# Instalar proyecto Activo Fijo

Se han enviado 2 archivos junto con este documento:

1. afijo\_v0.1.zip
2. afijo\_dumped\_20210719.json

## Requisitos

Independiente del S.O. donde se requiera instalar, es necesario:

1. PYTHON 3.8 o superior
2. DJANGO 2.2: se desarrolló para esta versión dado que es la que SYSTECH tiene en su servidor actual.
3. POSTGRESQL 12

Opcionalmente si se desea continuar con el desarrollo se recomienda instalar Visual Studio Code, con los plugins para Python y Django.

## Instalar fuentes

Crear un directorio de trabajo y descomprimir el archivo afijo\_v0.1.zip:

|  |
| --- |
| mkdir workspace  cd workspace  unzip <rutda\_descarga>/afijo\_v0.1.zip |

Esto creará los siguientes archivos y carpetas:

|  |
| --- |
| afijo/  manage.py  requirements.txt |

Parado en este misma se requiere instalar todas las bibliotecas que el sistema utiliza, para esto ejecutar el comando:

|  |
| --- |
| pip install -r requirements.txt |

El comando PIP es el manejador de paquetes de PYTHON y se instala por defecto junto con PYTHON.

**IMPORTANTE:**

Para Instalar en producción simplemente ejecutar el comando de descompresión y PIP en el directorio actual donde se encuentra DJANGO de producción.

## Exportar e importar datos del proyecto

Django proporciona dos comandos de administración para este propósito:

1. dumpdata
2. loaddata

### Comando: Dumpdata

El comando 'dumpdata' vuelca los datos en formato JSON. Estos datos JSON se pueden exportar a cualquier proyecto de Django independientemente de la base de datos detrás, sin embargo la base utilizada en nuestro caso es POSTGRES.

Para nuestro caso se utilizó la opción ‘--indent’ para obtener un JSON legible:

|  |
| --- |
| python manage.py dumpdata --indent=4 > afijo\_dumped\_20210719.json |

### Comando: Loaddata

El comando de administración 'loaddata' carga los datos que acabamos de exportar.

Aquí está la sintaxis para el archivo que se ha enviado como adjunto:

|  |
| --- |
| python manage.py loaddata < afijo\_dumped\_20210719.json |

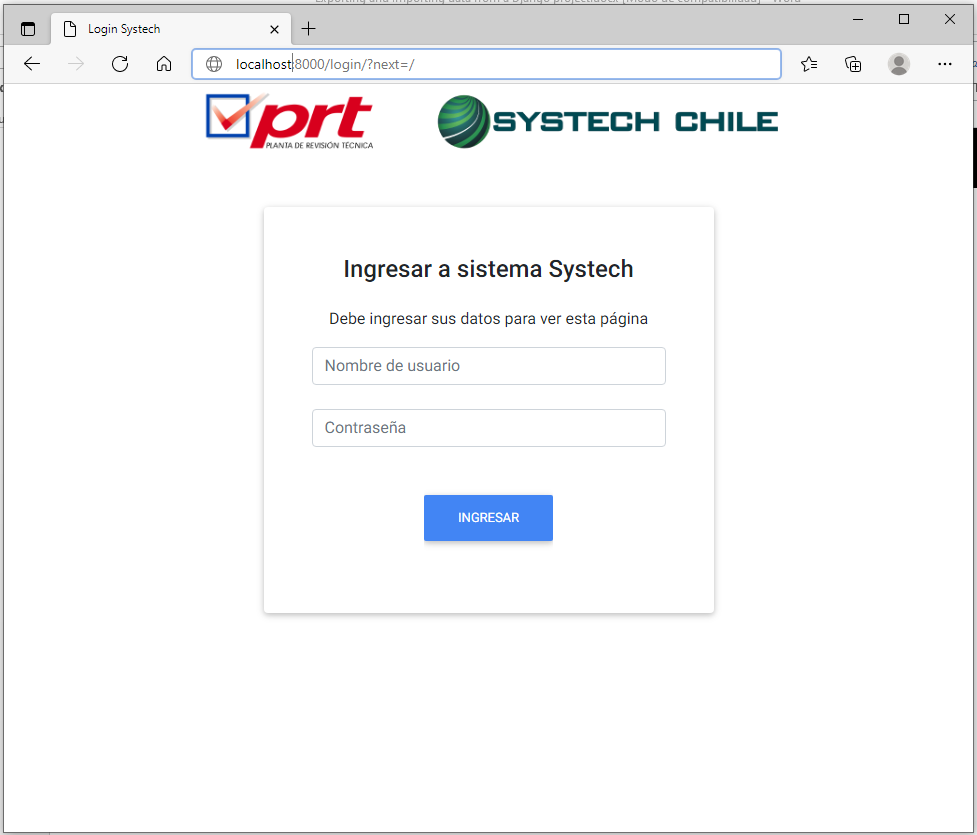
## Verificar

Para verificar el correcto funcionamiento se recomienda arrancar el servicio local de DJANGO usando el siguiente comando para do en el mismo directorio donde se descomprimió:

|  |
| --- |
| python manage.py runserver 0:8000 |

El puerto 8000 es sugerido, se puede utilizar otro si es necesario.

Ahora se puede abrir el navegador y debería aparece el siguiente formulario:



Para ingresar usar el usuario **‘desa’** con contraseña **‘desa’**.