**Manual del Sistema – Proyecto PIA en Laravel**

**Requerimientos del Sistema**

**Requisitos del software**

* **PHP 8.x**
* **Composer**
* **Node.js**
* **PostgreSQL**
* **Laravel 11**
* **Visual Studio Code**
* **Git y GitHub**
* **Navegador web (Chrome, Firefox, etc.)**

**Requisitos del hardware**

* **Procesador: 1.6 GHz o superior**
* **RAM: 4 GB mínimo**
* **Espacio en disco: 1 GB disponible**
* **Conexión a internet para instalar dependencias**

**Creación del Proyecto – Paso a Paso**

**Este proyecto fue realizado a partir de una plantilla base proporcionada por el profesor. A continuación, se explican los pasos desde la clonación hasta su ejecución local.**

**Paquetes Utilizados**

**Breeze**

* **Función: Maneja la autenticación de usuarios (login, registro, recuperación de contraseña).**

**Instalación:**

**en el CMD = composer require laravel/breeze --dev**

**php artisan breeze:install**

**npm install && npm run dev**

**php artisan migrate**

**Spatie Laravel Permission**

* **Función: Permite gestionar roles y permisos de usuarios.**
* **Instalación:**

**en el CMD=composer require spatie/laravel-permission**

**php artisan vendor:publish --provider="Spatie\Permission\PermissionServiceProvider"**

**php artisan migrate**

**Clonación del Proyecto del Profesor**

**Pasos para clonar el proyecto:**

1. **Ubicarse en una carpeta del disco local  
    Ejemplo: cd C:\xampp\htdocs**
2. **Clonar el repositorio del profesor: en el CMD=git clone** [**https://github.com/jaimesoto/clase\_aut.git**](https://github.com/jaimesoto/clase_aut.git)
3. **Ingresar a la carpeta del proyecto: en el CMD= cd clase\_aut**
4. **Actualizar dependencias con Composer: en el CMD= composer update**
5. **Crear el archivo .env: Copiar el archivo .env.example y renombrarlo a .env en el CMD=cp .env.example .env**
6. **Configurar el archivo .env:**

* **Configurar nombre de base de datos, usuario y contraseña.  
  en el CMD= DB\_CONNECTION=pgsql**
* **DB\_HOST=127.0.0.1**
* **DB\_PORT=5432**
* **DB\_DATABASE=aut**
* **DB\_USERNAME=postgres**
* **DB\_PASSWORD=tu\_contraseña**

**7. Crear la base de datos en pgAdmin o desde consola:**

**Nombre de la base de datos: aut**

**8. Ejecutar las migraciones:**

**en el CMD= php artisan migrate**

**9. Generar la key de la aplicación:**

**php artisan key:generate**

**Configuración del Proyecto Laravel**

**Archivo .env:**

* **Contiene configuración sensible como la base de datos, claves API y nombre de aplicación.**

**Base de datos en PostgreSQL:**

* **Crear una base llamada aut.**
* **Configurar credenciales en el archivo .env.**

**Archivo php.ini:**

* **Asegurarse de que las siguientes extensiones estén activas:**
  + **extension=pgsql**
  + **extension=pdo\_pgsql**

**Migraciones:**

**Las migraciones se ejecutan con:**

**en el CMD= php artisan migrate**

**Modelos:**

* **Se encuentran en app/Models/, representan las tablas de la base de datos.**

**Controladores:**

* **Ubicados en app/Http/Controllers/, contienen la lógica de las acciones del sistema.**

**Vistas (Blade):**

* **Están en resources/views/, contienen el HTML y diseño con Tailwind.**

**Rutas:**

* **Definidas en routes/web.php, conectan URLs con controladores.**

**Puesta en Marcha del Proyecto (Localmente)**

**Pasos Finales para ejecutar el sistema:**

1. **Abrir Visual Studio Code y el proyecto clase\_aut.**
2. **Abrir dos terminales:**
   * **Terminal 1: Laravel Artisan**
   * **Terminal 2: Node.js para Tailwind**

**3. Terminal 1 – Servidor Laravel:**

**en el CMD= php artisan serve**

**Acceder al sistema desde:** [**http://localhost:8000**](http://localhost:8000)

**4. Terminal 2 – Compilación de estilos con Tailwind:**

**en el CMD=**

**npm install**

**npm run dev**