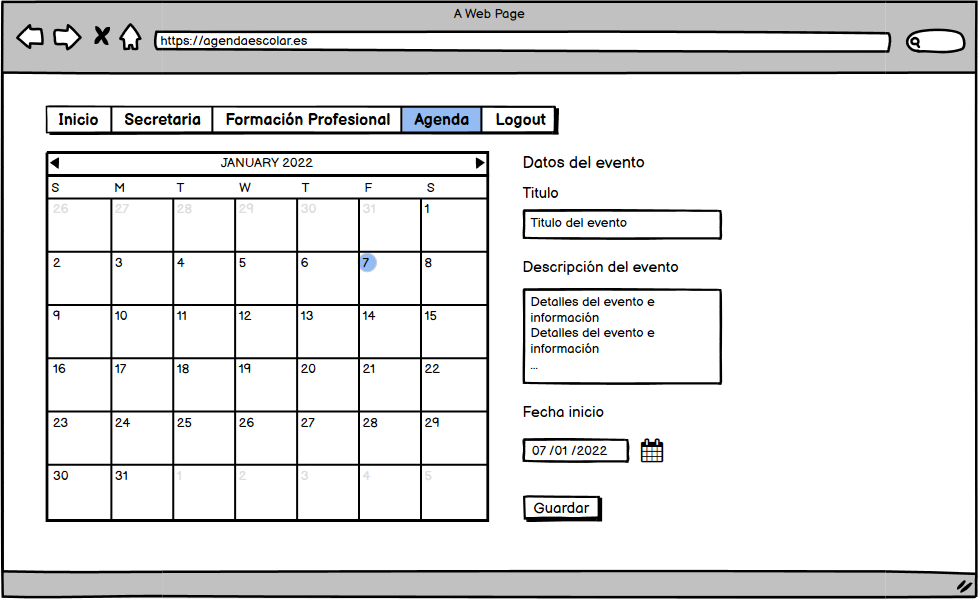
Formaciión Profesiionall mostrara el logo e información del centro. Tambien se mostraran los botones de redes sociales.



Usuariios mostrara el logo e información del centro. Tambien se mostraran los botones de redes sociales.

Agenda mostrara el logo e información del centro. Tambien se mostraran los botones de redes sociales.

### Interfaz para usuario registrado

Iniiciio mostrara el logo e información del centro. Tambien se mostraran los botones de redes sociales.

# Implementación y pruebas

## Pruebas Funcionales

|  |  |
| --- | --- |
| RF1 | El administrador podrá crear,, editar y borrar usuarios |
| Precondición | Estar logeado como Administrador |
| Datos de entrada | Se crearán, editarán o borrarán los usuarios |
| Datos de salida | a). El administrador pulsará sobre la opción que deseé. OK |
| Datos de salida | b). Para crear deberá rellenar todos los datos del formulario. OK |
| Datos de salida | c). Para editar deberá elegir los campos a editar. OK |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| RF1 El administrador podrá crear,, editar y borrar usuarios | | | | | |
| Datos de salida | d). Deberá pulsar el botón de Guardar Datos para que se produzcan los cambios. OK | | | | |
| Datos de salida | e). Para borrar deberá elegir el usuario y pulsar sobre el botón de Borrar. OK | | | | |
| RF2 | El administrador podrá crear,, editar y borrar eventos del calendario | |  | | |
| Precondición | Estar logeado como Administrador | |
| Datos de entrada | Se crearán, editarán o borrarán los eventos | |
| Datos de salida | a). El administrador pulsará sobre la opción que deseé. OK | |
| Datos de salida | b). Para crear deberá rellenar todos los datos del formulario. OK | |
| Datos de salida | c). Para editar deberá elegir los campos a editar. OK | |
| Datos de salida | d). Deberá pulsar el botón de Guardar para que se produzcan los cambios. OK | |
| Datos de salida | e). Para borrar deberá elegir el evento y pulsar sobre el botón de Borrar. OK | |
| RF3 | El administrador puede crear,, editar y borrar cursos | | | |  |
| Precondición | Estar logeado como Administrador | | | |  |
| Datos de entrada | Se crearán, editatán, o borrarán los cursos | | | |  |
| Datos de salida | a). El administrador pulsará sobre la opción que deseé. OK | | | |  |
| Datos de salida | b). Para crear un curso deberá rellenar todos los datos del formulario. OK | | | |  |
| Datos de salida | c). Para editar un curso deberá elegir los campos a editar. OK | | | |  |
| Datos de salida | d). Deberá pulsar el botón de Guardar Dato para que se produzcan los cambios. OK | | | |  |
| Datos de salida | e). Para borrar deberá elegir el curso y pulsar sobre el botón de Borrar. OK | | | |  |
| RF4 | Identificación del usuario registrado | | |  | |
| Precondición | Ser alumno, profesor o padre del centro | | |
| Datos de entrada | El usuario debe estar logeado en la aplicación | | |
| Datos de salida | a). El usuario elige el día en el que crear el evento. OK | | |
| Datos de salida | b). Aparecerá un formulario que deberá rellenar con titulo, descripción y fecha. OK | | |
| Datos de salida | c). Deberá pulsar el botón de Guardar para que se produzcan los cambios. OK | | |
| Datos de salida | e). El evento se mostrará en el calendario. OK | | |
| RF5 | Requisitos agenda |  | | | |
| Precondición | Haber iniciado sesión |
| Datos de entrada | El usuario debe estar logeado en la aplicación |
| Datos de salida | a). El usuario debe ir a la opción del menu Agenda. OK |

* 1. Manual de Instalación

La aplicación se encuentra actualmente en desarrollo. Para el despliegue en local se ha utilizado Docker. El repositorio de este proyecto se encuentra en GitHub: https://github.com/ingrid-munoz/Proyecto

Pasos Installaciión Locall Los requisitos son:

* Servidor web Apache2
* Base de datos MySQL
* PhpStorm 7.3

1º. Instalamos Docker Desktop para Windows desde su web oficial https://[www.docker.com/products/docker-desktop](http://www.docker.com/products/docker-desktop) 2º. Una vez instalado, abrimos una ventana de cmd y escribimos:

Se trabajará sobre la utlima versión de Ubuntu.

* docker pull ubuntu:latest
* docker run --name nombreDelProyecto -it -p 8000:8000 -p 80:80 -v C:\CarpetaDondeSeGuardara:/var/www/html ubuntu:latest bash
* apt-get update
* apt-get install apache2 php libapache2-mod-php vim net-tools

3º. Ahora creamos un nuevo proyecto con Laravel

* cd rutaDelProyecto
* Laravel new NombreDelProyecto
* curl -sS https://getcomposer.org/installer

4º. Abrimos PhpStorm y seleccionamos el nuevo proyecto. Dentro iremos al archivo .env y añadimos:

DB\_CONNECTION=mysql DB\_HOST=127.0.0.1 DB\_PORT=3306

DB\_DATABASE=agenda DB\_USERNAME=root DB\_PASSWORD=root

5º. Creamos los controladores y migraciones

php artisan make:model Nombre -mcr

6º. Una vez que tengamos todo configurado podemos acceder al servidor local escribiendo:

php artisan serve --host 0.0.0.0 &

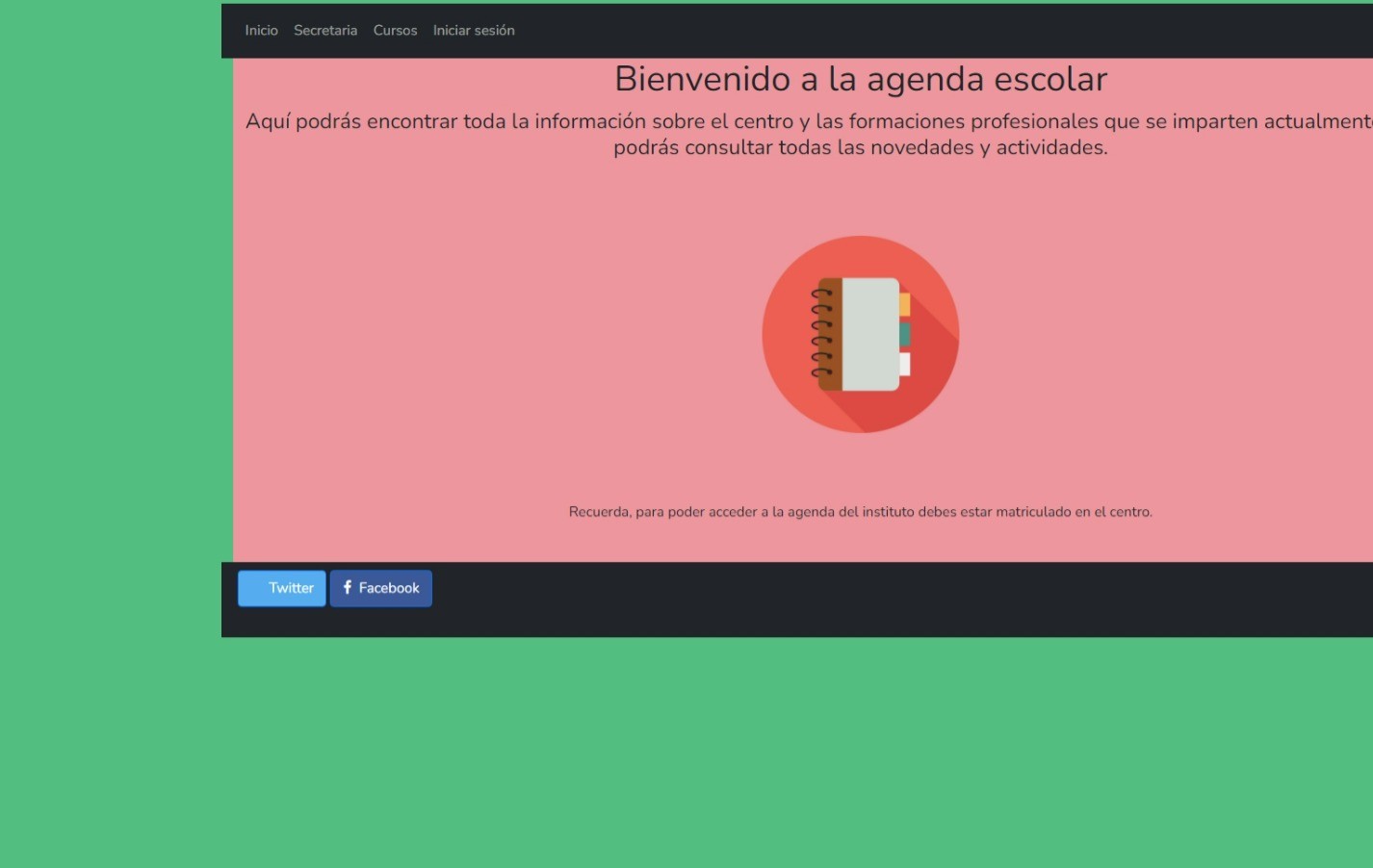
7º. Escribimos en el navegador: localhost:8000

