

## Prácticas esenciales de ingeniería de software para el agilismo



# Instructivo de Instalación proyectos MISW4101 – Prácticas Esenciales de Agilismo

## Introducción

Este instructivo te guiará paso a paso para instalar y ejecutar las aplicaciones del curso. Este tutorial se realizará con el proyecto de gestión de concesionarios AutoHub, sin embargo, los pasos aquí descritos te servirán para instalar y ejecutar los demás proyectos. Sigue cada paso cuidadosamente.

## Requisitos Previos

Antes de comenzar, necesitas tener instalado en tu computadora:

### 1. Python 3.8 o superior

- **¿Cómo verificar si lo tienes?** Abre una terminal y escribe:

```
python --version
```

o

```
python3 --version
```

- **Si no lo tienes:** Descárgalo desde [python.org](https://python.org)

## 2. Node.js 18 o superior

- **¿Cómo verificar si lo tienes?** Abre una terminal y escribe:

```
node --version
```

- **Si no lo tienes:** Descárgalo desde [nodejs.org](https://nodejs.org)

## 3. Un editor de código

- Recomendamos [Visual Studio Code](#)



## Parte 1: Configuración del Backend (Servidor)

El backend es el servidor que maneja los datos. Corre en el puerto **5000**.

### Paso 1: Abrir la Terminal

#### En Windows:

- Presiona `Windows + R`, escribe `cmd` y presiona Enter

#### En Mac:

- Presiona `Cmd + Espacio`, escribe `terminal` y presiona Enter

#### En Linux:

- Presiona `Ctrl + Alt + T`

### Paso 2: Navegar a la Carpeta del Backend

Escribe el siguiente comando (ajusta la ruta según donde guardaste el proyecto):

```
cd ruta/a/tu/proyecto/Concesionario/backend
```

### Ejemplo en Windows:

```
cd C:\Users\TuNombre\Documentos\Concesionario\backend
```

### Ejemplo en Mac/Linux:

```
cd /Users/TuNombre/Documentos/Concesionario/backend
```

## Paso 3: Crear el Entorno Virtual

Un entorno virtual mantiene las dependencias del proyecto aisladas.

### En Windows:

```
python -m venv venv
```

### En Mac/Linux:

```
python3 -m venv venv
```



**Espera:** Este proceso puede tardar 1-2 minutos.

## Paso 4: Activar el Entorno Virtual

### En Windows:

```
venv\Scripts\activate
```

**En Mac/Linux:**

```
source venv/bin/activate
```

✅ **Verificación:** Deberías ver `(venv)` al inicio de tu línea de comandos.

## Paso 5: Instalar las Dependencias

```
pip install -r requirements.txt
```

⌚ **Espera:** Este proceso puede tardar 2-3 minutos.

## Paso 6: Ejecutar el Backend

```
python app.py
```

✅ **Verificación:** Deberías ver un mensaje como:

```
* Running on http://127.0.0.1:8000  
* Running on http://localhost:8000
```

🎉 **¡Listo!** El backend está corriendo. **NO CIERRES ESTA TERMINAL.**

---

## 🎨 Parte 2: Configuración del Frontend (Interfaz)

El frontend es la interfaz visual que verás en el navegador. Corre en el puerto **4200**.

## Paso 1: Abrir una NUEVA Terminal

**IMPORTANTE:** Necesitas una segunda terminal. El backend debe seguir corriendo en la primera.

- Abre otra ventana de terminal siguiendo los pasos de la Parte 1, Paso 1

## Paso 2: Navegar a la Carpeta del Frontend

```
cd ruta/a/tu/proyecto/Concesionario/frontend
```

### Ejemplo en Windows:


```
cd C:\Users\TuNombre\Documentos\Concesionario\frontend
```

### Ejemplo en Mac/Linux:

```
cd /Users/TuNombre/Documentos/Concesionario/frontend
```


## Paso 3: Instalar las Dependencias

```
npm install
```

 **Espera:** Este proceso puede tardar 3-5 minutos. Verás muchos mensajes en pantalla, es normal. Te recomendamos no realizar o aplicar los ajustes sugeridos, pues tendrías que realizar más cambios los cuales no son necesarios.

