

PROYECTO CONJUNTO

2º DAW



Alexandru Dumitru y Alberto Privado



Documentación

Descripción del proyecto.....	3
Tecnologías usadas.....	3
Tecnologías y explicación.....	3
Porque hemos usado esas tecnologías.....	4
Forma de combinar las tecnologías.....	5
Iniciando el proyecto.....	5
Como trabaja WordPress con Laravel y viceversa.....	5
Posible mejora.....	6
Características especiales del proyecto.....	7
Multilenguaje.....	7
Las horas de activación.....	7
Manual de usuario.....	7
Usuario administrador.....	7
Usuario técnicos.....	7
Usuarios normales.....	7

Descripción del proyecto

Proyecto para una empresa del sector tecnológico, concretamente en el nicho de las redes, con el nombre “GoWifi” la cual se encarga de instalar puntos de acceso en diferentes instituciones publicas y privadas o simplemente en vías públicas.

Con el fin de aportar una conectividad pública fuera del domicilio personal de sus clientes.

“¿Quién no ha echado en falta el Wifi fuera de casa?”.

Esta empresa tiene a su cargo unos técnicos, los cuales realizan la instalación y mantenimiento de los equipos.

Tecnologías usadas

Tecnologías y explicación

- **Git y GitHub**: Herramienta para control de versiones.
- **WordPress**: Es un CMS que permite crear y mantener un blog u otro tipo de web.
- **Laravel**: Es un Framework PHP simple, potente y con una interfaz elegante.
- **Bootstrap**: Es un Framework CSS.
- **AJAX**: Tecnología para crear aplicaciones interactivas, ejecutada en el cliente.
- **JSON**: Formato para la manipulación de datos.
- **DataTables**: Librería para crear tablas.
- **Chart.js**: Librería para crear gráficos.

Porque hemos usado esas tecnologías

Git y GitHub Nos va a facilitar mucho el trabajo en grupo y el control de versiones de la aplicación.

WordPress va a ser el encargado de brindar una llamativa interfaz tanto para los clientes que desean solicitar información de la empresa como a los administradores que van a interactuar y gestionar los servicios de la empresa.

Laravel va a ser el encargado de apoyar a WordPress en tareas más pesadas de Backend. Ya que programarlas con WordPress sería mucho más laborioso.

Bootstrap ahorra mucho tiempo a la hora de desarrollar plantillas o modificar una. La temática a usar es empresarial entonces no va a ser un diseño súper extravagante, por lo cual Bootstrap viene fenomenal para desarrollar cualquier diseño.

AJAX nos permite fusionar de forma sencilla los datos de Laravel con Wordpress para crear las tablas y los gráficos de estadísticas.

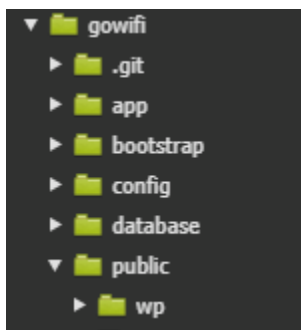
➔ Resumiendo:

Laravel va a ser el encargado de gestionar todas las interacciones de Backend y Bootstrap junto con WordPress se encargará de todo el Frontend aunque WordPress también hará sus aportaciones.

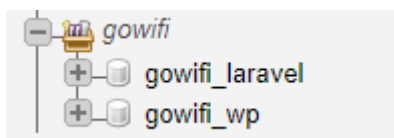
Forma de combinar las tecnologías

Iniciando el proyecto

Creamos una aplicación Laravel y dentro de la aplicación en la carpeta public instalamos nuestro WordPress.



Se crea una base de datos apartada de WordPress para Laravel.



Una vez todo configurado, hemos optado por empezar a desarrollar el tema y tener una base de diseño en la que trabajar.

El siguiente paso ha sido desarrollar los recursos (Controladores, Modelos y Migraciones) en Laravel.

Como trabaja WordPress con Laravel y viceversa

Hemos utilizado esta disposición de carpetas para que WordPress sea la web que veas siempre pero pueda tener la posibilidad de comunicarse con Laravel.

WordPress funcionará de forma normal, para las diferentes vistas/páginas del sitio se debe crear una plantilla personalizada.

En Laravel para cada funcionalidad que se añada al sitio hay que crear habilitarla en los recursos creados.

WordPress desde la vista enviará los datos a Laravel y los procesará devolviéndolos a WordPress.

WordPress tiene alguna API personalizada para que Laravel al enviarle algún dato comprometido sea el propio WordPress quien gestione la petición para que no haya ningún problema entre medias.

→ **Ejemplo:**

Una vista de WordPress hace login, pero el login es enviado a Laravel que hace login en la app de Laravel y seguido Laravel consume la API de WordPress para hacer login y WordPress redirige a la página correspondiente.

Posible mejora

Se intento combinar WordPress y Laravel de diferentes formas y no conseguí resolverlo. Al final todo se diferencio en que debí haber escrito una linea encima de otra.

La idea era cargar dentro de Laravel la funcionalidad de WordPress, consiguiendo que dentro de Laravel pudieses utilizar todas las funciones de WordPress pero con toda la potencialidad que ofrece Laravel.

Características especiales del proyecto

Multilinguaje

Se ha configurado un plugin que realiza la traducción del sitio. Y gracias a que todo lo que se muestra es en WordPress podemos traducir todo, porque los datos que le pedimos a Laravel se muestran en WordPress.

Las horas de activación

No se pueden solapar las fechas ni horas a la hora de crear rangos activos en los que recibir datos de conexiones.

Se ha creado un algoritmo que compruebe si la conexión va a ser guardada o no.

Para hacer pruebas hemos dejado que los usuarios logueados puedan conectarse a los puntos de acceso.

Manual de usuario

Usuario administrador

El administrador podrá acceder a todas las características, entre ellas crear técnicos.

Usuario técnicos

Los técnicos tendrán acceso a todo menos a crear o ver los técnicos empleados.

Usuarios normales

Un usuario normal podrá ver los puntos de acceso y conectarse a ellos. Pero no tendrán acceso a nada más.