## Manual de Usuario

Usuario no registrado

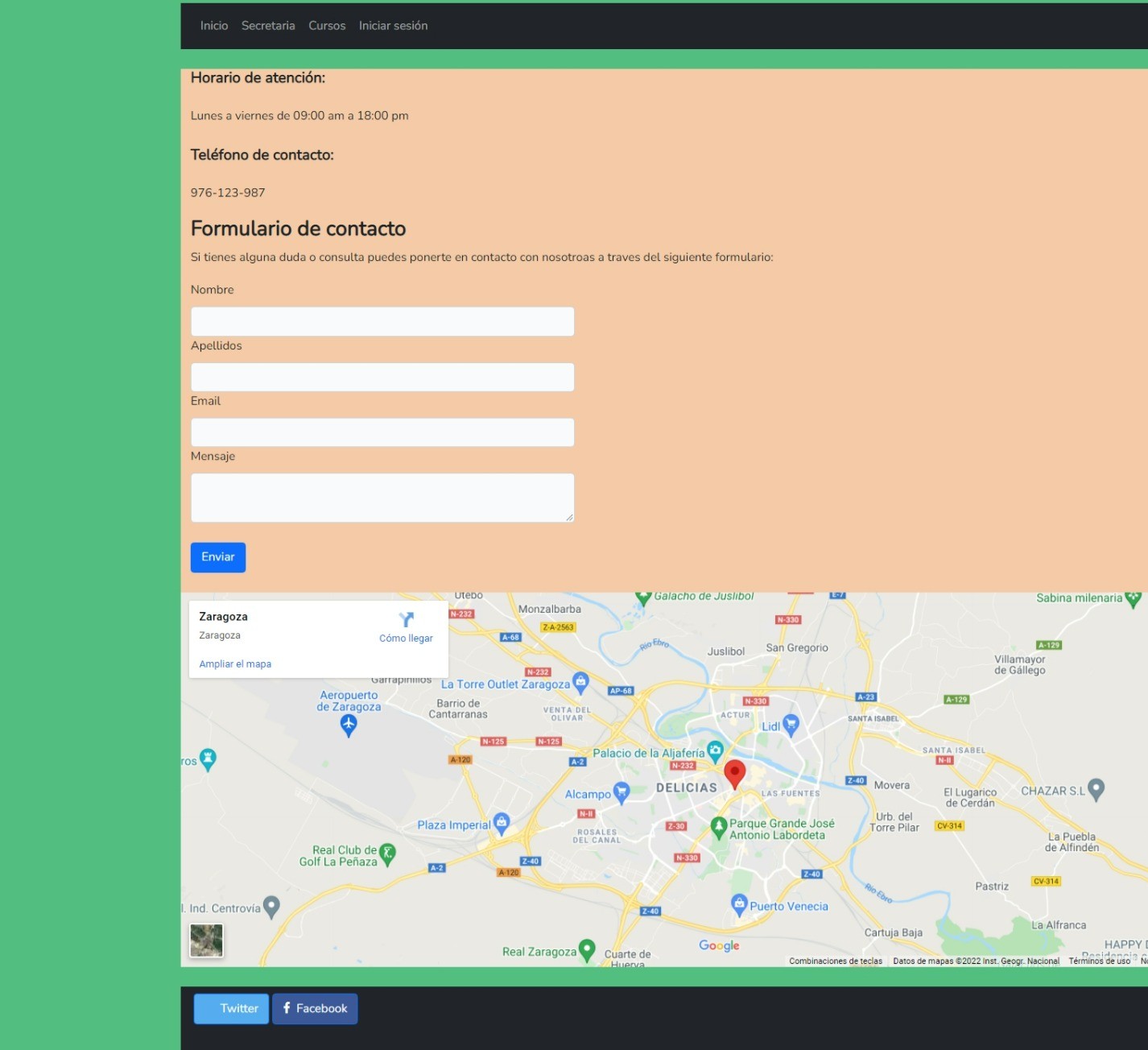
### Inicio

Aquí se dara la bienvenida a la web del centro. Tambien se muestra las opciones de Inicio, Secretaria, Cursos e Iniciar sesión.



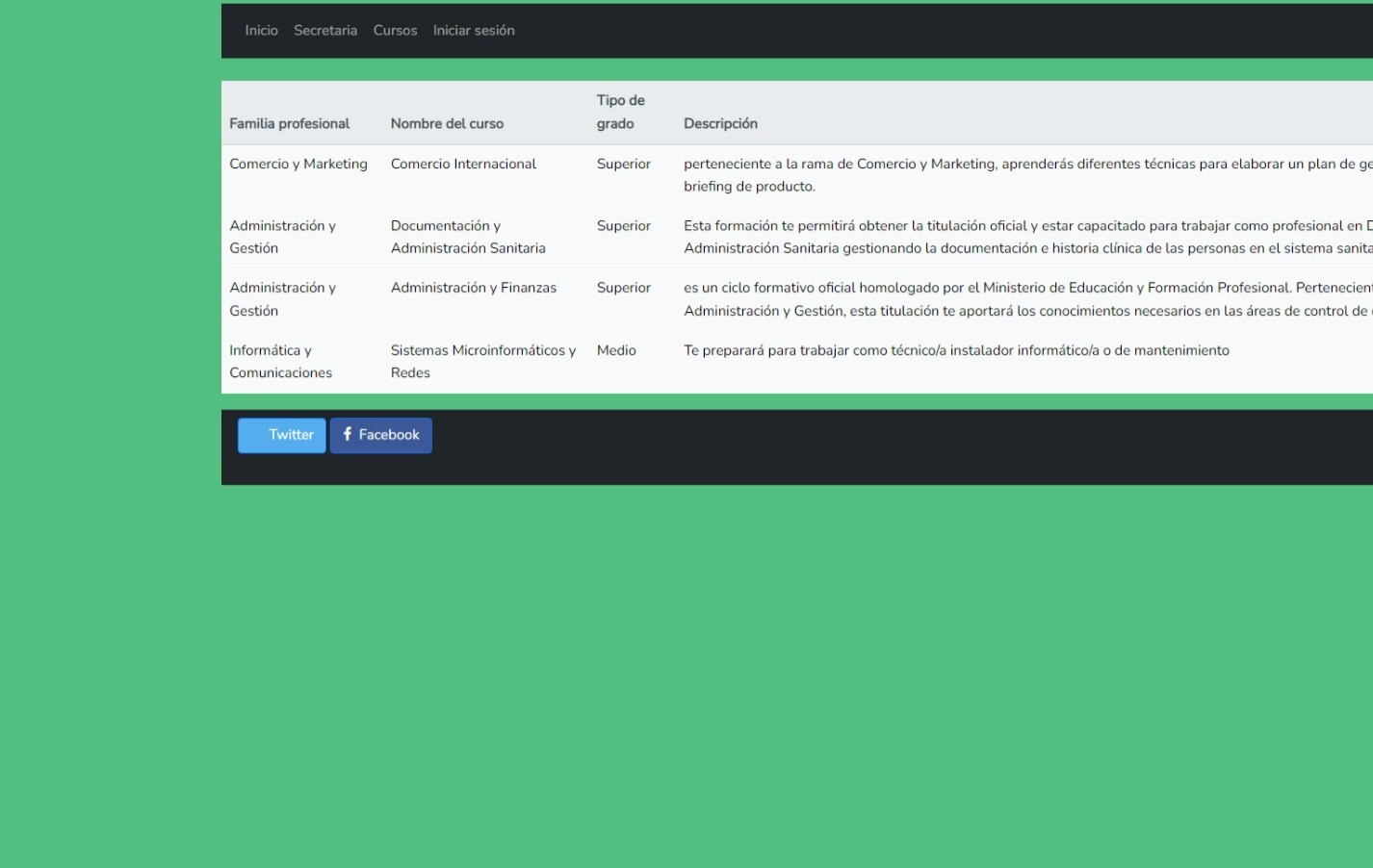
### Secretaria

Aquí encontraremos el horario y telefono de contacto del centro. Ademaá, dispone de un formulario de contacto y un mapa con la ubicación.



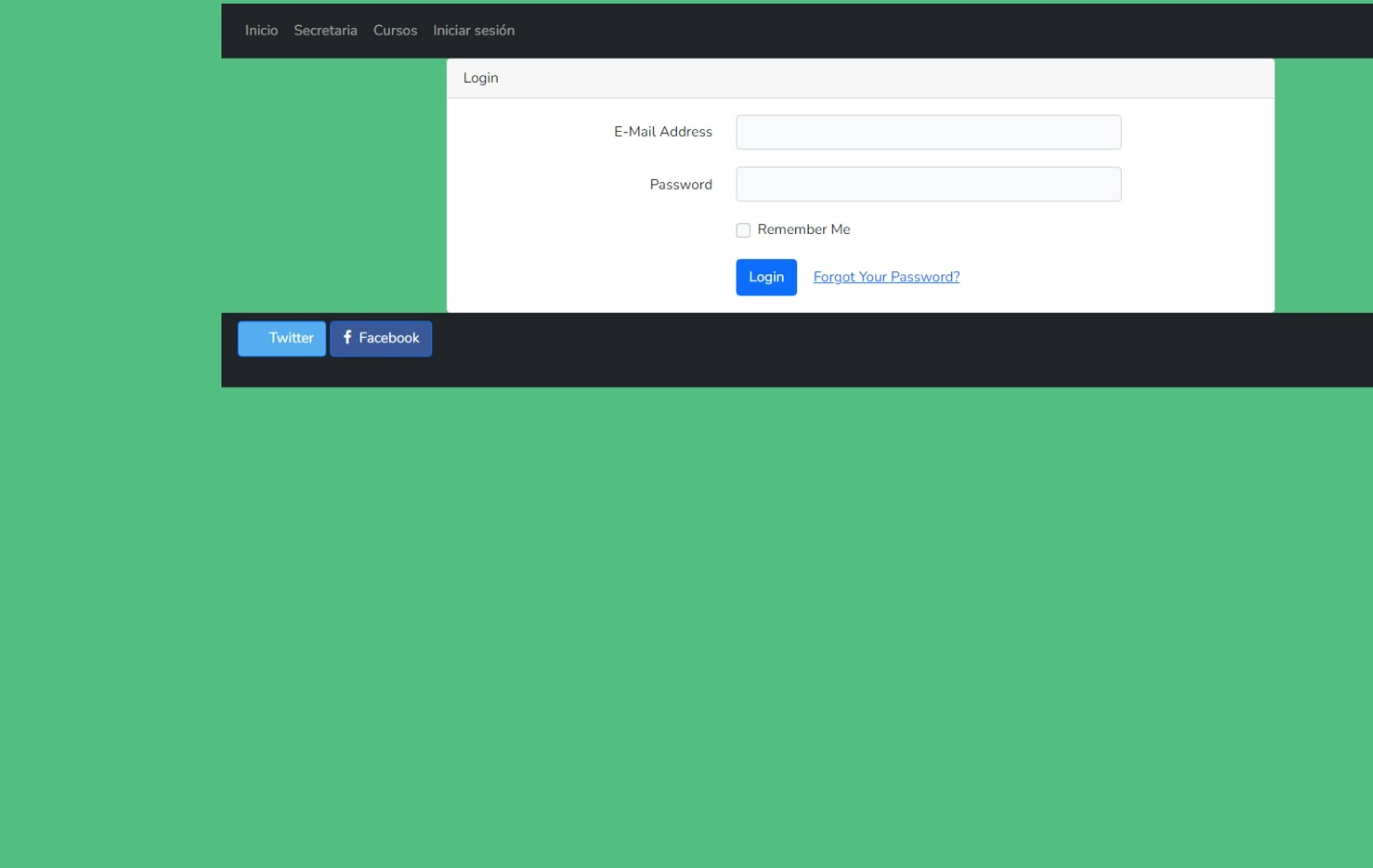
### Cursos

Aquí encontraremos el listado de todos los cursos que imparte el centro.



### Iniciar sesión

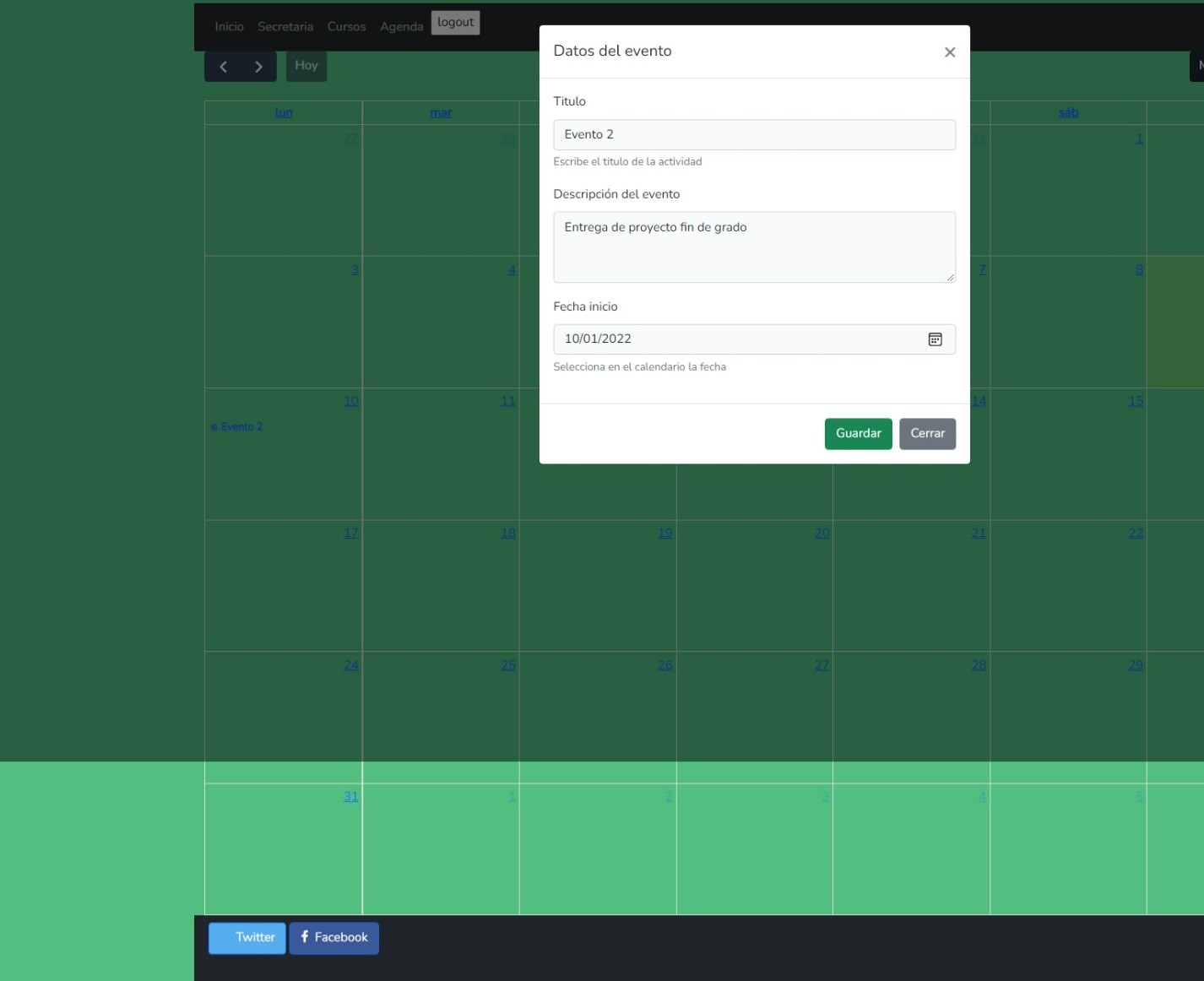
Aqui el usuario podra iniciar sesión.



