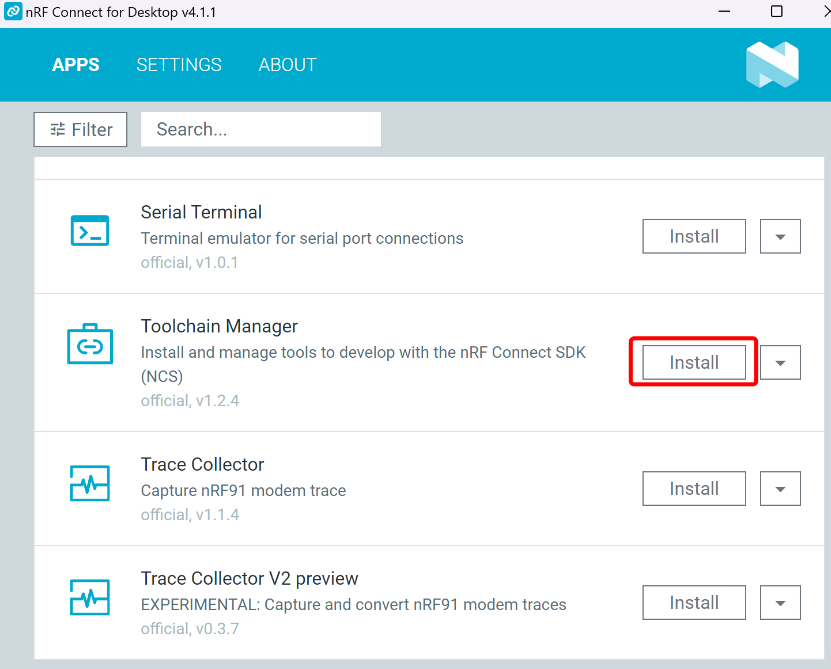
XIAO SENSE BUILD STEPS

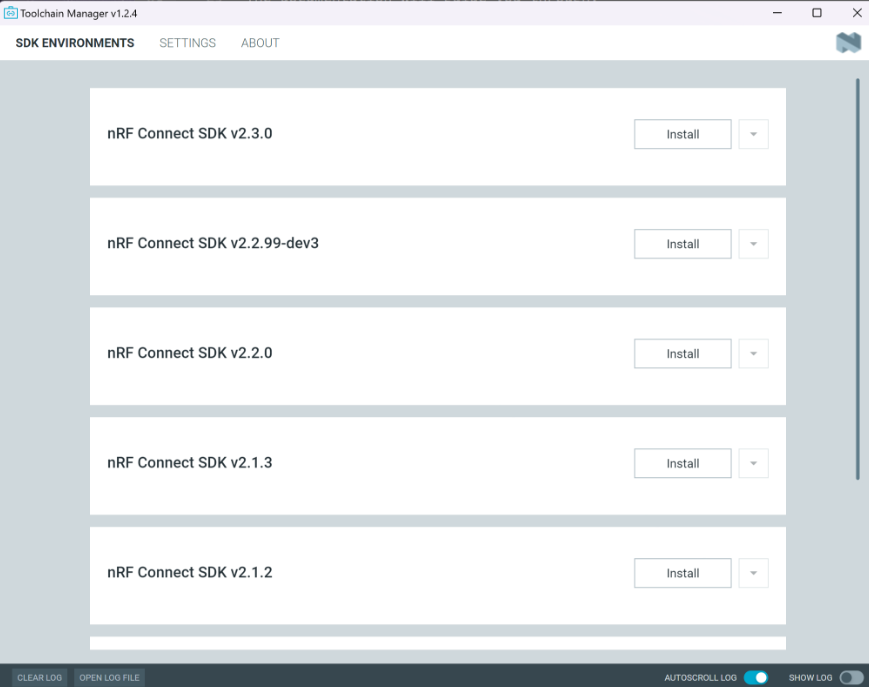
* Install prerequisites based on the following link

<https://developer.nordicsemi.com/nRF_Connect_SDK/doc/2.5.0/nrf/installation/recommended_versions.html>.