## Paso 4: Ejecutar el Frontend

```
npm start
```

 **Espera:** La primera vez puede tardar 1-2 minutos en compilar.

✅ **Verificación:** Deberías ver un mensaje como:

```
** Angular Live Development Server is listening on localhost:4200 **
```

🎉 ¡**Listo!** El frontend está corriendo. **NO CIERRES ESTA TERMINAL.**

---

## Parte 3: Acceder a la Aplicación

### Paso 1: Abrir el Navegador

Abre tu navegador favorito (Chrome, Firefox, Edge, Safari)

### Paso 2: Ir a la Aplicación

Escribe en la barra de direcciones:

```
http://localhost:4200
```



✅ ¡**Felicidades!** Deberías ver la aplicación AutoHub funcionando.

---

## Verificación Final

Asegúrate de que todo funcione correctamente:

1. ✅ **Terminal 1:** Backend corriendo en puerto 8000
2. ✅ **Terminal 2:** Frontend corriendo en puerto 4200

3.  **Navegador:** Aplicación visible en <http://localhost:4200>
  4.  **Funcionalidad:** Puedes ver la lista de vehículos y marcas
- 

## Cómo Detener la Aplicación

Cuando termines de trabajar:

1. **En cada terminal** presiona: ``Ctrl + C``
  2. **En Windows (backend):** Escribe ``deactivate`` para salir del entorno virtual
  3. **Cierra las terminales**
- 

## Cómo Volver a Ejecutar la Aplicación

La próxima vez que quieras usar la aplicación:

### Backend (Terminal 1):

```
cd ruta/a/tu/proyecto/Concesionario/backend
# Windows: venv\Scripts\activate
# Mac/Linux: source venv/bin/activate
python app.py
```

### Frontend (Terminal 2):

```
cd ruta/a/tu/proyecto/Concesionario/frontend
npm start
```

---

## ! Solución de Problemas Comunes

### Problema 1: "python no se reconoce como comando"

**Solución:** Intenta usar ``python3`` en lugar de ``python``

### Problema 2: "npm no se reconoce como comando"

**Solución:** Node.js no está instalado correctamente. Reinstálalo desde [nodejs.org](https://nodejs.org)

### Problema 3: "Puerto 8000 ya está en uso"

**Solución:**

1. Abre ``backend/app.py``
2. Cambia la línea final a: ``app.run(debug=True, port=8080)``
3. Actualiza los servicios del frontend para usar puerto 8080

### Problema 4: "Error al cargar datos en el navegador"

**Solución:**

1. Verifica que el backend esté corriendo (Terminal 1)
2. Abre las herramientas de desarrollador del navegador (F12)
3. Revisa la consola en busca de errores

### Problema 5: La página no carga después de ``npm start``

**Solución:**

1. Espera 2 minutos completos
2. Refresca el navegador (F5)
3. Si persiste, detén con Ctrl+C y ejecuta ``npm start`` nuevamente



## Problema 6: "Permission denied" en Mac/Linux

**Solución:** Agrega ``sudo`` antes del comando:

```
sudo npm install
```

## Problema 7: "ERR\_UNSAFE\_PORT" en el navegador

**Solución:** El puerto configurado no es seguro. Verifica que:

1. El backend esté corriendo en puerto 8000 (no 6000)
  2. Los servicios del frontend apunten a ``http://localhost:8000``
- 



## Recursos Útiles

- **Documentación de Python:** <https://docs.python.org/es/3/>
  - **Documentación de Node.js:** <https://nodejs.org/docs/>
  - **Documentación de Angular:** <https://angular.io/docs>
  - **Documentación de Flask:** <https://flask.palletsprojects.com/>
- 



## Notas Importantes

- **Dos terminales:** Siempre necesitas DOS terminales abiertas simultáneamente (una para backend, otra para frontend)
- **No cerrar terminales:** Mientras uses la aplicación, ambas terminales deben permanecer abiertas
- **Orden de ejecución:** Primero inicia el backend, luego el frontend
- **Datos mock:** Esta aplicación usa datos de prueba. Como estudiante, deberás implementar la persistencia real