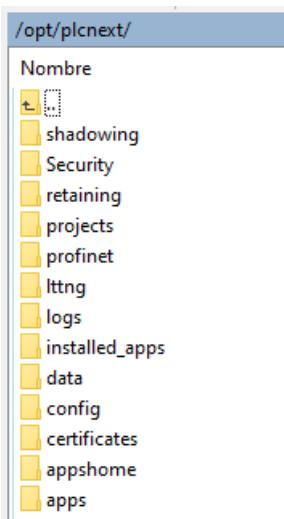


## Creación de App para PlcNext basada en proyecto de PlcNext Engineer

Debemos partir de un proyecto de PlcNext Engineer (control + visualización) compilado correctamente y descargado al Plc.

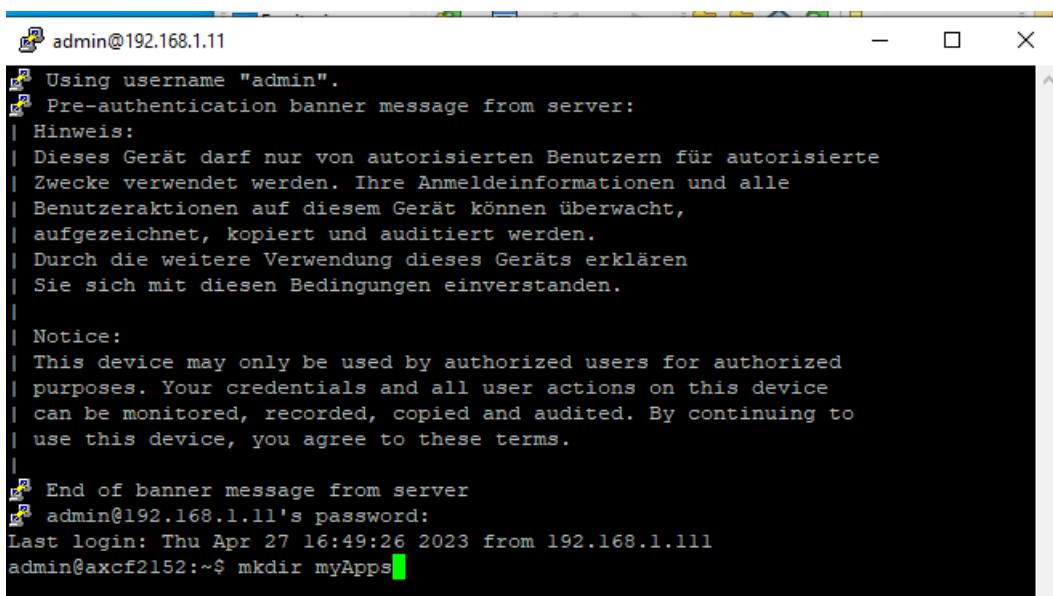
Comprobaremos que todo está correcto y el Plc está funcionando correctamente.

Acceder mediante WinSCP (por ejemplo) a la ruta /opt/plcnext /del controlador.



Abrir una sesión de consola con Putty (por ejemplo) y ejecutar el comando:

mkdir myApps (en la ruta en la que estamos: /opt/plcnext/)



```
Using username "admin".
Pre-authentication banner message from server:
| Hinweis:
| Dieses Gerät darf nur von autorisierten Benutzern für autorisierte
| Zwecke verwendet werden. Ihre Anmeldeinformationen und alle
| Benutzeraktionen auf diesem Gerät können überwacht,
| aufgezeichnet, kopiert und audited werden.
| Durch die weitere Verwendung dieses Geräts erklären
| Sie sich mit diesen Bedingungen einverstanden.

| Notice:
| This device may only be used by authorized users for authorized
| purposes. Your credentials and all user actions on this device
| can be monitored, recorded, copied and audited. By continuing to
| use this device, you agree to these terms.

End of banner message from server
admin@192.168.1.11's password:
Last login: Thu Apr 27 16:49:26 2023 from 192.168.1.111
admin@axcf2152:~$ mkdir myApps
```