## Usuario registrado

### Agenda

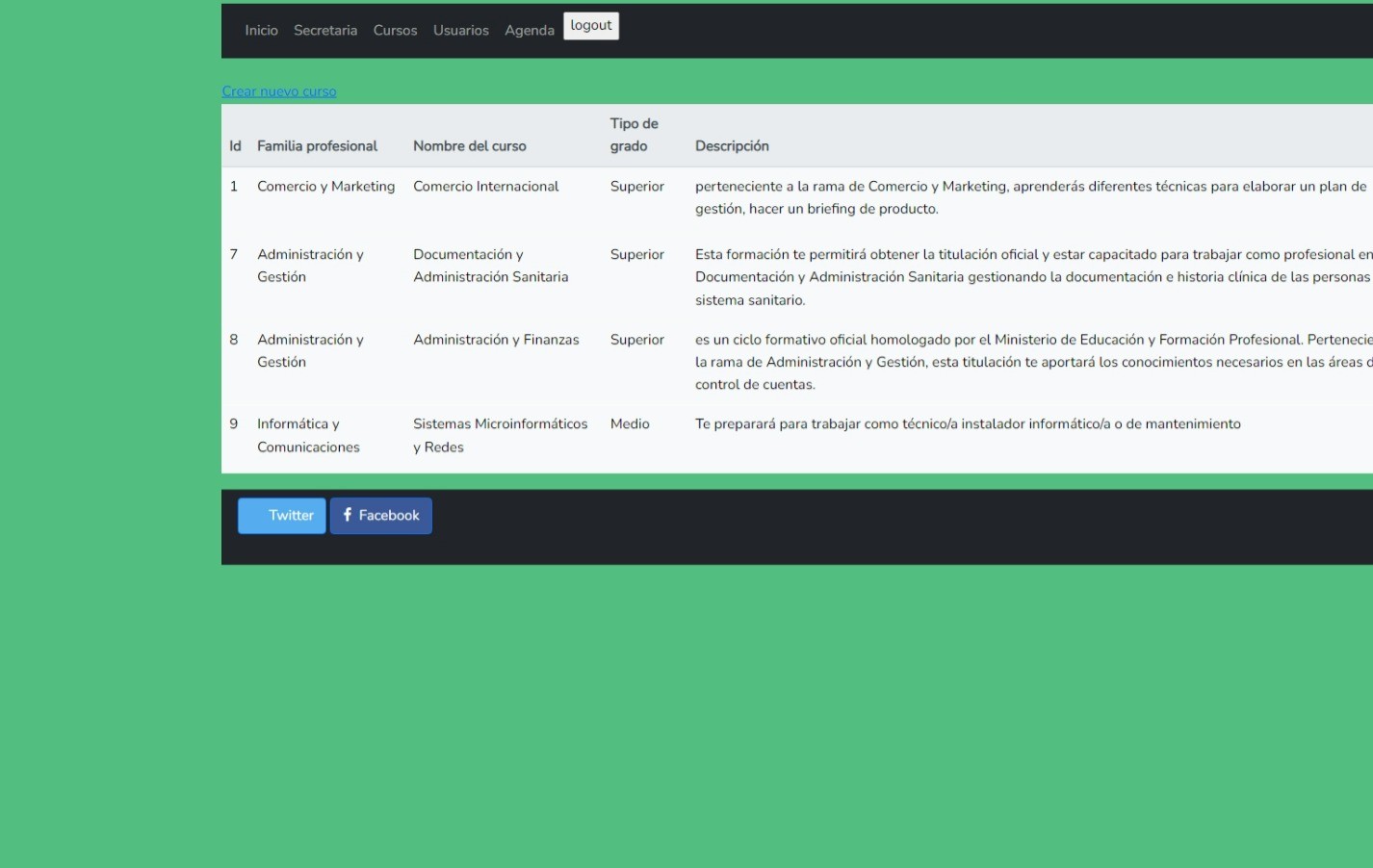
El usuario podrá ver todo lo anterior, ademas se añade al menu la opción Agenda. Para crear un nuevo evento debera seleccionar el día y rellenar los datos del formulario.



## Usuario Administrador

### Cursos

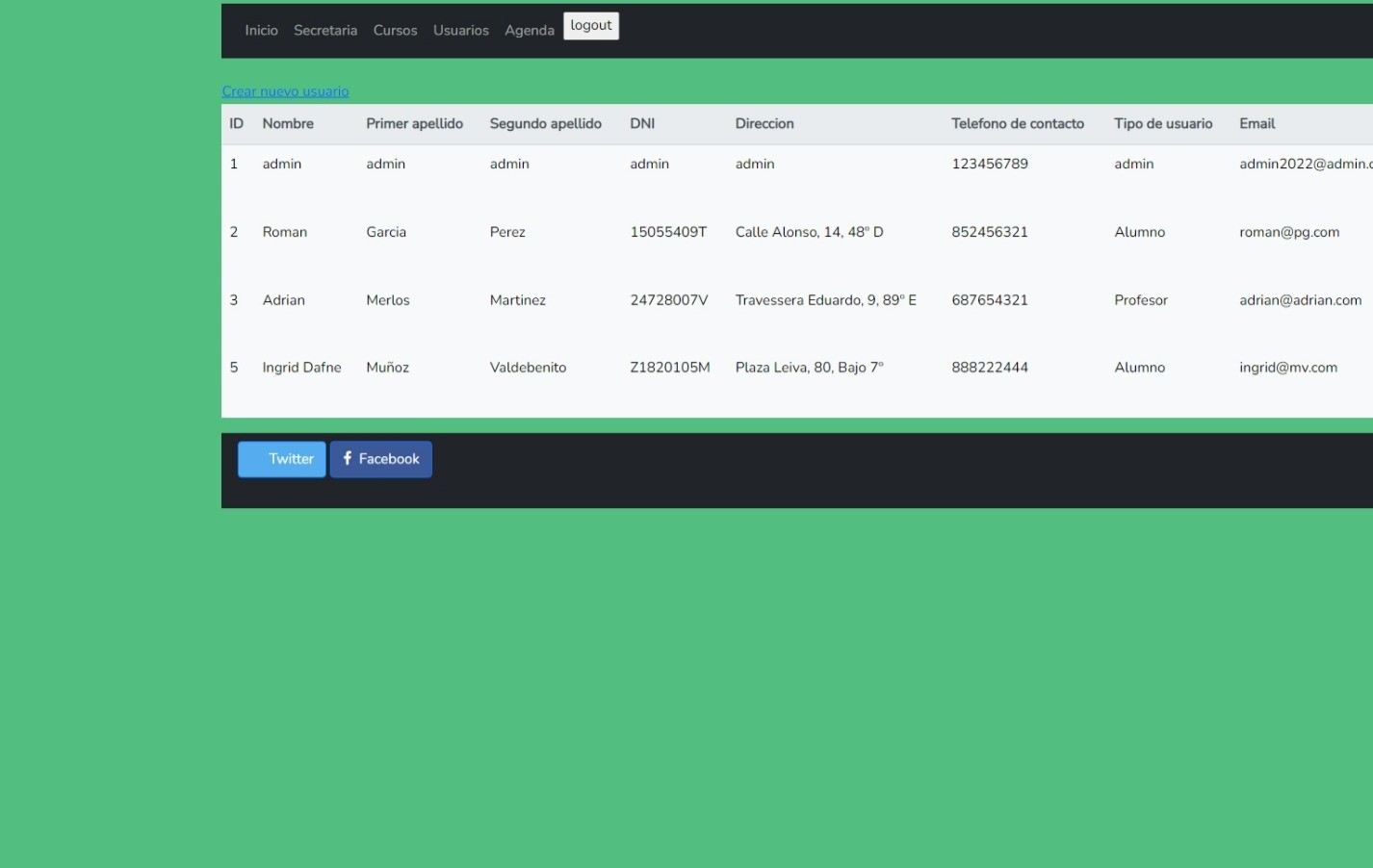
Aqui, se listan todos los cursos que imparte el centro. El administrador podrá crear, editar o borrar un curso.



### Formulario para crear un nuevo curso

Usuarios

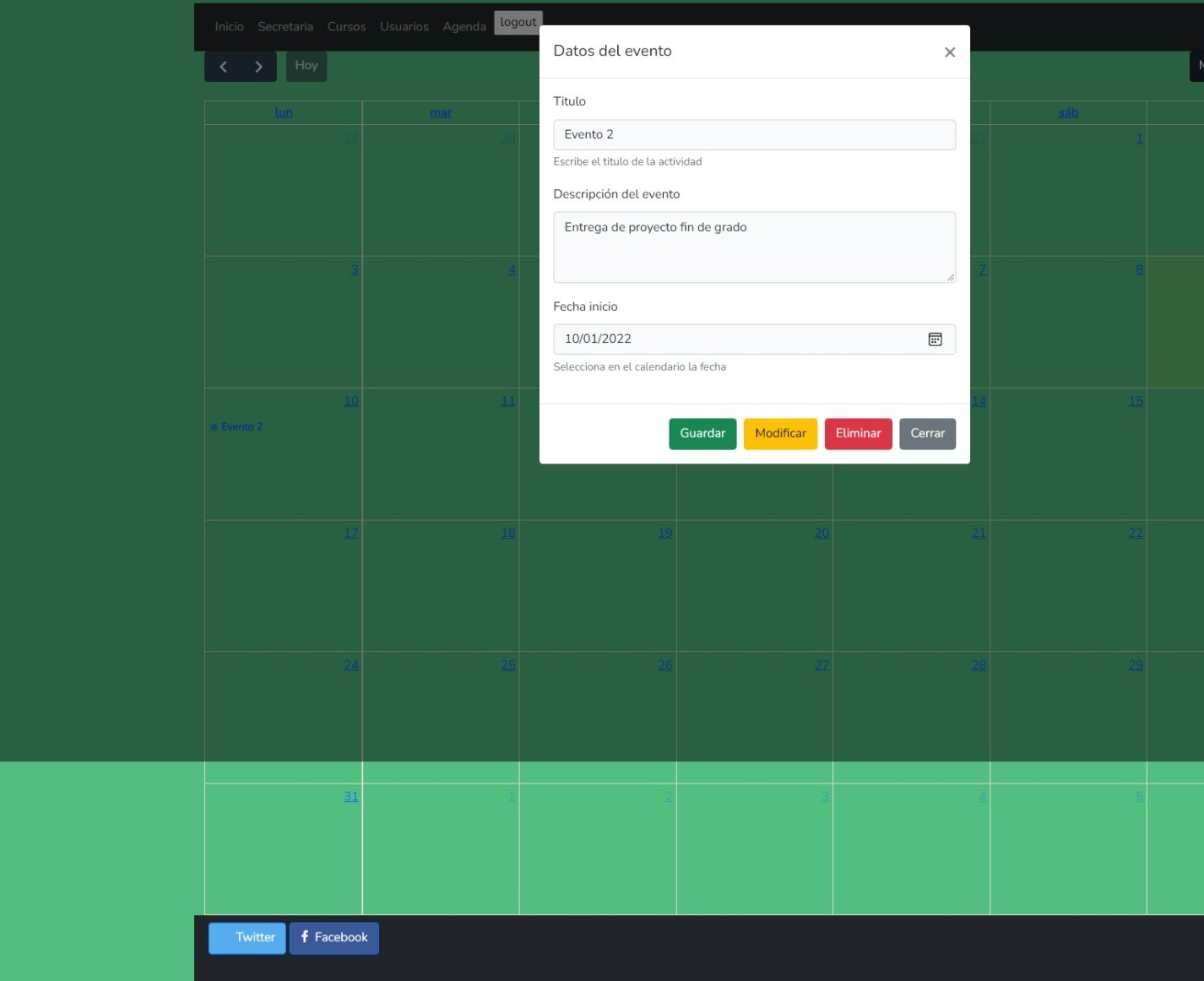
Aqui, se listan todos los usuarios del centro. El administrador podrá crear, editar o borrar un usuario.