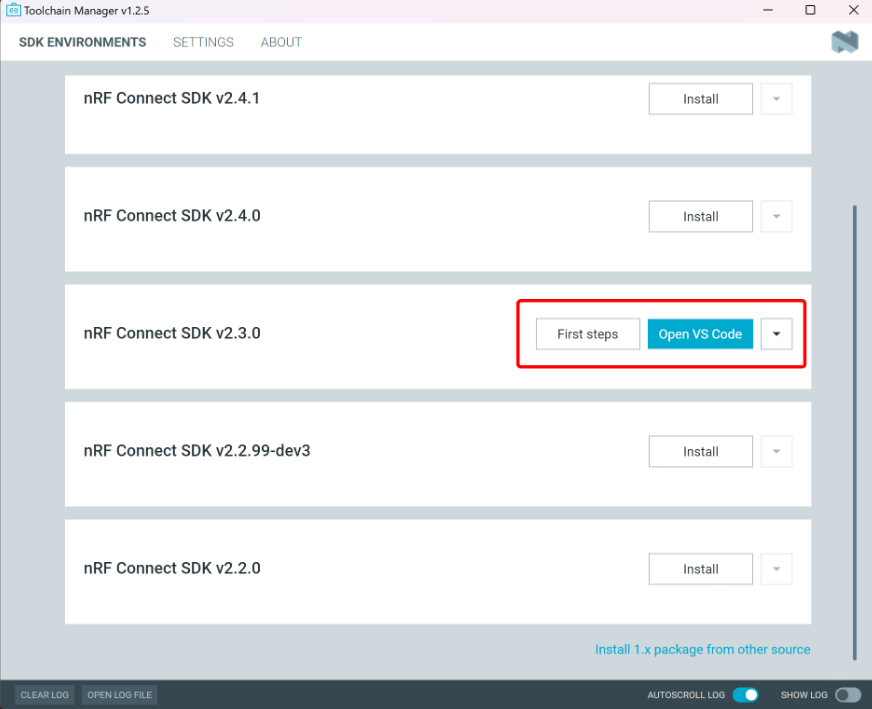
* Download [nRF Connect for Desktop](https://www.nordicsemi.com/Software-and-Tools/Development-Tools/nRF-Connect-for-desktop) for your operating system(Ubuntu 20.04 LTS)
* Install and run the tool on your machine
* In the APPS section, click Install next to Toolchain Manager



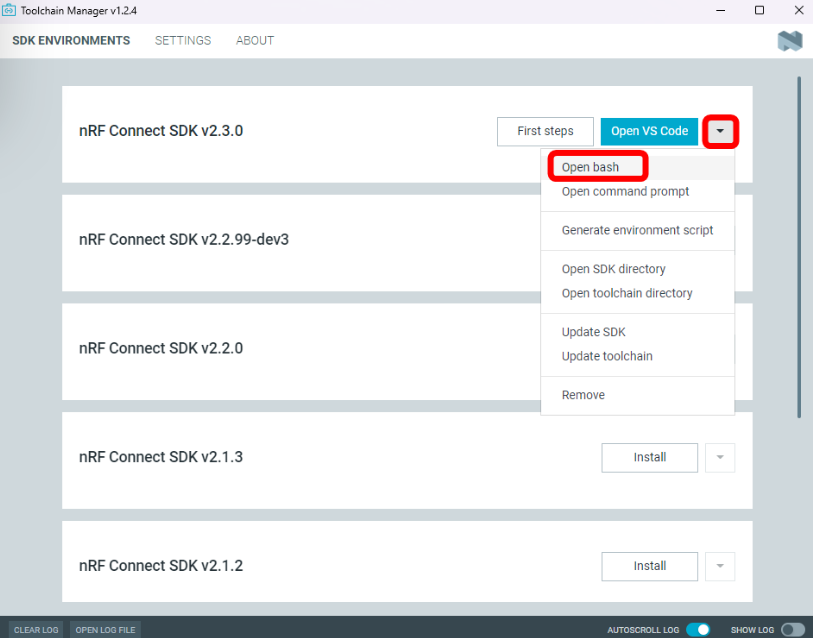
* The app is installed on your machine, and the **Install** button changes to **Open**
* Open Toolchain Manager in nRF Connect for Desktop.  
  Click **SDK ENVIRONMENTS** in the navigation bar to specify where you want to install the nRF Connect SDK