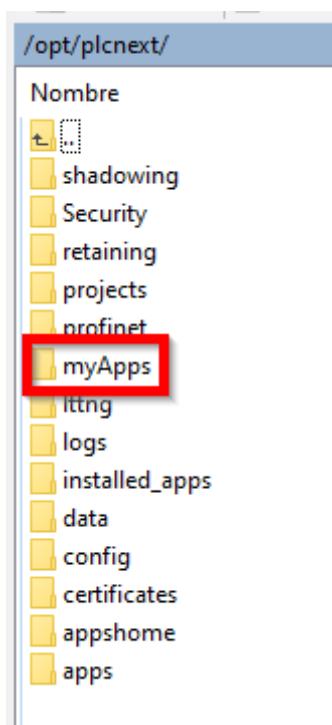
Ejecutar ahora:



cp -r projects/PCWE myApps

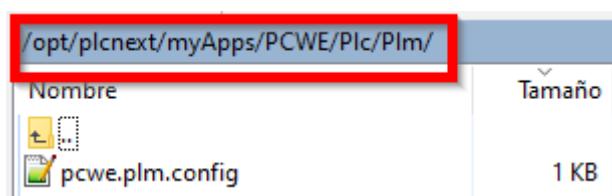
```
admin@axcf2152:~$ cp -r projects/PCWE myApps
admin@axcf2152:~$
```

Se ha creado el directorio myApps



Dentro de ese directorio vamos a la ruta:

/opt/plcnext/myApps/PCWE/Plc/Plm/



y editamos el fichero pcwe.plm.config .

Si dentro del mismo vemos alguna entrada similar a esta:

```
#!/opt/plcnext/myApps/PCWE/Plc/Plm/pcwe.plm.config - admin@192.168.1.11 - Editor - WinSCP
Codificación ▾ Color ▾
<?xml version="1.0" encoding="utf-8"?>
<PlmConfigurationDocument xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" schemaVersion="1.0">
  <Libraries>
    <Library name="PLCnextBaseServices" binaryPath="$ARP_PROJECTS_DIR$/PCWE/Libs/PLCnextBaseServices/libPLCnextBaseServices.so">
      <Dependencies />
    </Library>
  </Libraries>
  <Components>
    <Component name="ServiceProviderComponent1" type="PLCnextBaseServices.ServicesProviderComponent" library="PLCnextBaseServices">
      <Settings path="" />
    </Component>
  </Components>
</PlmConfigurationDocument>
```



Deberemos sustituir el original \$ARP\_PROJECTS\_DIR\$/PCWE/ por \$ARP\_PROJECT\_CURRENT\_DIR\$/

De esta forma el fichero quedará:

The screenshot shows a WinSCP Editor window displaying an XML configuration file. The file path is /opt/plcnext/myApps/PCWE/Plc/Plm/pcwe.plm.config. The code is as follows:

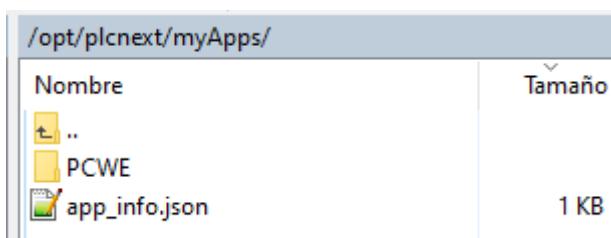
```
<?xml version="1.0" encoding="utf-8"?>
<PlmConfigurationDocument xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" schemaVersion="1.0">
  <Libraries>
    <Library name="PLCnextBaseServices" binaryPath="$ARP_PROJECT_CURRENT_DIR$/Libs/PLCnextBaseServices/libPLCnextBaseServices.so">
      <Dependencies />
    </Library>
  </Libraries>
  <Components>
    <Component name="ServiceProviderComponent1" type="PLCnextBaseServices.ServicesProviderComponent" library="PLCnextBaseServices">
      <Settings path="" />
    </Component>
  </Components>
</PlmConfigurationDocument>
```

The line <binaryPath="\$ARP\_PROJECT\_CURRENT\_DIR\$/Libs/PLCnextBaseServices/libPLCnextBaseServices.so"> is highlighted with a red box.