### Formulario de registro de usuarios

Agenda

Aqui, el administrador podra crear, modificar y eliminar eventos.



## Resultados obtenidos y conclusiones

La realización del Trabajo Fin de Ciclo (TFC) contribuye a demostrar las habilidades adquiridas a lo largo de estos dos años de formación en el Ciclo Superior en Desarrollo de Aplicaciones Web. Este proyecto me ha servido para ver cómo se desarrolla una aplicación en el mundo laboral, y el buscar soluciones a problemas inesperados.

Este proyecto se basaba en una aplicación para crear eventos o actividades de un centro educativo. Cuando comencé con el proyecto no sabia muy bien como guiarlo y que necesitaba. Fuí añadiendo muchas opciones y me dí cuenta que estaba añadiendo cosas que no necesitaba.

Lo que mas me costó realizar fue la agenda y el CRUD de usuarios. Conseguí resolverlo y que me funcionará todo como queria. Me senti muy aliviada.

## Agradecimientos

Estos 2 años han sido complicados debido a la Covid-19 tanto para profesores como para alumnos y por eso quiero agradecir a todos los profesores que tuve estos 2 años por el tiempo y esfuerzo dedicado en enseñarnos y prepararnos para el futuro. Ademas, quiero agradecer a Adrian por escucharme y animarme cuando mas lo necesitaba y a mi compañero y mejor amigo Armando Gregorio Bizarro por todo sus consejos durante el proyecto.

De estos 2 años me llevo muchos recuerdos bonitos y graciosos que recordaré siempre.

## Webgrafía

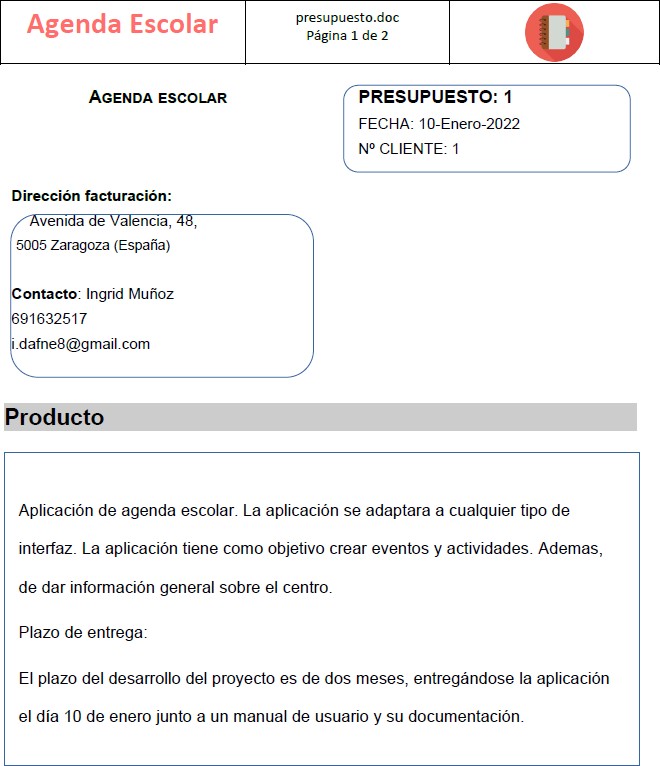
Creacion agenda: https://[www.youtube.com/watch?v=iyHt8Ew-l6U&ab\_channel=Develoteca](http://www.youtube.com/watch?v=iyHt8Ew-l6U&ab_channel=Develoteca)

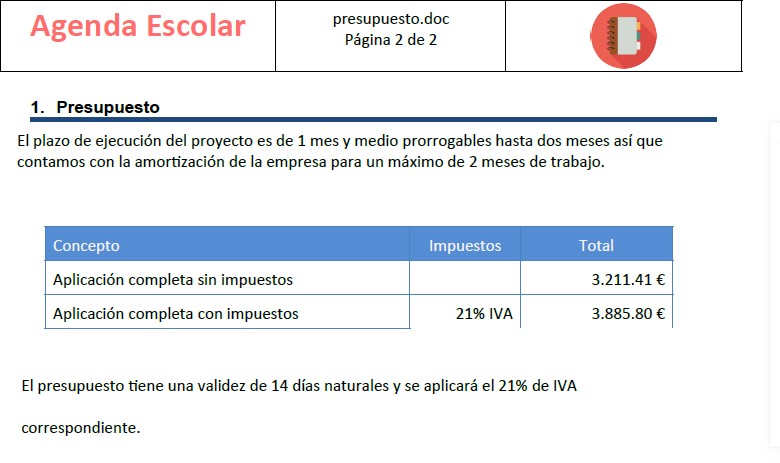
Guia Markdown: https://[www.youtube.com/watch?v=oxaH9CFpeEE&t=919s&ab\_channel=FaztCode](http://www.youtube.com/watch?v=oxaH9CFpeEE&t=919s&ab_channel=FaztCode) Guia Bootstrap: https://[www.w3schools.com/bootstrap4/](http://www.w3schools.com/bootstrap4/)

Guia Laravel: https://laravel.com/docs/8.x

# Anexos

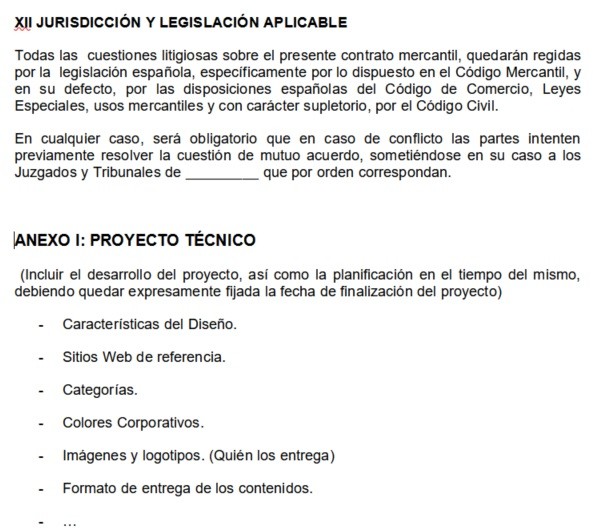
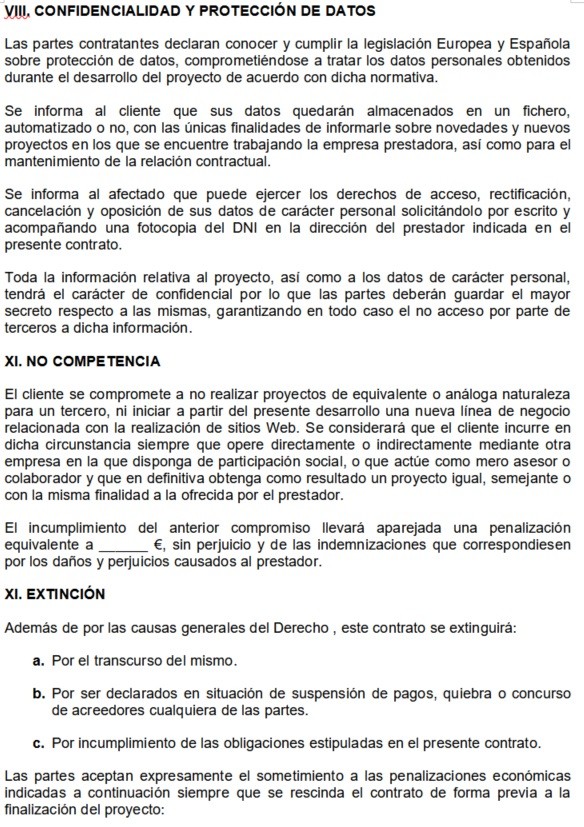
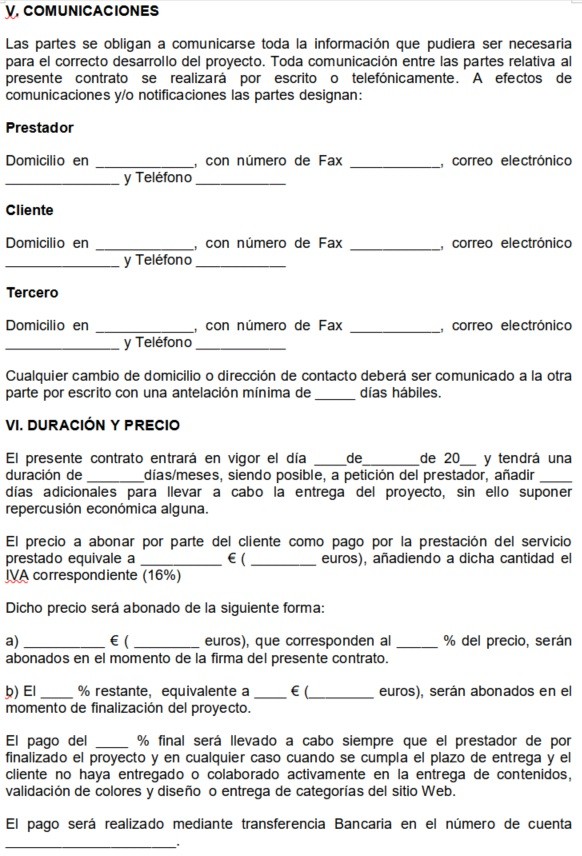
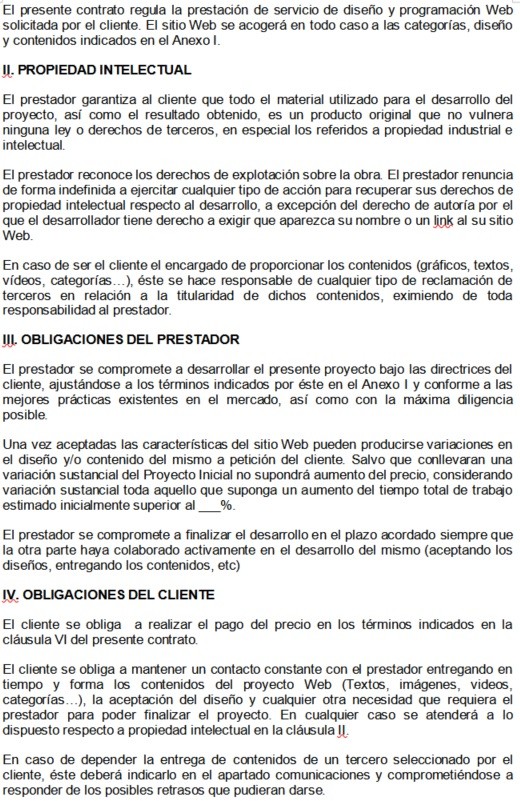
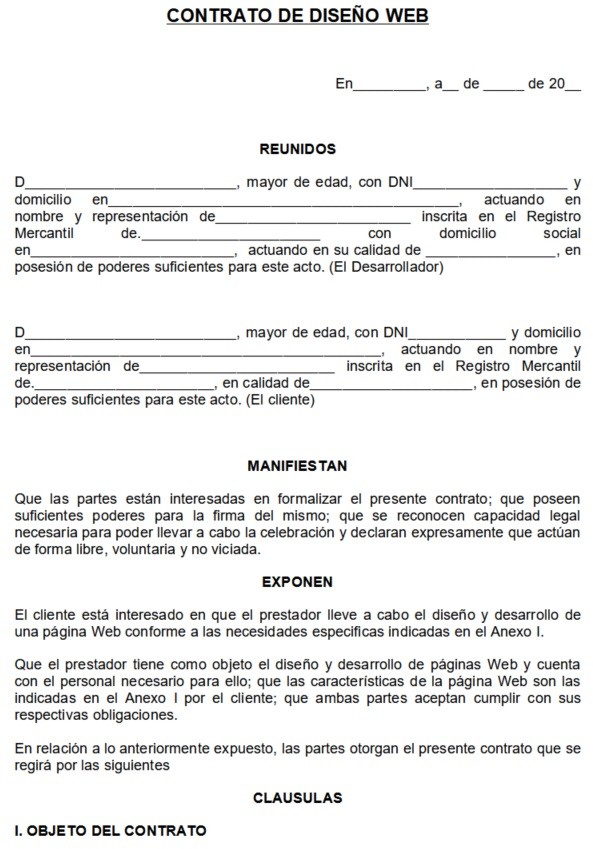
## Presupuesto