* In SDK ENVIRONMENTS, click the Install button next to the nRF Connect SDK version (2.3.0) that you want to install
* The nRF Connect SDK version of your choice is installed on your machine. The Install button changes to Open VS Code



* For our project, the use of the command line is sufficient. Click the down arrow next to the version you installed, and select **Open bash**

****

* Clone the [Amazon Sidewalk application repository](https://github.com/nrfconnect/sdk-sidewalk) from the sdk-sidewalk repository to the nRF Connect SDK folder, and name it sidewalk by running the following command:  
    
  **git clone https://github.com/nrfconnect/sdk-sidewalk.git sidewalk**
* **pip install -r sidewalk/requirements.txt**
* Create a folder **internal\_sidewalk.** copy the file **sidewalk/internal\_west.yml** to **internal\_sidewalk.**
* Rename **internal\_west.yml** to **west.yml**
* Check the current manifest path:

**west manifest --path**

* Set the manifest path

**west config manifest.path internal\_sidewalk**

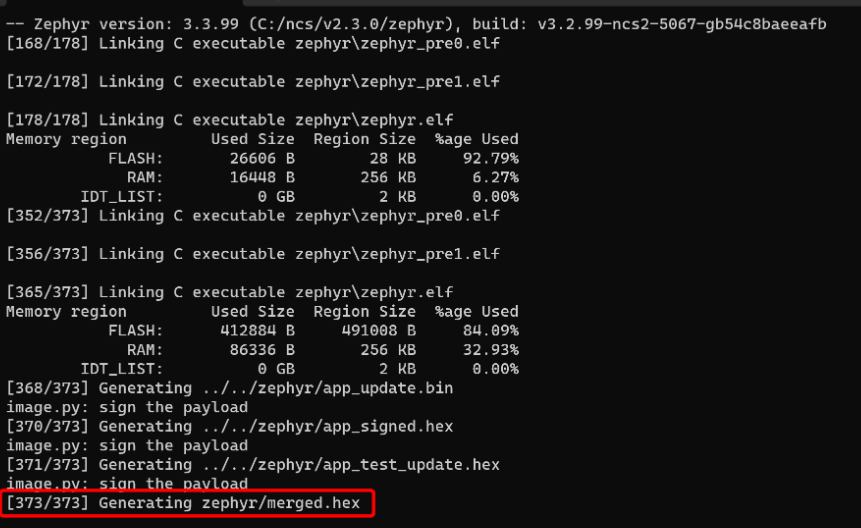
* **west config manifest.group-filter "+sidewalk"**
* **west update**
* Download the program written for the XIAO nRF52840 from Github  
  <https://github.com/limengdu/SeeedStudio-XIAO-nRF52840-sidewalk/releases/tag/v1.0.0>
* Once downloaded, please unzip the folder into the ncs directory. The default path is as follows

**ncs\v2.3.0\zephyr\boards\arm\nrf52840dk\_nrf52840**

* Enter the following command to perform the firmware generation  
  **cd sidewalk/samples/template\_ble/**

**west build -b nrf52840dk\_nrf52840**

If the execution goes well, you will see the following output



At this point we've got the sample firmware, which is named: **merged.hex** and stored in the default location of:

**ncs\v2.3.0\sidewalk\samples\template\_ble\build\zephyr**

**Provision binary build steps**

* Configure *credentials* file on your local machine

If you haven't already installed the [AWS CLI](https://aws.amazon.com/cli/), then you may need to install it.

* Execute the command  
  **aws configure**
* The window will then ask you to enter your keys, please enter them separately and enter to confirm.

**aws\_access\_key\_id = YOUR\_ACCESS\_KEY**

**aws\_secret\_access\_key = YOUR\_SECRET\_KEY**

**region=us-east-1**

For the rest, we simply enter and leave the default.

