

Configuración de proyecto con **Plugcore**



Plug Soluciones TIC S.L.
info@plugcore.com



<https://github.com/plugcore/plugcore>



<https://www.linkedin.com/company/plugcore>



<https://twitter.com/plugcoreit>

Configuración en plugcore

- El sistema de configuración de plugcore se basa en unos archivos JSON que se leerán al iniciarse el proyecto.
- Esta configuración servirá tanto para configurar otras partes de plugcore, como por ejemplo los logs o las conexiones a las bases de datos.
- Pero también está pensada para ser extendida y utilizada para la configuración propia del proyecto que se esté creando.
- De momento no se puede cambiar los valores en tiempo de ejecución, pero se está preparando un sistema para el futuro.



Ejecución

- El sistema de configuración en plugcore se ejecuta en 3 pasos
 - **1 - Carpeta:** Primero se tiene que ver dónde está la carpeta que tiene la configuración, por defecto será la carpeta “./configuration/” si es un proyecto hecho con el cli, pero se puede cambiar.
 - **2 - Entorno:** Ahora se revisa la variable de entorno “NODE_ENV” para ver que archivos de configuración vamos a leer.
 - **3 - Resolución:** Ahora que ya tenemos los archivos, vamos a cargarlos y a resolver todas sus dependencias, como por ejemplo variables de entorno o referencias a otros archivos.



1 - Carpeta de configuración

- Por defecto la carpeta donde leerán los archivos de configuración es “./configuration”
- Esta ruta se puede cambiar en el comando de inicio de un proyecto hecho en plugcore, por defecto es “`npx plug start`” pero podemos usar la versión extendida del comando:
 - `npx plug start [DIST FOLDER] [CONFIGURATION FOLDER]`: Es decir, el segundo parámetro es la carpeta de configuración.
 - Ejemplo de ejecución: `npx plug start dist/src otherconfiguration`



2 - Entorno

- Dentro de la carpeta de configuración lo primero que se va a buscar es el archivo **"configuration.json"**
- En este archivo tendremos el objeto de configuración base que se va a utilizar siempre independientemente del entorno en el que nos encontremos.
- Después se leerá la variable de entorno "NODE_ENV", y si por ejemplo tiene el valor de "pro", se buscará a ver si hay el archivo "configuration.pro.json".
- Si existe el archivo, entonces se sobrescribirán todas las claves definidas en este archivo en la configuración.



3 - Resolución

- Además de los valores normales que podemos definir en un json (string, boolean, number, etc.) podemos usar algunos valores especiales para importar variables de entorno o otros archivos:
 - `${VARIABLE_ENTORNO}`: Para definir una variable de entorno
 - `@[import:RUTA_ARCHIVO]` Para importar un JSON y que sea el valor de esta propiedad, útil si por ejemplo tenemos algunas partes de la configuración muy grandes.

- Ejemplo:

```
"custom": {  
  "exampleWithEnv": "${NODE_ENV}",  
  "exampleImport": "@[import:example-import.json]"  
}
```



Usar la configuración

- Una vez dentro de nuestro proyecto, es recomendable crearse una interfaz que nos permita definir la estructura de la configuración que queremos usar, por ejemplo en "src/configuration/configuration.interface.ts"
- Y después, podemos importar esa configuración en cualquier servicio de la siguiente manera:

```
constructor(  
    // CustomConfiguration es la interfaz que hemos definido antes  
    @InjectConfiguration() private config: CustomConfiguration  
) {  
    console.log(this.config.custom.exampleCustomProp);  
}
```



GRACIAS POR SU TIEMPO

Para más información

info@plugcore.com

germanml@plugcore.com

sergiolc@plugcore.com



Diario de Mallorca

Última Hora

#EMPRENBIT