[presupuesto.pd](https://github.com/ingrid-munoz/Proyecto/files/7835771/presupuesto.pdf)f

## Contrato

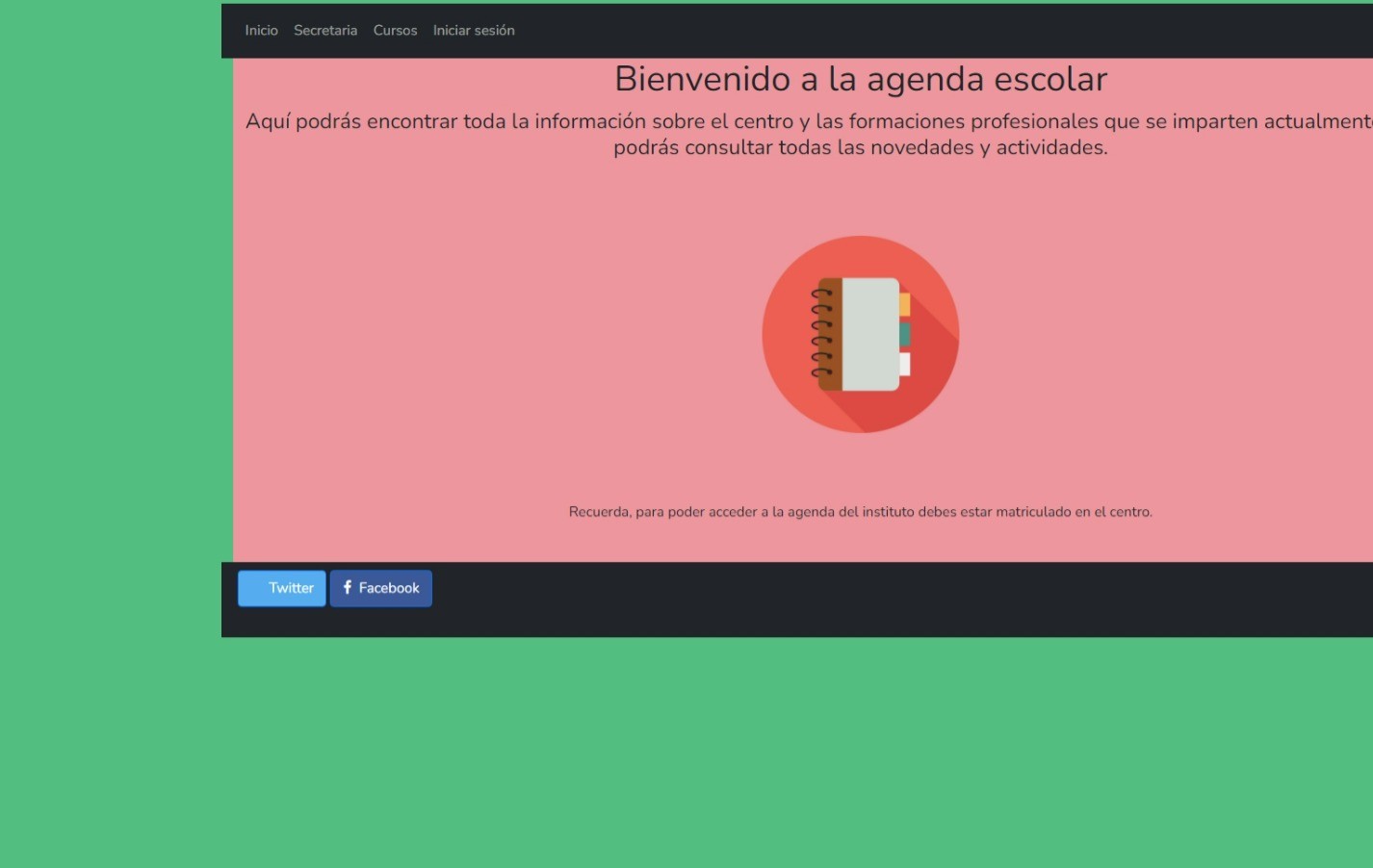


* 1. Documento de pruebas funcionales

## Usuario no registrado

### Inicio

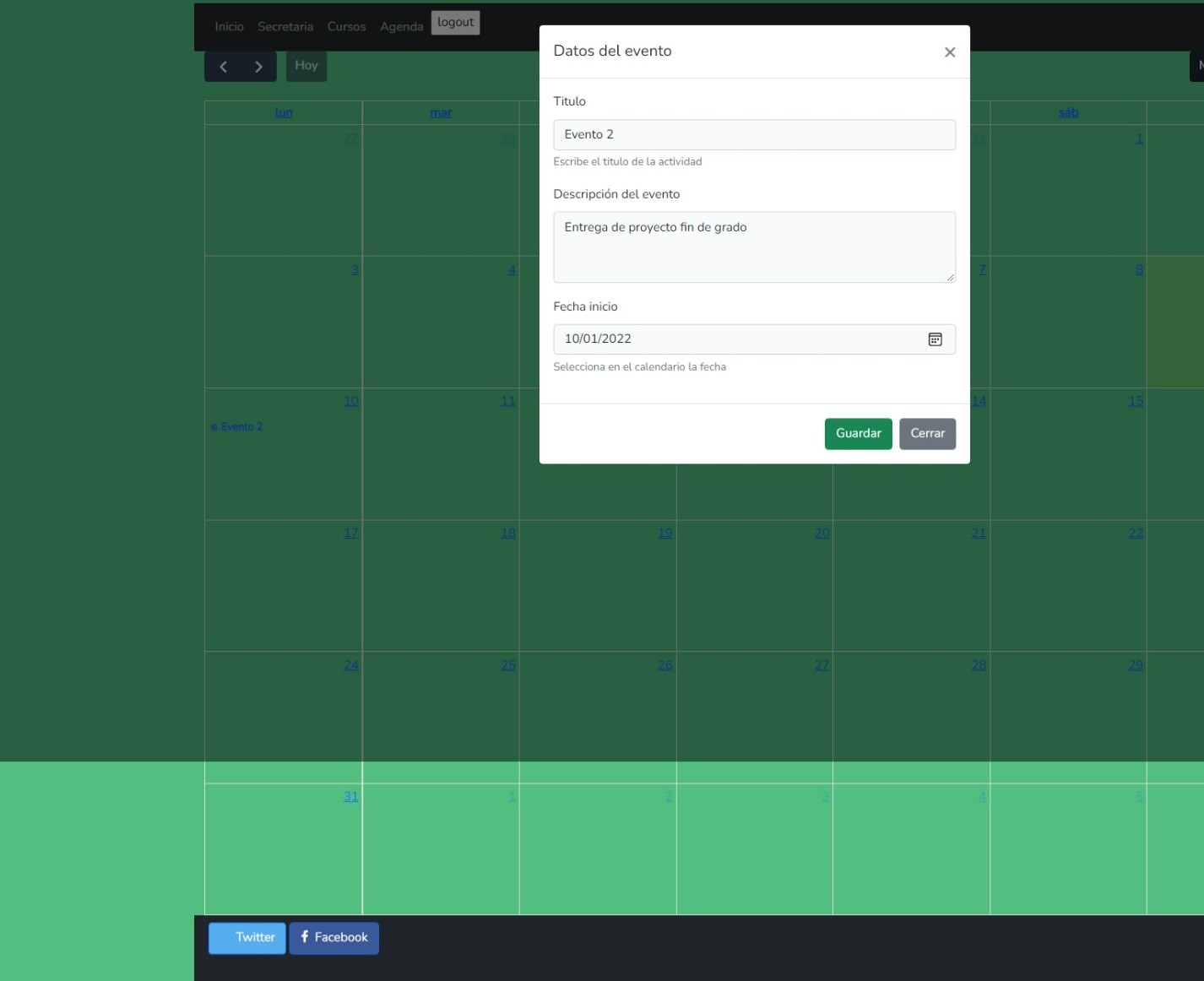
El usuario no registrado no podrá ver la opción de agenda en el menu.



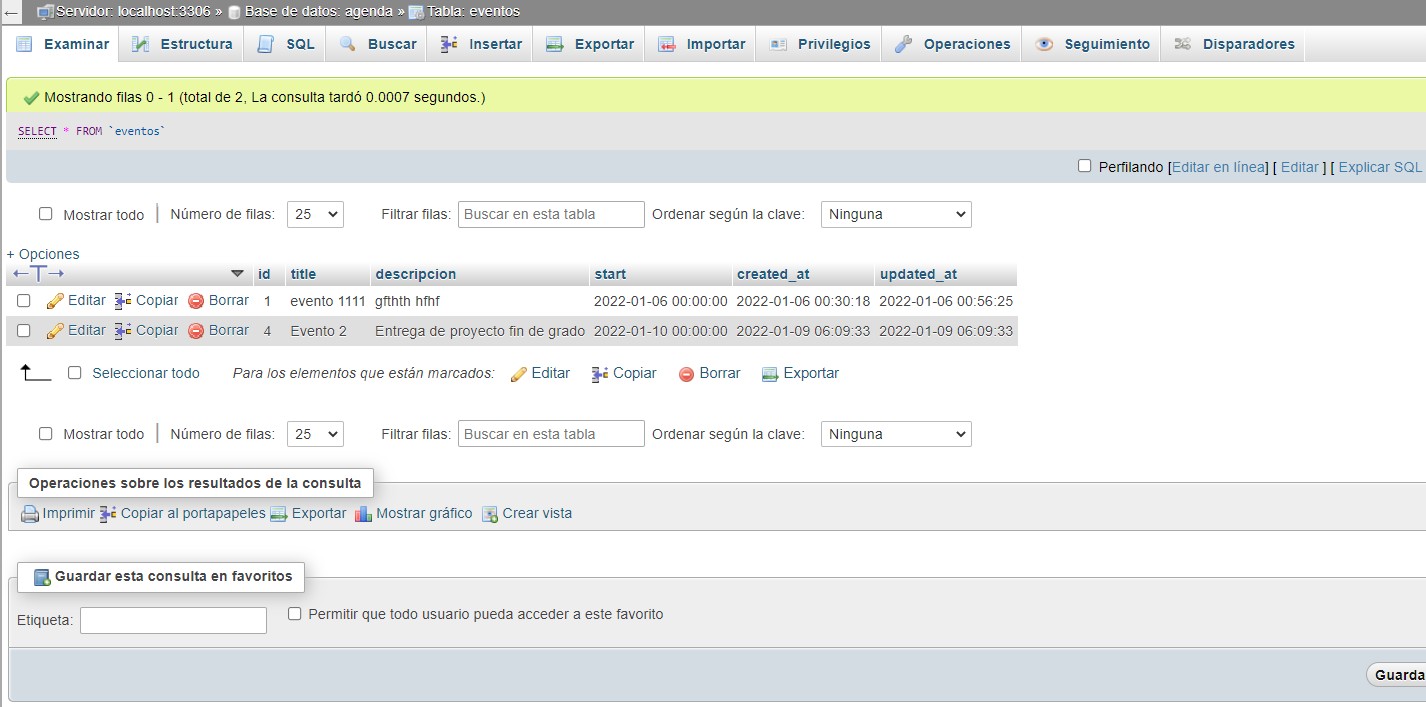
## Usuario registrado

### Agenda

El usuario registrado podrá crear un nuevo evento.