Necesitamos un fichero en formato json con la descripción de la app que estamos creando. Se ha de llamar app\_info.json y ha de ir colocado en la ruta /opt/plcnext/myApps/. El formato es este:

```
{
  "plcnextapp": {
    "name": "MiPrimeraApp",
    "identifier": "0000000000000000",
    "version": "2.0",
    "target": "AXC F 2152",
    "minfirmware_version": "22.0.4",
    "manufacturer": "Phoenix Contact",
    "licensetype": "Free"
  },
  "engineerapp": {
    "folder": "/PCWE/"
  }
}
```

Indicaremos el nombre de nuestra app, un número (mantener los 14 dígitos), la versión de nuestra app, el controlador para el que está preparada, la revisión mínima de fw que ha de tener ese controlador, el nombre del creador de la app, tipo de licencia. Del resto no se ha de modificar nada.

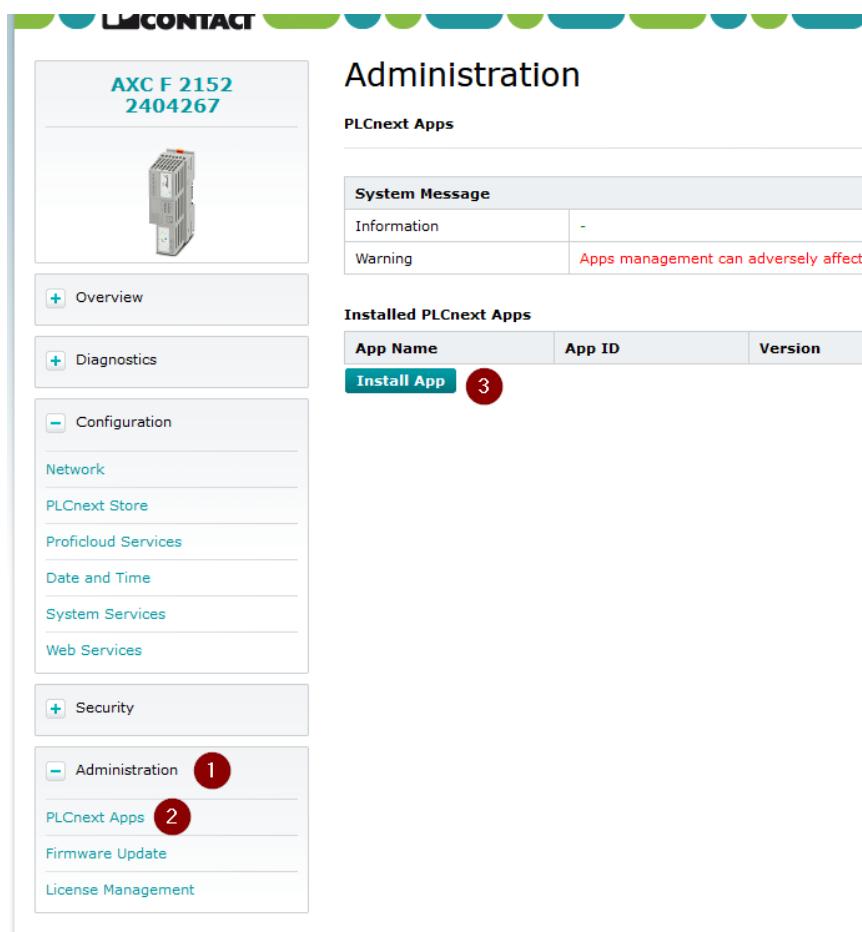


Ejecutaremos ahora el comando: `plcnextapp create myApps MiPrimeraApp`

La app estará creada en `/opt/plcnext`

/opt/plcnext/		
Nombre	Tamaño	M
..	3	
shadowing	1	
Security	0	
retaining	2	
projects	1	
myApps	1	
Itng	0	
logs	1	
installed_apps	1	
data	1	
config	2	
appshome	1	
apps	1	
<b>MiPrimeraApp.app</b>	<b>2.812 KB</b>	<b>1</b>
PLCnextBase.ServicesProvider.log	4 KB	1

Ya podremos instalarla en cualquier PlcNext.



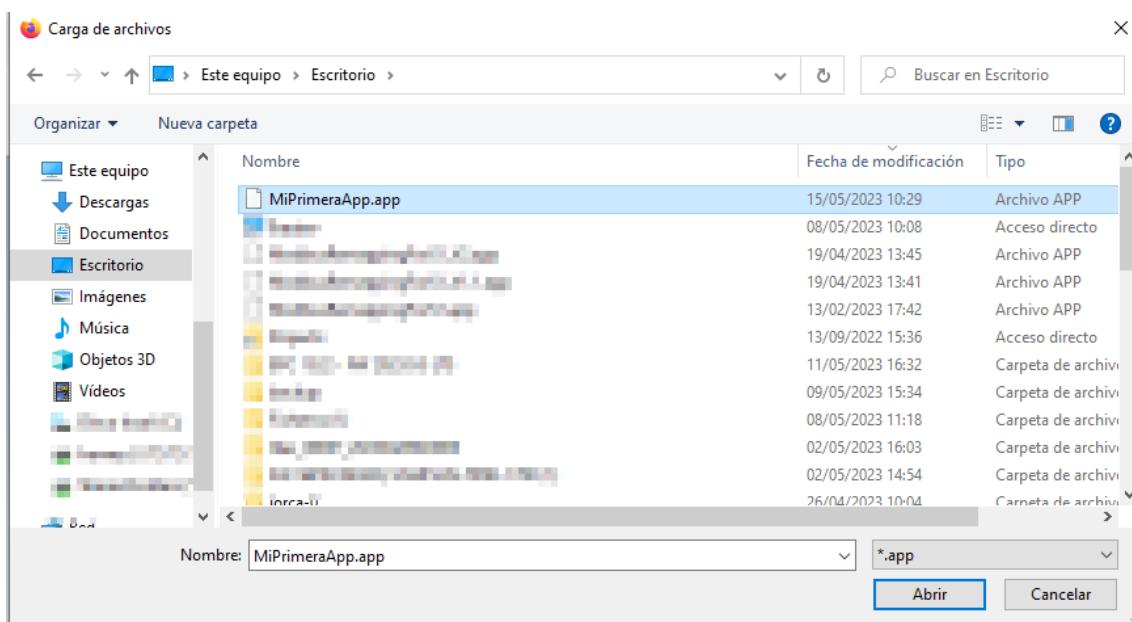
The screenshot shows the PLCnext Administration interface. On the left, there is a sidebar with various navigation options:

- AXC F 2152 2404267 (highlighted)
- Overview
- Diagnostics
- Configuration
- Network
- PLCnext Store
- Profilcloud Services
- Date and Time
- System Services
- Web Services
- Security
- Administration (highlighted, circled with '1')
- PLCnext Apps (highlighted, circled with '2')
- Firmware Update
- License Management

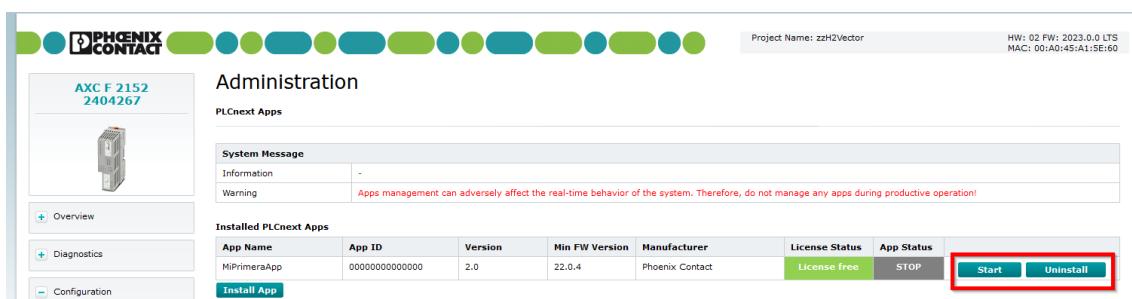
The main content area is titled "Administration" and contains the following sections:

- PLCnext Apps**
- System Message** (Information: -, Warning: Apps management can adversely affect t)
- Installed PLCnext Apps** (Table):
 

App Name	App ID	Version
MiPrimeraApp	3	
- Install App** (highlighted, circled with '3')



Ahora ya podremos arrancar la app, detenerla o desinstalarla:



App Name	App ID	Version	Min FW Version	Manufacturer	License Status	App Status
MiPrimeraApp	0000000000000000	2.0	22.0.4	Phoenix Contact	License free	STOP