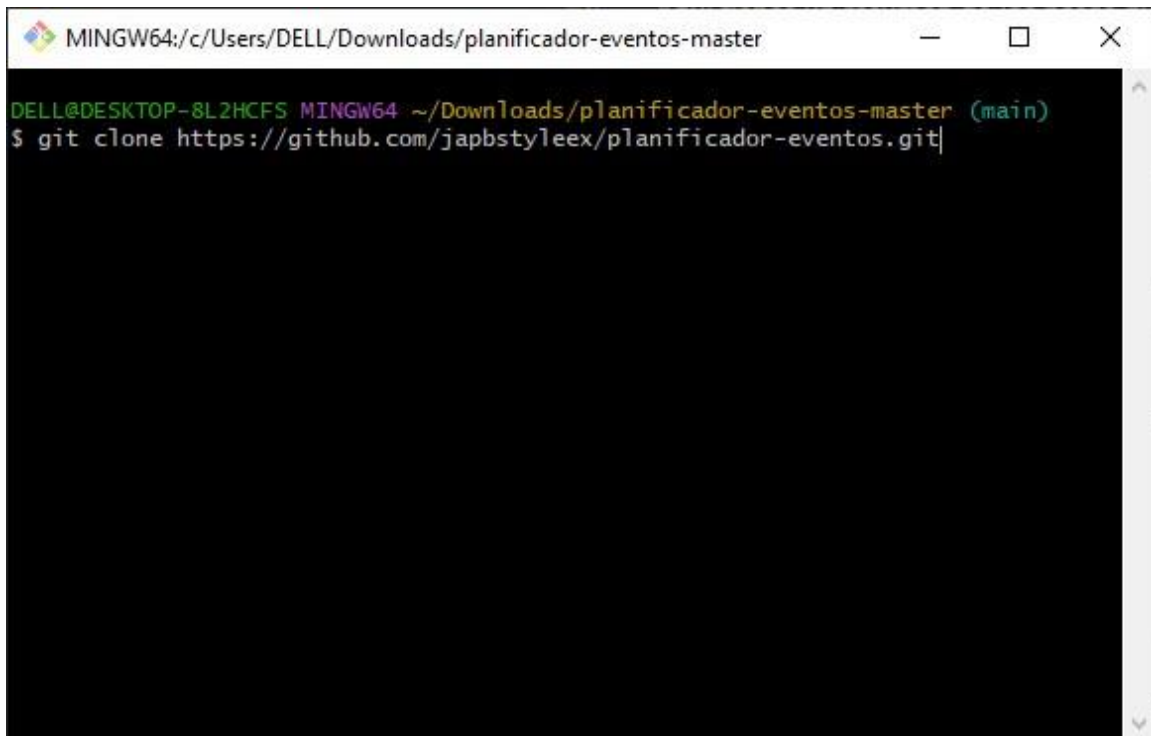


Guía de Despliegue del Proyecto Web

1. Clonación del repositorio

- Herramientas utilizadas: Git.
- Pasos:
 1. Abrimos una terminal en la máquina local.
 2. Creamos una carpeta para alojar el proyecto.
 3. Clonamos el repositorio utilizando el comando:

A screenshot of a terminal window titled 'MINGW64:/c/Users/DELL/Downloads/planificador-eventos-master'. The terminal shows the command '\$ git clone https://github.com/japbstyleex/planificador-eventos.git' being entered. The prompt is 'DELL@DESKTOP-8L2HCFS MINGW64 ~/Downloads/planificador-eventos-master (main)'. The terminal has a black background with green and white text. The window has standard Windows window controls (minimize, maximize, close) in the top right corner.

```
MINGW64:/c/Users/DELL/Downloads/planificador-eventos-master
DELL@DESKTOP-8L2HCFS MINGW64 ~/Downloads/planificador-eventos-master (main)
$ git clone https://github.com/japbstyleex/planificador-eventos.git
```

Resultado: Los archivos del proyecto quedaron disponibles localmente para su modificación y configuración.

2. Creación de la base de datos

- Herramientas utilizadas: CDmon (gestión de hosting) y phpMyAdmin.
- Pasos:
 1. Accedimos al panel de CDmon con las credenciales proporcionadas.

Información del Hosting

Eliminar

Dominio asociado	armave.es	Cambiar	Caducidad	23/02/2025	Renovar
Tipo de plan	Plataforma de Pruebas	Publicar	Espacio web ocupado	0,02 GB de 5,00 GB	
Titular	Sin cliente facturación ()		Versión PHP	8.2	Cambiar
Usuario MySQL	myarmaveae		Contraseña MySQL	*****	Cambiar
Usuario FTP	armavedd		Contraseña FTP	*****	Cambiar
IP web	134.0.10.37		Transferencia consumida	22,09 MB de 5,00 GB	+ info

2. Creamos una base de datos en la sección de gestión de bases de datos.

Listado de bases de datos MySQL

Crear una nueva base de datos MySQL

Acciones	Nombre	Espacio	Acciones
<input type="checkbox"/>	armave		Eliminar

3. Nos aseguramos de utilizar un nombre único para evitar conflictos en un entorno de alojamiento compartido.

4. Conectamos a phpMyAdmin para gestionar la base de datos.

The screenshot shows the phpMyAdmin interface for a database named 'armave'. The left sidebar shows the database structure with 'preferencias_usuarios' and 'usuarios' tables. The main area displays a list of tables with columns: Tabla, Acción, Filas, Tipo, Cotejamiento, Tamaño, and Residuo a depurar. The 'usuarios' table is highlighted, showing 6 rows and a size of 80.0 KB. Below the table list, there is a 'Crear nueva tabla' section with fields for 'Nombre de la tabla' and 'Número de columnas' (set to 4), and a 'Crear' button.

3. Importación del archivo SQL

- Herramienta utilizada: phpMyAdmin.

- Pasos:

1. Nos autenticamos en phpMyAdmin.
2. Seleccionamos la base de datos creada previamente.
3. Utilizamos la opción Importar para cargar el archivo .sql proporcionado por el proyecto.
4. Verificamos que las tablas y datos se hayan importado correctamente.

```

1  -- Crear base de datos (si no existe)
2  CREATE DATABASE IF NOT EXISTS armave;
3
4  -- Usar la base de datos
5  USE armave;
6
7  -- Crear la tabla de usuarios
8  CREATE TABLE IF NOT EXISTS usuarios (
9      id INT AUTO_INCREMENT PRIMARY KEY,
10     nombre_usuario VARCHAR(50) NOT NULL UNIQUE,
11     correo_electronico VARCHAR(100) NOT NULL UNIQUE,
12     contrasena VARCHAR(255) NOT NULL, -- Se recomienda almacenar contraseñas hasheadas
13     rol ENUM('admin', 'usuario') DEFAULT 'usuario',
14     fecha_registro TIMESTAMP DEFAULT CURRENT_TIMESTAMP
15 );
16
17 -- Insertar varios usuarios
18 INSERT INTO usuarios (nombre_usuario, correo_electronico, contrasena, rol)
19 VALUES
20     ('admin', 'admin@ejemplo.com', 'admin', 'admin'),
21     ('joseperales', 'joseantonioperalesbayon@gmail.com', 'contrasenia1', 'usuario'),
22     ('aroa', 'aroafcor@gmail.com', 'contrasenia2', 'usuario'),
23
24 -- Crear una tabla adicional si fuera necesario (por ejemplo, para almacenar preferencias de usu
25 CREATE TABLE IF NOT EXISTS preferencias_usuarios (
26     id INT AUTO_INCREMENT PRIMARY KEY,
27     id_usuario INT,
28     preferencia VARCHAR(100),
29     valor VARCHAR(100),
30     FOREIGN KEY (id_usuario) REFERENCES usuarios(id)
31 );
32
33 -- Insertar algunas preferencias para los usuarios
34 INSERT INTO preferencias_usuarios (id_usuario, preferencia, valor)
35 VALUES
36     (1, 'tema', 'oscuro'),
37     (2, 'tema', 'claro'),
38     (3, 'notificaciones', 'activadas'),
39     (4, 'notificaciones', 'desactivadas');
40
41 -- Ver los usuarios
42 SELECT * FROM usuarios;
43

```

Se encontró un repositorio Git en las carpetas primarias del área de trabajo o en los archivos abiertos. ¿Desea abrir el repositorio?

Origen: GIT. Sí Siempre Nunca

4. Configuración del archivo conexion_be.php

- Herramientas utilizadas: Editor de texto local (Nano en Git Bash o cualquier otro editor de código).

- Pasos:

1. Accedimos al archivo conexion_be.php en la carpeta raíz del proyecto.
2. Configuramos las credenciales de conexión con la base de datos:

```

1  <?php
2  $servername = "localhost"; // Servidor de base de datos, normalmente 'localhost'
3  $username = "myarmaveae"; // El nombre de usuario de tu base de datos
4  $password = "BvWd9A5K"; // La contraseña de tu base de datos
5  $dbname = "armave"; // El nombre de la base de datos
6
7  // Crear la conexión
8  $conexion = new mysqli($servername, $username, $password, $dbname);
9
10 /*
11 if ($conexion){
12     echo "conectado";
13 }else{
14     echo "no conectado";
15 }
16 */
17 ?>
18

```

Se encontró un repositorio Git en las carpetas primarias del área de trabajo o en los archivos abiertos. ¿Desea abrir el repositorio?

Origen: GIT. [Sí](#) [Siempre](#) [Nunca](#)

No se puede validar porque no se ha encontrado una instalación de PHP. Utilice el ajuste "php.validate.executablePath" para configurar el ejecutable de PHP.

Origen: Características del lenguaje PHP. [Abrir configuración](#)

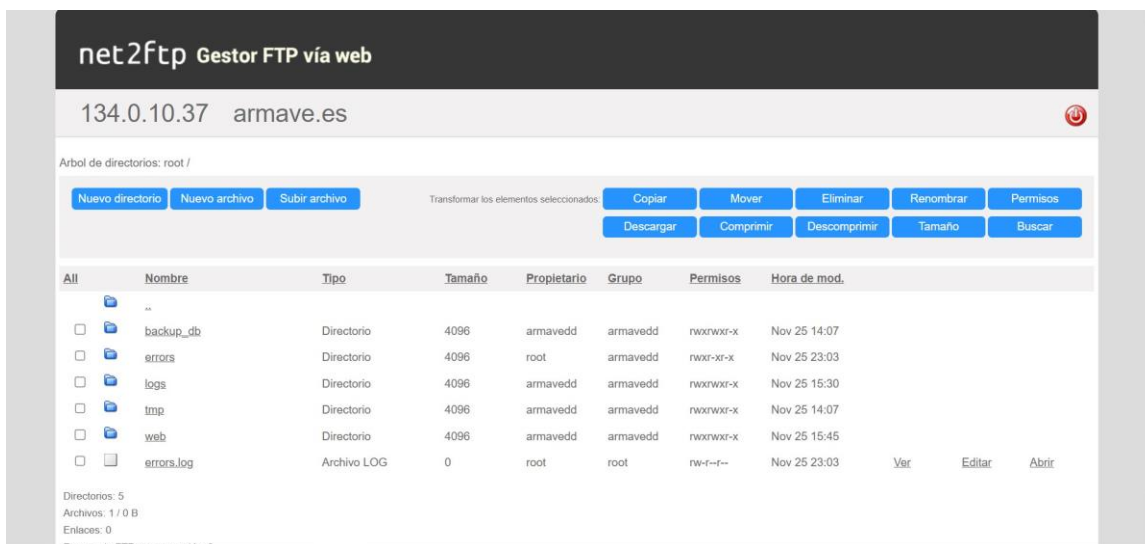
3. Guardamos los cambios y probamos la conexión.

5. Subida de archivos al hosting

- Herramientas utilizadas: Net2FTP (gestor FTP web).

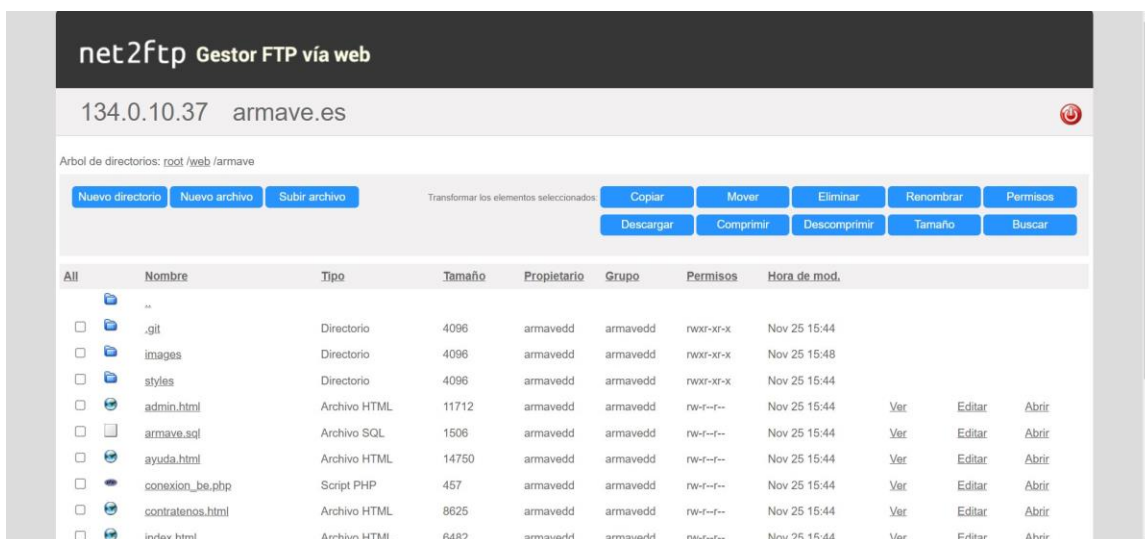
- Pasos:

1. Accedimos a Net2FTP utilizando las credenciales de FTP del hosting en CDmon.



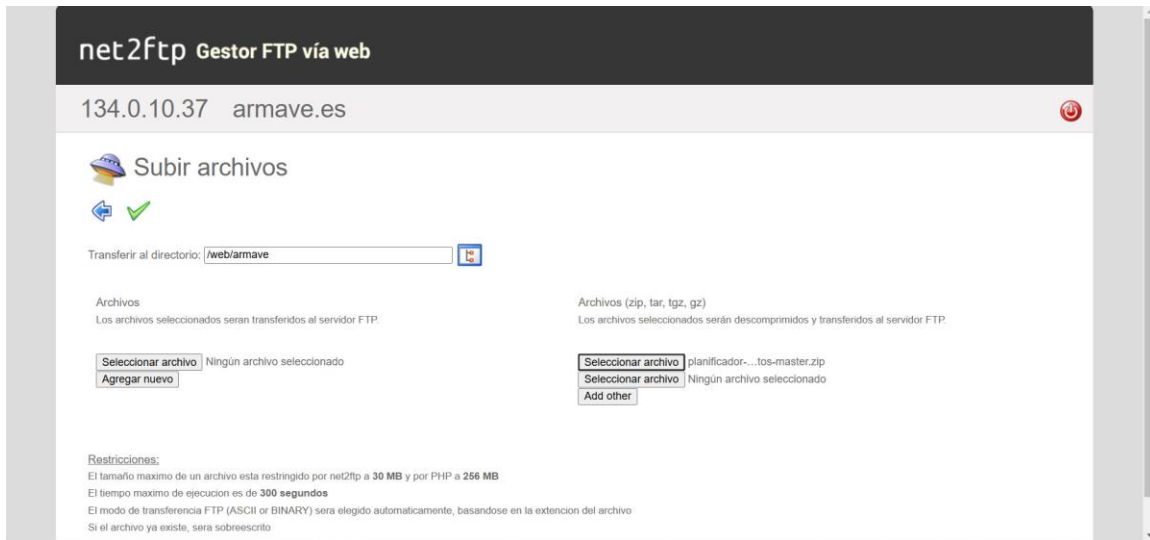
2. Navegamos al directorio /web dentro de la raíz del servidor.

3. Creamos una carpeta específica para el proyecto.



4. Comprimimos la carpeta del proyecto localmente y subimos el archivo comprimido al servidor.

5. Utilizamos la opción Descomprimir para extraer los archivos en la carpeta creada.

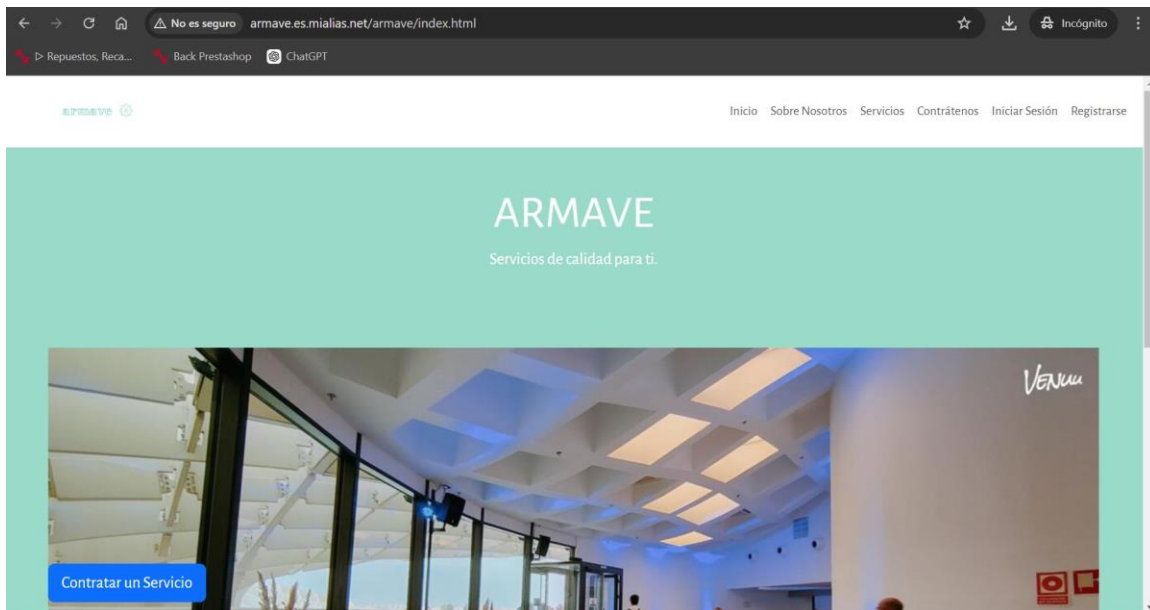


Nota: Como el archivo es demasiado grande subí primero los códigos y luego las imágenes para que no me diese problemas

6. Pruebas del proyecto desplegado

- Pasos:

1. Accedimos al proyecto mediante la URL pública proporcionada por el hosting.



7. Actualización del repositorio en GitHub

- Herramientas utilizadas: GitHub, Git Bash.
- Pasos:
 1. Agregamos cambios al repositorio local:
`git add .`
`git commit -m 'Descripción de los cambios'`
 2. Subimos los cambios al repositorio remoto:
`git push origin main`

Tecnologías y herramientas utilizadas

1. Lenguaje de programación: PHP.
2. Base de datos: MySQL.
3. Gestores y servicios:
 - Hosting: CDmon.
 - Gestor FTP: Net2FTP.
 - Gestor de base de datos: phpMyAdmin.
4. Control de versiones: Git y GitHub.

Aclaraciones

El proyecto como se puede comprobar no esta completo aún, falta la configuración del código para conectar con la base de datos y sus debidas pruebas, más las inclusiones necesarias explicadas en la idea para que el proyecto sea 100% funcional y correcto a lo que se quiere mostrar.