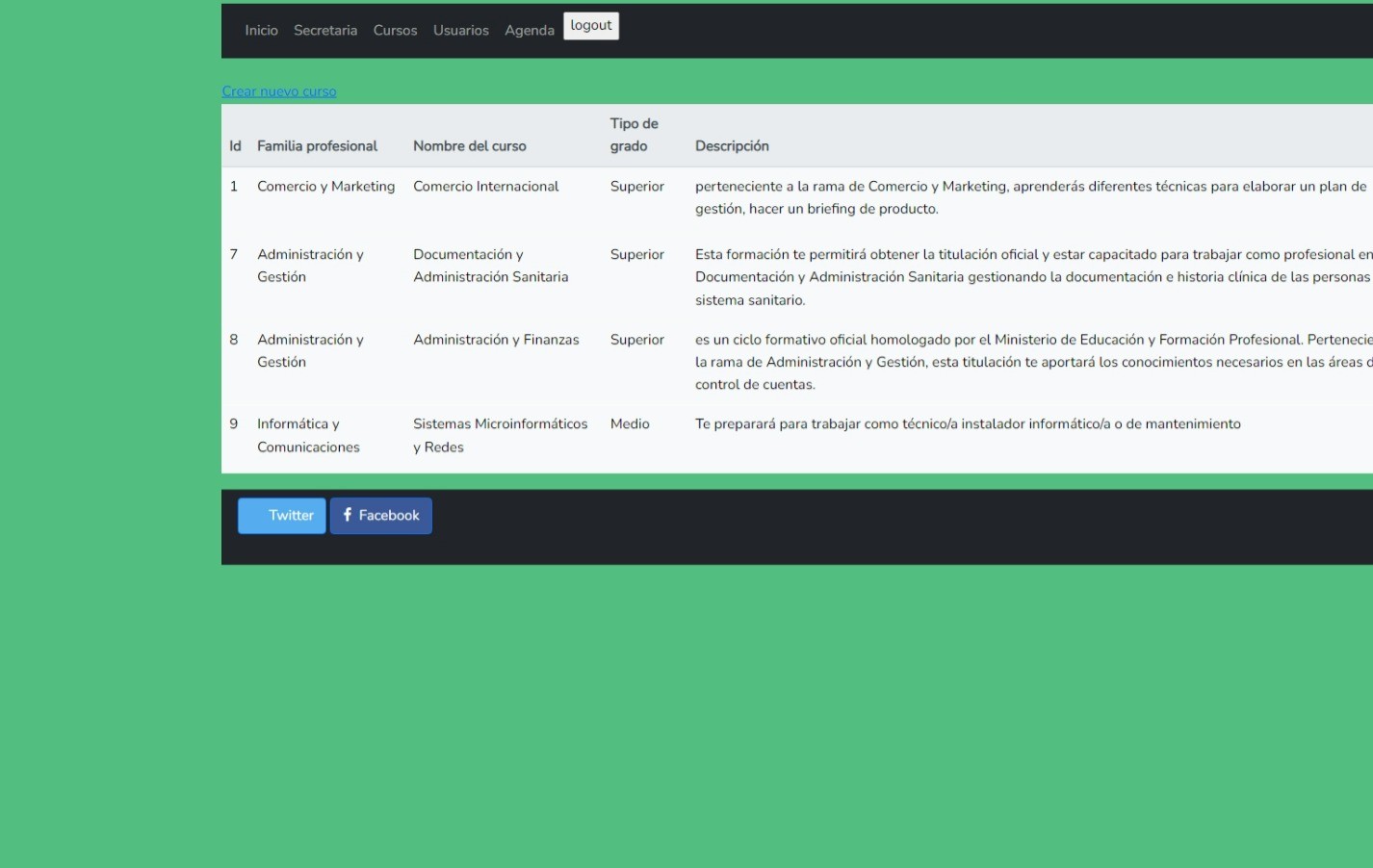
Comprobamos en la base de datos que se ha creado:



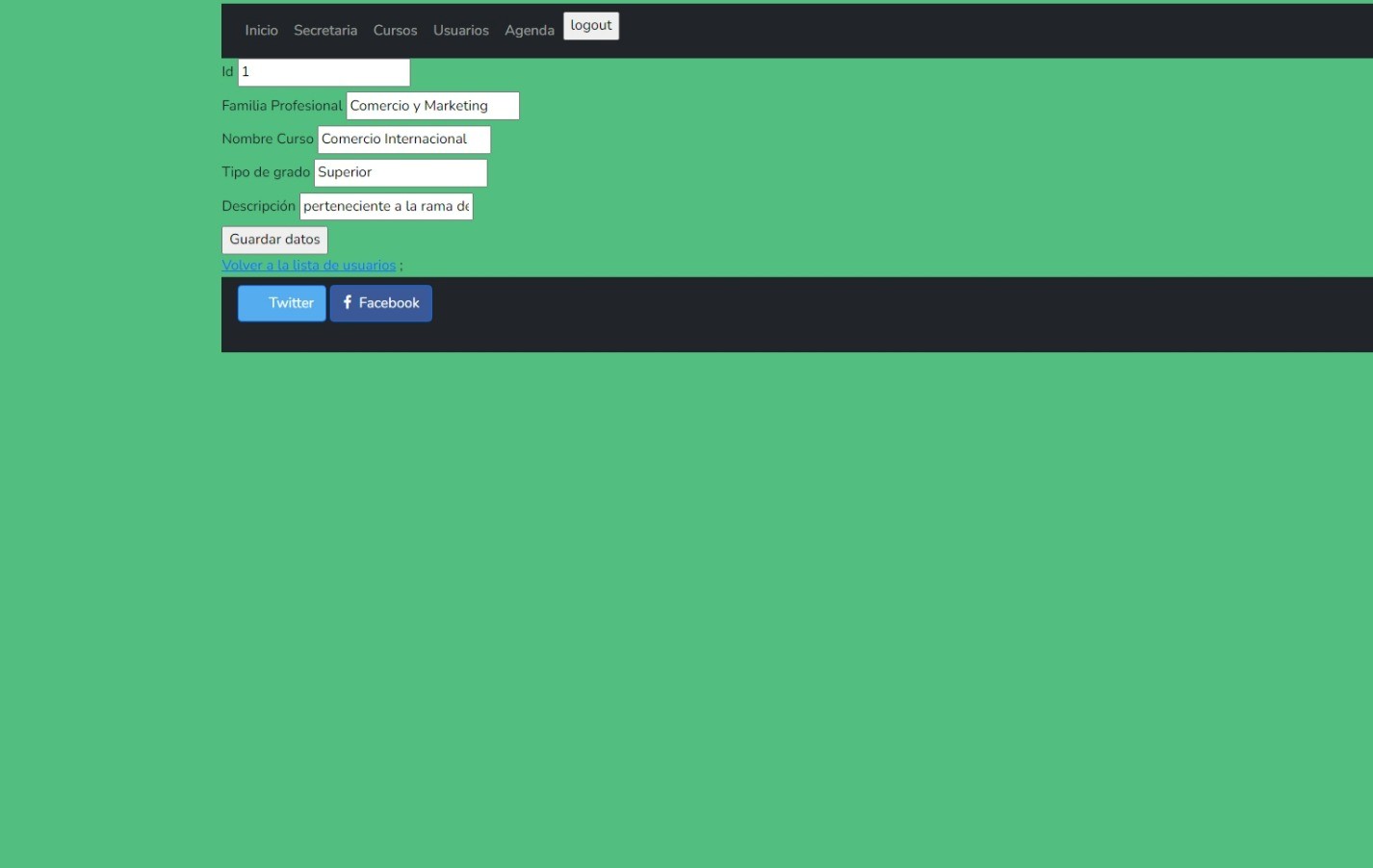
## Usuario Administrador

### Cursos

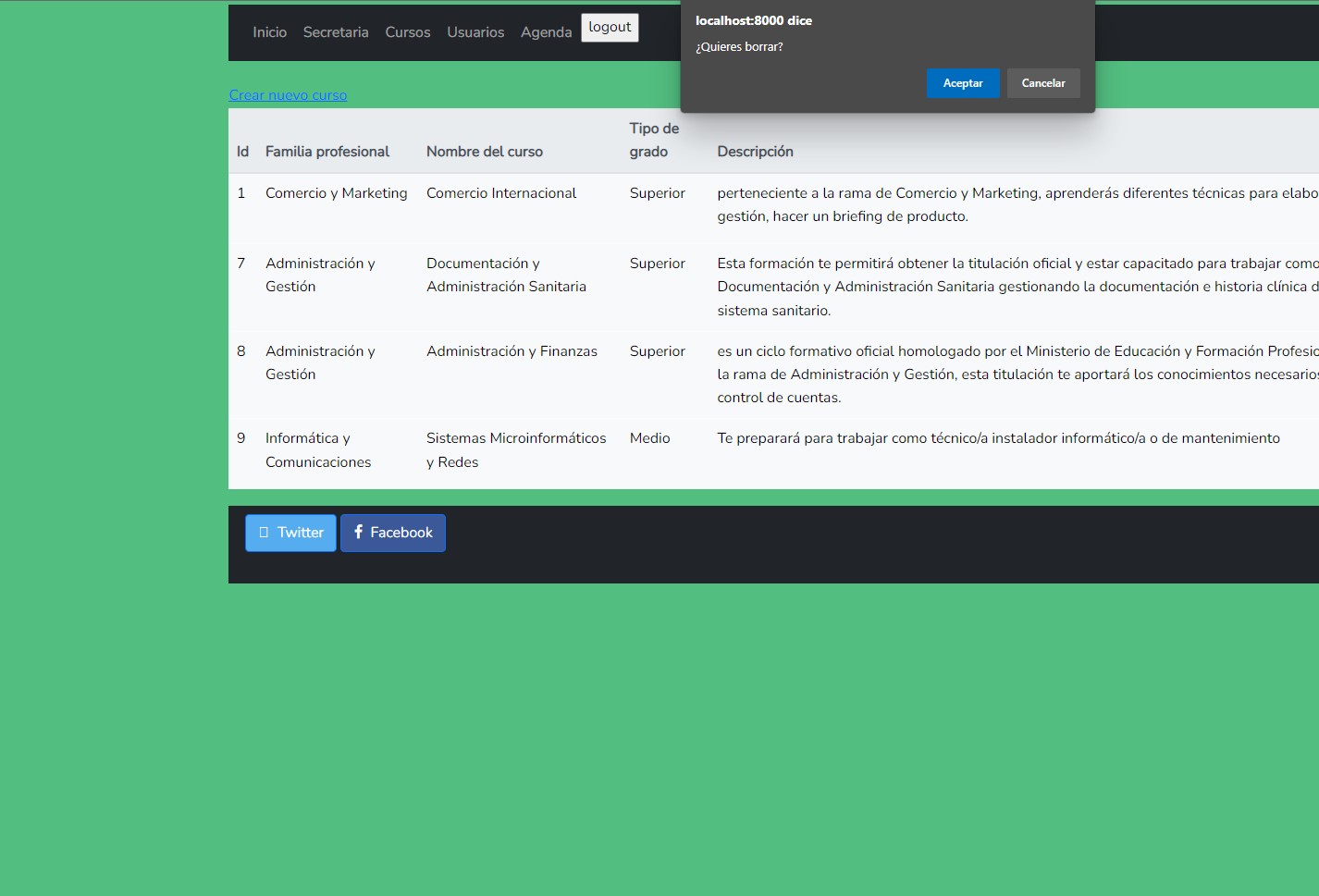
Aqui, se listan todos los cursos que imparte el centro. El administrador podrá crear, editar o borrar un curso.