Note : The aws **Iam user account** should have the admin privilege

* Download the Amazon Sidewalk Sample IoT App Repository to your local machine  
  **git clone** [**https://github.com/aws-samples/aws-iot-core-for-amazon-sidewalk-sample-app.git**](https://github.com/aws-samples/aws-iot-core-for-amazon-sidewalk-sample-app.git)
* Open the folder aws-iot-core-for-amazon-sidewalk-sample-app
* Populate the **config.yaml** configuration file. Set **NORDIC** hardware platform.
* Set up Python virtual environment for the provisioning tools:  
    
  **cd aws-iot-core-for-amazon-sidewalk-sample-app  
  sudo apt install python3-venv**

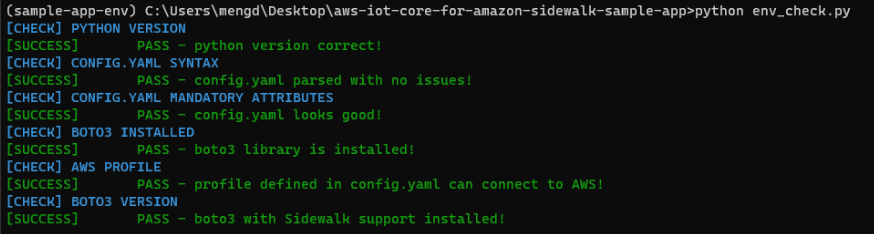
**python3 -m venv sample-app-env  
source sample-app-env/bin/activate**

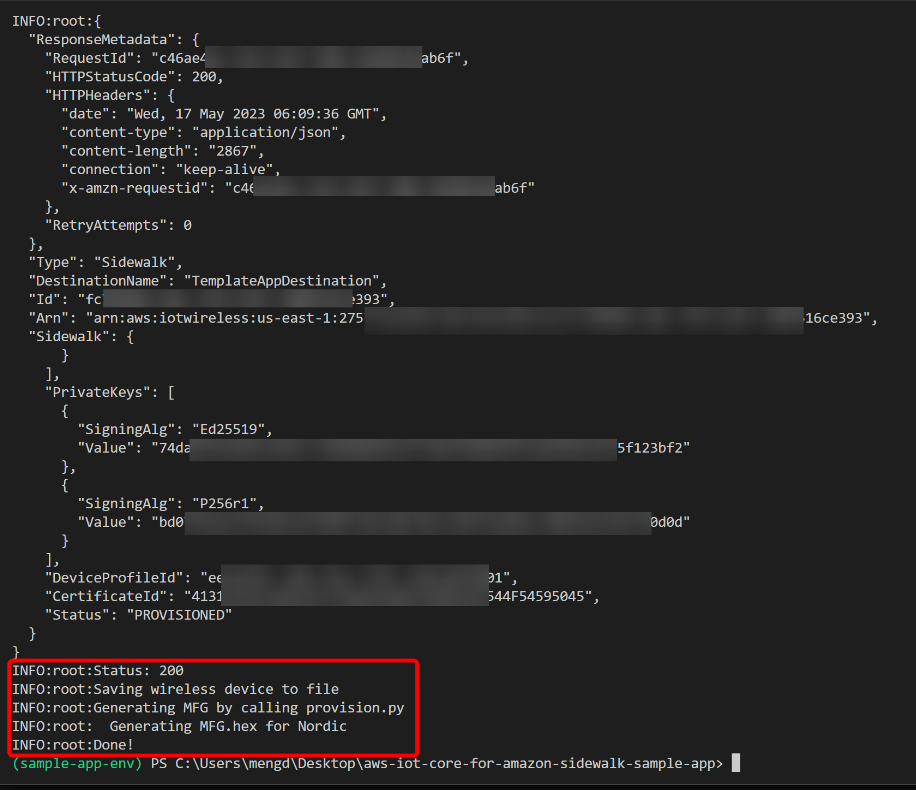
**pip install pip==22.3.1**

**python -m pip install -r requirements.txt**

**python -m pip install pyjwt -t ./ApplicationServerDeployment/lambda/authLibs**

* Sanity check  
  **python env\_check.py**

If the message appears as shown, then your environment installation has gone well.  
  


* Run the device provisioning script  
    
  **python3 EdgeDeviceProvisioning/provision\_sidewalk\_end\_device.py  
    
  **
* Exit python environment  
  **deactivate**