

**PRÁCTICA FINAL LARAVEL. 2ª EVALUACIÓN.**

**DESARROLLO WEB EN ENTORNO SERVIDOR.**

**Juan Antonio Dólera Jiménez**

**2º Desarrollo de Aplicaciones Web**

**Profesor/a: Fernando Ureña**

Contenido

[Sprint 1: Configuración inicial del proyecto y autenticación 3](#_Toc188430413)

# Sprint 1: Configuración inicial del proyecto y autenticación

**Objetivo**: Configurar el entorno, establecer la arquitectura básica (MVC), y

preparar el sistema de autenticación y control de versiones.

**Tareas:**

1. **Configuración inicial:**

* Crear el proyecto Laravel.

Para crear el proyecto con laravel, debemos escribir la siguiente línea de comandos en la PowerShell, dentro de la carpeta de proyecto que queremos empezar.

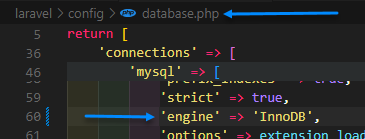


* Configurar base de datos (MySQL) y archivo .env.

Dentro del archivo .env, cambiamos esta línea de código para asignar el nombre de la base de datos.



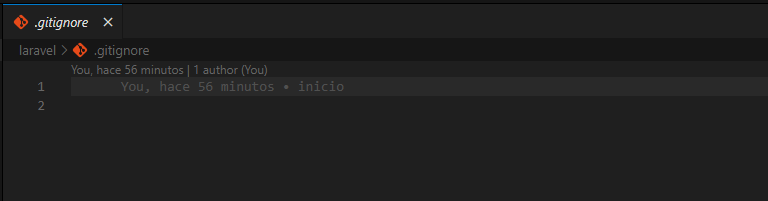
Más adelante, cuando hagamos las migraciones de las tablas desde la PowerShell se creará automáticamente la base de datos en PHPMyAdmin. Para evitar fallos con las migraciones, debemos cambiar esta línea en el siguiente archivo:



* Configurar un sistema de control de versiones con Git y establecer un repositorio en GitHub.

Para unir nuestro repositorio local con GitHub hemos usado Fork. Hemos creado un nuevo repositorio en la web de GitHub, y con Fork hemos iniciado el repositorio en local. Para subirlo a GitHub simplemente debemos hacer un push con el mismo Fork e implementar el enlace del repositorio en la nube, el cual se nos crea cuando hacemos el repositorio.

Antes de subirlo a GitHub, debemos borrar el archivo de .gitignore para que no haya ningún archivo que perdamos cuando hagamos el push. Este archivo lo encontramos en la carpeta de laravel. En mi caso lo he dejado vacío.



1. **Autenticación:**

* Usar Laravel Breeze o Laravel UI para generar un sistema básico de autenticación.
* Configurar acceso a parte pública y privada.

1. **Modelo y migración inicial:**

* Crear las tablas principales con migraciones:
* Usuarios (users).
* Productos (productos).
* Categorías (categorias).
* Implementar relaciones entre las tablas:
* Cada producto pertenece a una categoría.

1. **Estructura inicial del proyecto:**

* Organizar controladores, modelos y vistas bajo el patrón MVC.
* Implementar middleware para proteger las rutas del administrador.

1. **Documentación:**

* Redactar instrucciones para configurar y ejecutar el proyecto en local.