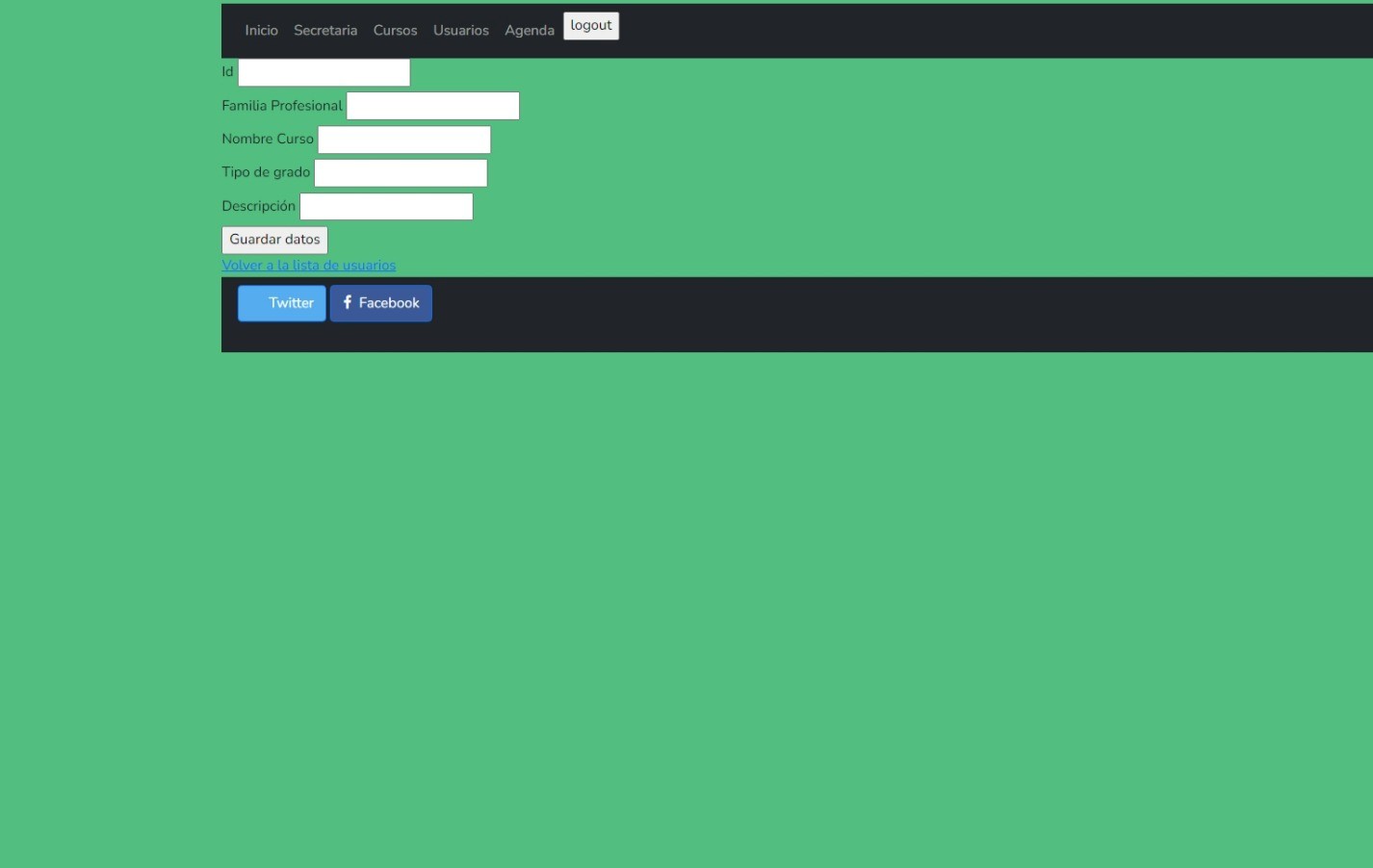
Opción Editar curso:



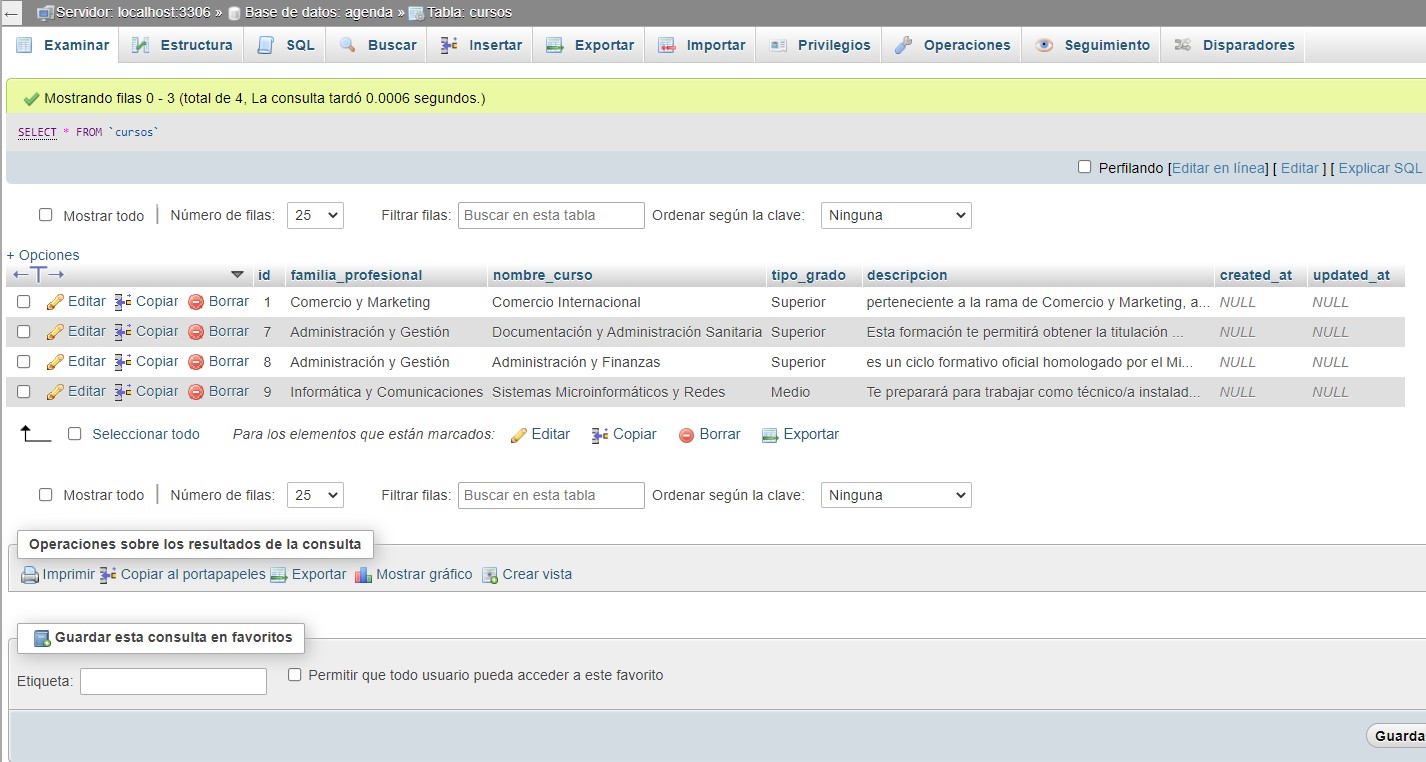
Opción Borrar curso:



Formulario para crear un nuevo curso:

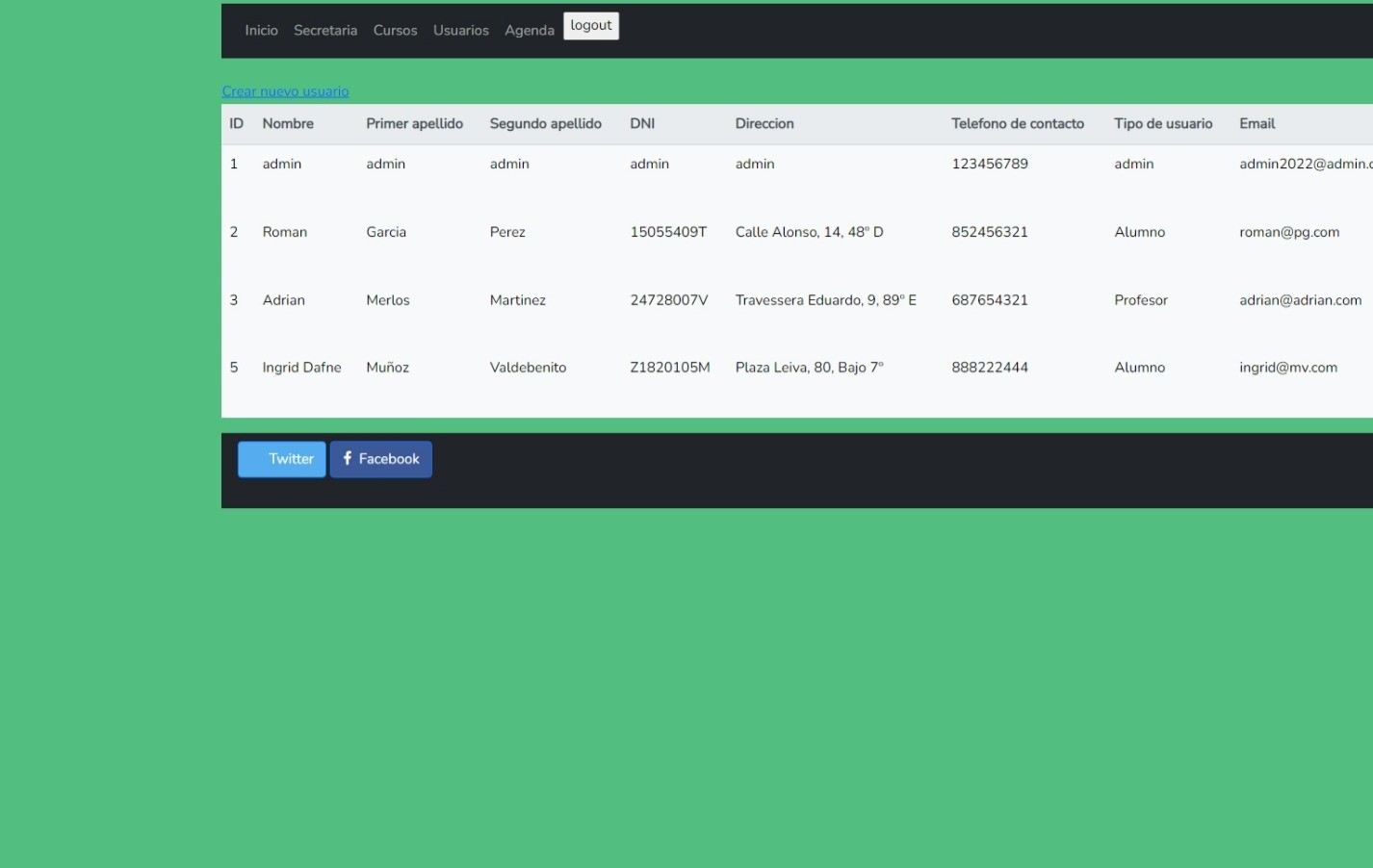


Comprobamos en la base de datos:

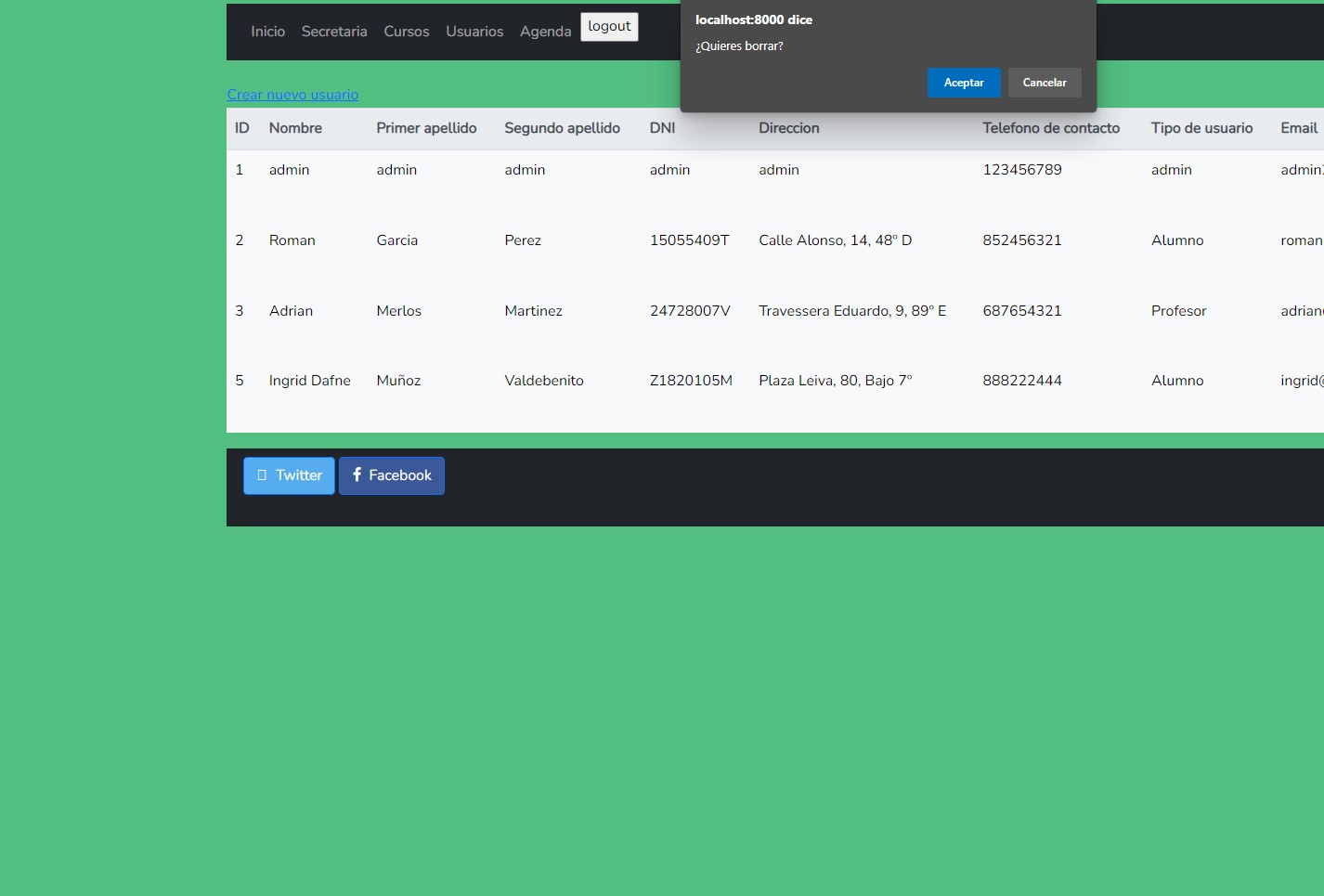


### Usuarios

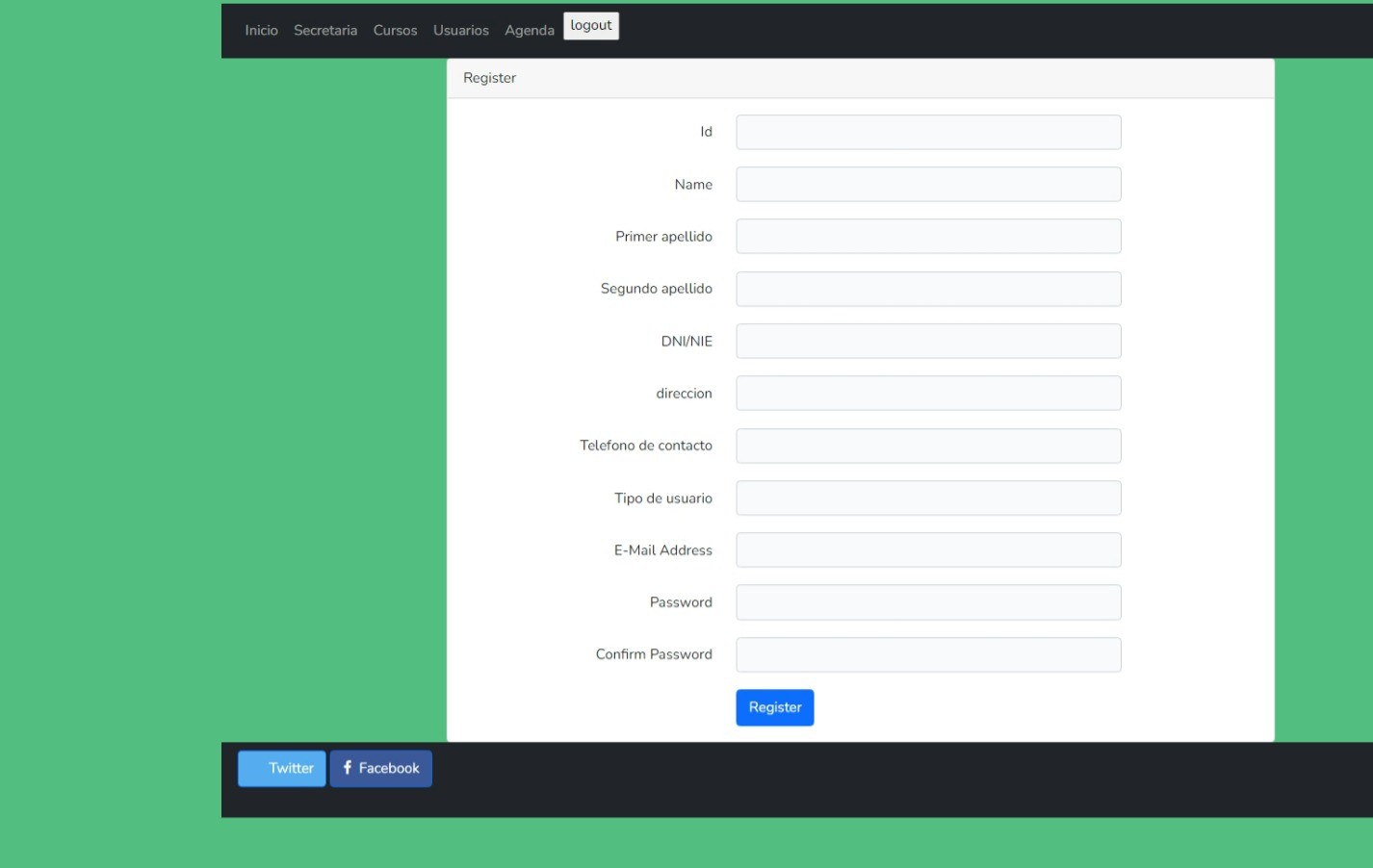
Aqui, se listan todos los usuarios del centro. El administrador podrá crear, editar o borrar un usuario.



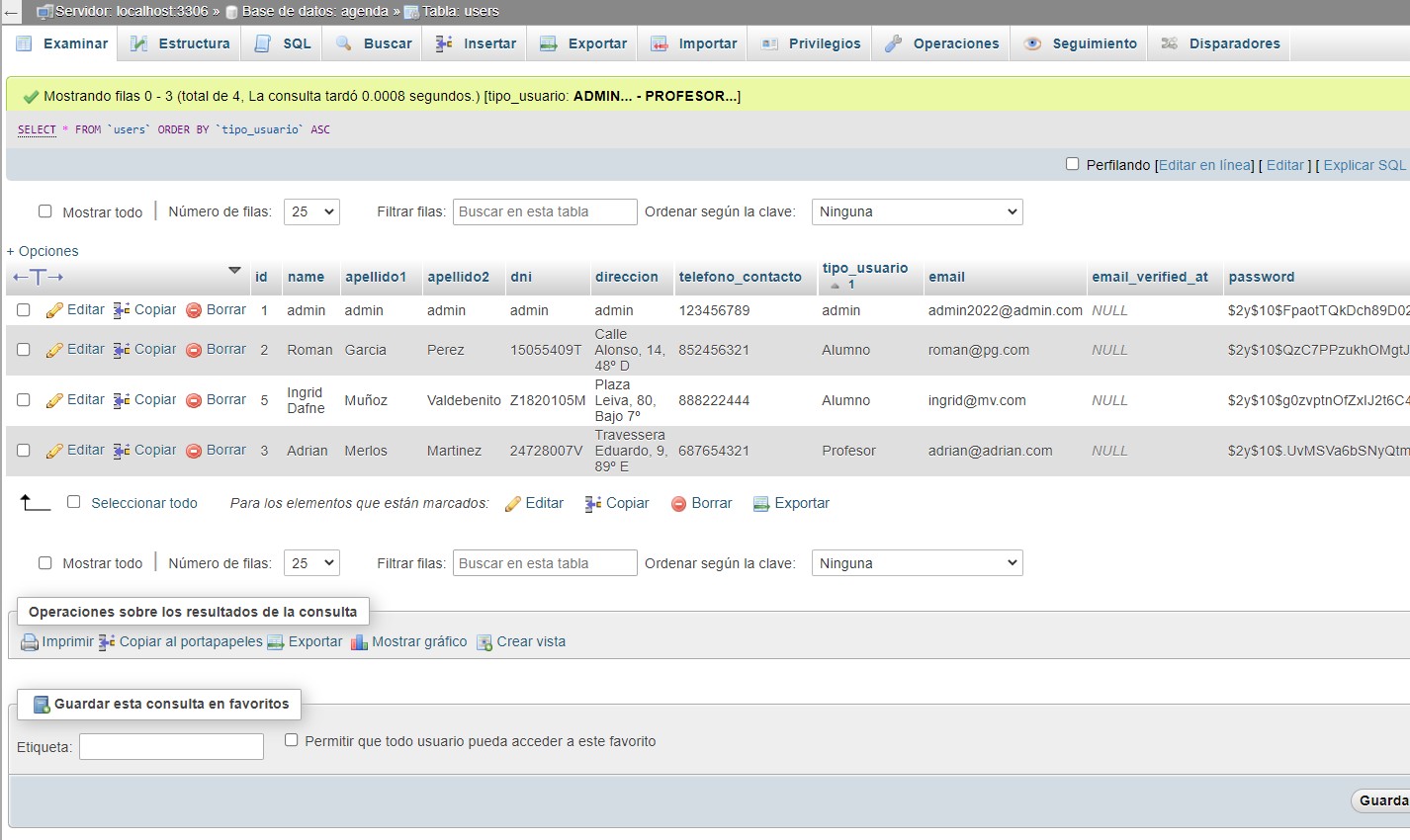
Opción Borrar usuario:



Formulario de registro de usuarios:

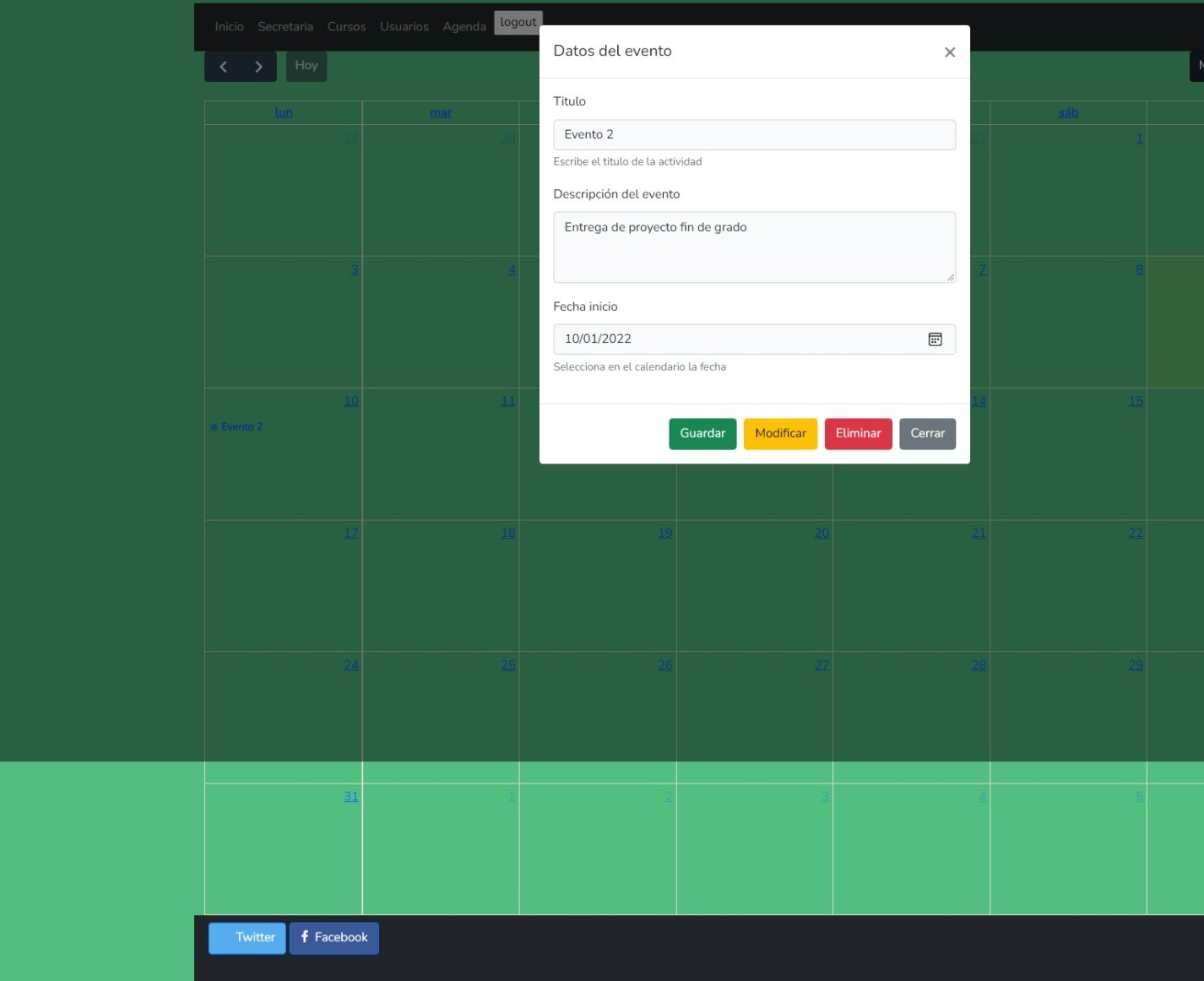


Comprobamos en la base de datos:



### Agenda

Aqui, el administrador podra crear, modificar y eliminar eventos.



Comprobamos en la base de datos:

