

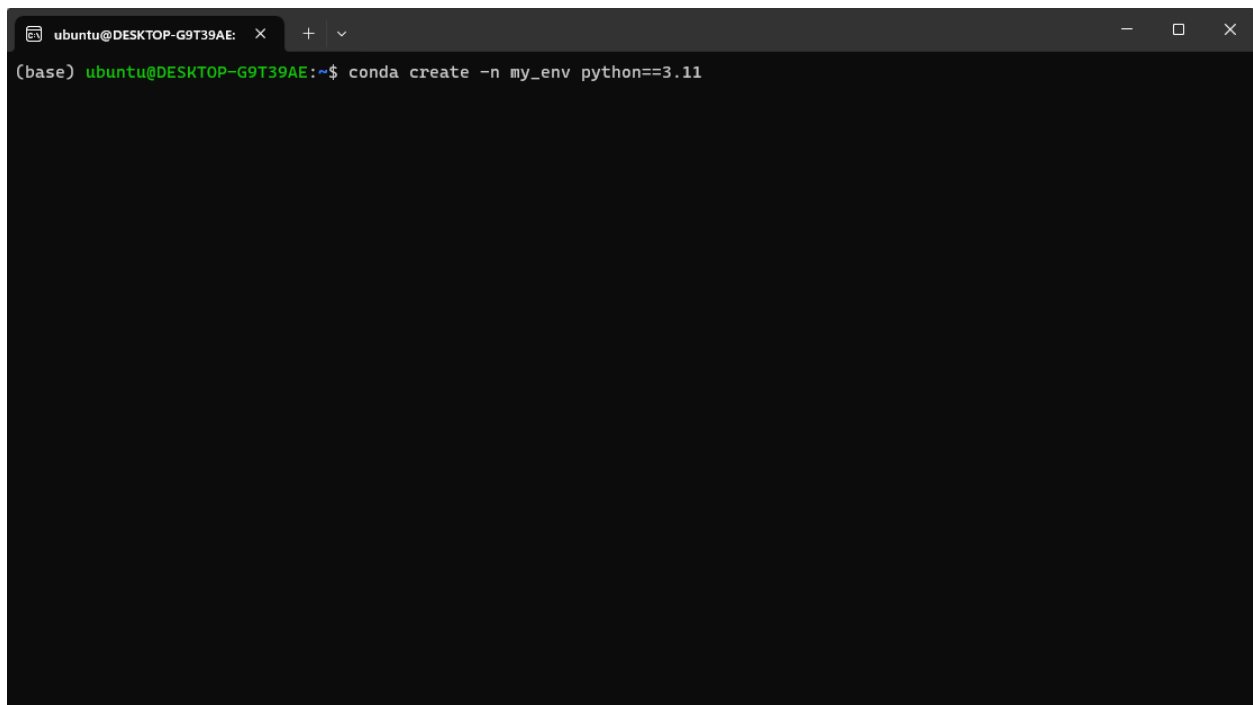
## Build Mobile Application With Kivy Framework

\*\*\* ใช้ WSL (Ubuntu) terminal ในการใช้งาน Buildozer

สร้างแอปสำหรับ Scan Bluetooth Devices

ขั้นตอนการสร้าง Application ด้วย Kivy Framework

1. สร้าง Conda Environment ขึ้นมาก่อน
  - `conda create -n <ENV_NAME> python==<VERSION>`
  - ตัวอย่าง `conda create -n my_env python==3.11`

A screenshot of a terminal window with a dark background. The window title bar shows 'ubuntu@DESKTOP-G9T39AE: X' and standard window controls. The terminal prompt is '(base) ubuntu@DESKTOP-G9T39AE:~\$'. The command 'conda create -n my\_env python==3.11' has been entered and is visible on the line.

```
(base) ubuntu@DESKTOP-G9T39AE:~$ conda create -n my_env python==3.11
```

```
ubuntu@DESKTOP-G9T39AE: X + v
- python==3.11

The following NEW packages will be INSTALLED:

_libgcc_mutex      pkgs/main/linux-64::_libgcc_mutex-0.1-main
_openmp_mutex      pkgs/main/linux-64::_openmp_mutex-5.1-1_gnu
bzip2              pkgs/main/linux-64::bzip2-1.0.8-h7b6447c_0
ca-certificates    pkgs/main/linux-64::ca-certificates-2023.12.12-h06a4308_0
ld_impl_linux-64   pkgs/main/linux-64::ld_impl_linux-64-2.38-h1181459_1
libffi             pkgs/main/linux-64::libffi-3.4.4-h6a678d5_0
libgcc-ng          pkgs/main/linux-64::libgcc-ng-11.2.0-h1234567_1
libgomp            pkgs/main/linux-64::libgomp-11.2.0-h1234567_1
libstdcxx-ng       pkgs/main/linux-64::libstdcxx-ng-11.2.0-h1234567_1
libuuid            pkgs/main/linux-64::libuuid-1.41.5-h5eee18b_0
ncurses            pkgs/main/linux-64::ncurses-6.4-h6a678d5_0
openssl            pkgs/main/linux-64::openssl-1.1.1w-h7f8727e_0
pip               pkgs/main/linux-64::pip-23.3.1-py311h06a4308_0
python            pkgs/main/linux-64::python-3.11.0-h7a1cb2a_3
readline          pkgs/main/linux-64::readline-8.2-h5eee18b_0
setuptools        pkgs/main/linux-64::setuptools-68.2.2-py311h06a4308_0
sqlite            pkgs/main/linux-64::sqlite-3.41.2-h5eee18b_0
tk                pkgs/main/linux-64::tk-8.6.12-h1ccaba5_0
tzdata            pkgs/main/noarch::tzdata-2024a-h04d1e81_0
wheel             pkgs/main/linux-64::wheel-0.41.2-py311h06a4308_0
xz                pkgs/main/linux-64::xz-5.4.5-h5eee18b_0
zlib              pkgs/main/linux-64::zlib-1.2.13-h5eee18b_0

Proceed ([y]/n)? y
```

```
ubuntu@DESKTOP-G9T39AE: X + v

pip                pkgs/main/linux-64::pip-23.3.1-py311h06a4308_0
python            pkgs/main/linux-64::python-3.11.0-h7a1cb2a_3
readline          pkgs/main/linux-64::readline-8.2-h5eee18b_0
setuptools        pkgs/main/linux-64::setuptools-68.2.2-py311h06a4308_0
sqlite            pkgs/main/linux-64::sqlite-3.41.2-h5eee18b_0
tk                pkgs/main/linux-64::tk-8.6.12-h1ccaba5_0
tzdata            pkgs/main/noarch::tzdata-2024a-h04d1e81_0
wheel             pkgs/main/linux-64::wheel-0.41.2-py311h06a4308_0
xz                pkgs/main/linux-64::xz-5.4.5-h5eee18b_0
zlib              pkgs/main/linux-64::zlib-1.2.13-h5eee18b_0

Proceed ([y]/n)? y

Downloading and Extracting Packages:

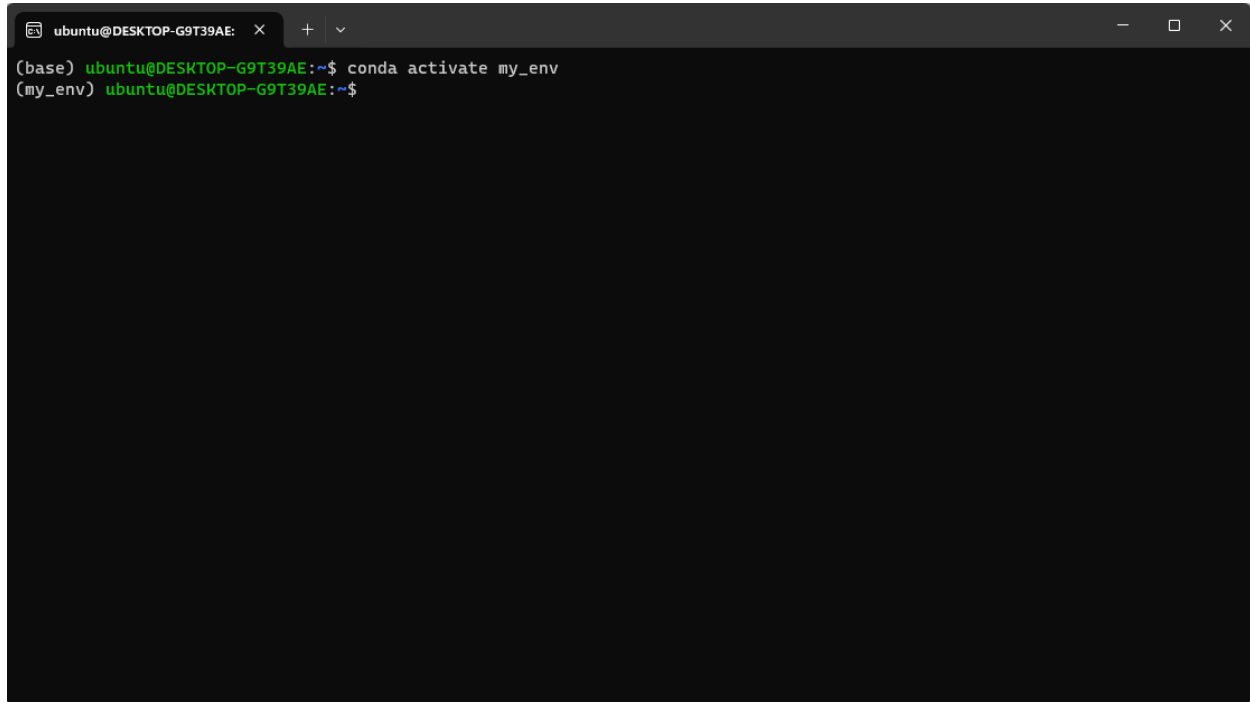
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#   $ conda activate my_env
#
# To deactivate an active environment, use
#
#   $ conda deactivate

(base) ubuntu@DESKTOP-G9T39AE:~$
```

จะได้หน้าต่างประมาณนี้

2. เข้าใช้งาน Conda Env ที่สร้างขึ้นมา หรือ ต้องการจะใช้งาน

- conda activate <ENV\_NAME>
- ตัวอย่าง conda activate my\_env

A terminal window with a dark background and light green text. The window title bar shows 'ubuntu@DESKTOP-G9T39AE:'. The terminal content shows the command 'conda activate my\_env' being executed, which changes the prompt from '(base)' to '(my\_env)'.

```
ubuntu@DESKTOP-G9T39AE: ~  
(base) ubuntu@DESKTOP-G9T39AE:~$ conda activate my_env  
(my_env) ubuntu@DESKTOP-G9T39AE:~$
```

### 3. ติดตั้ง Python Libraries ที่ต้องใช้งาน

- pip install <LIB\_NAME> or pip install <LIB\_NAME>==<VERSION>
- ตัวอย่าง pip install kivy

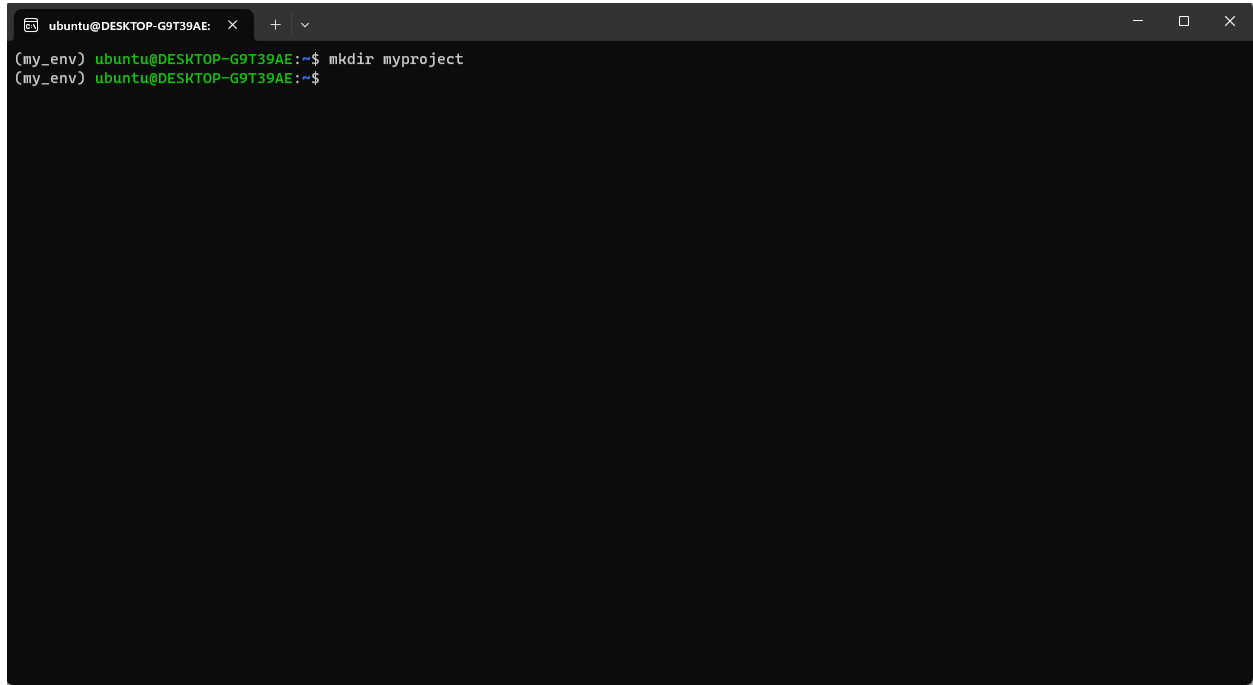
```
ubuntu@DESKTOP-G9T39AE: ~$ conda activate my_env
(my_env) ubuntu@DESKTOP-G9T39AE: ~$ pip install kivy
Collecting kivy
  Using cached Kivy-2.3.0-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (15 kB)
Collecting Kivy-Garden==0.1.4 (from kivy)
  Using cached Kivy_Garden-0.1.5-py3-none-any.whl.metadata (159 bytes)
Collecting docutils (from kivy)
  Using cached docutils-0.20.1-py3-none-any.whl.metadata (2.8 kB)
Collecting pygments (from kivy)
  Using cached pygments-2.17.2-py3-none-any.whl.metadata (2.6 kB)
Collecting requests (from Kivy-Garden==0.1.4->kivy)
  Using cached requests-2.31.0-py3-none-any.whl.metadata (4.6 kB)
Collecting charset-normalizer<4,>=2 (from requests->Kivy-Garden==0.1.4->kivy)
  Using cached charset_normalizer-3.3.2-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (33 kB)
Collecting idna<4,>=2.5 (from requests->Kivy-Garden==0.1.4->kivy)
  Using cached idna-3.6-py3-none-any.whl.metadata (9.9 kB)
Collecting urllib3<3,>=1.21.1 (from requests->Kivy-Garden==0.1.4->kivy)
  Using cached urllib3-2.2.1-py3-none-any.whl.metadata (6.4 kB)
Collecting certifi>=2017.4.17 (from requests->Kivy-Garden==0.1.4->kivy)
  Using cached certifi-2024.2.2-py3-none-any.whl.metadata (2.2 kB)
Using cached Kivy-2.3.0-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (22.9 MB)
Using cached Kivy_Garden-0.1.5-py3-none-any.whl (4.6 kB)
Using cached docutils-0.20.1-py3-none-any.whl (572 kB)
Using cached pygments-2.17.2-py3-none-any.whl (1.2 MB)
Using cached requests-2.31.0-py3-none-any.whl (62 kB)
Using cached certifi-2024.2.2-py3-none-any.whl (163 kB)
Using cached charset_normalizer-3.3.2-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (140 kB)
Using cached idna-3.6-py3-none-any.whl (61 kB)
Using cached urllib3-2.2.1-py3-none-any.whl (121 kB)
Installing collected packages: urllib3, pygments, idna, docutils, charset-normalizer, certifi, requests, Kivy-Garden, kivy
Successfully installed Kivy-Garden-0.1.5 certifi-2024.2.2 charset-normalizer-3.3.2 docutils-0.20.1 idna-3.6 kivy-2.3.0 pygments-2.17.2 requests-2.31.0 urllib3-2.2.1
(my_env) ubuntu@DESKTOP-G9T39AE: ~$
```

### Libraries ที่ใช้ทั้งหมด

- pyjnius==1.6.1
- pyproject-toml==0.0.10
- kivy==2.3.0
- buildozer==1.5.0
- cython==0.29.37
- python-for-android==2024.1.21
- cmake==3.28.1
- bleak==0.21.1
- kivymd==1.1.1
- plyer==2.1.0

#### 4. สร้าง Directory สำหรับโปรเจคของเรา

- mkdir <PROJECT\_NAME>
- ตัวอย่าง mkdir myproject

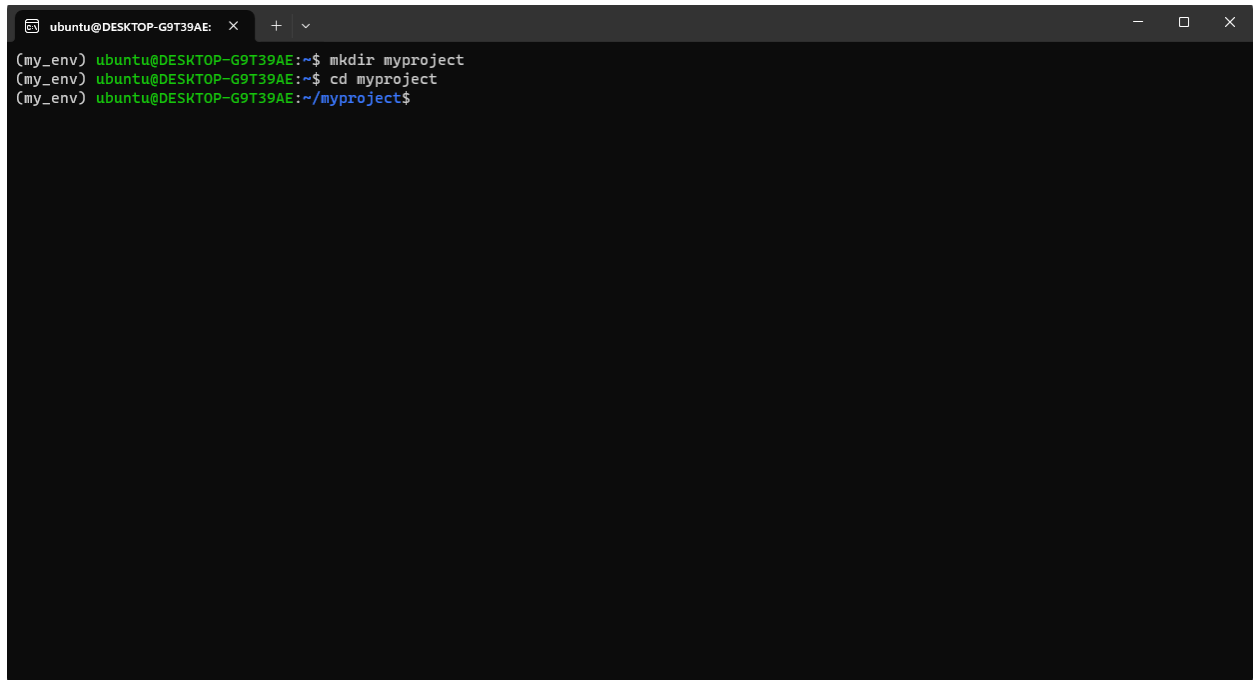


```
ubuntu@DESKTOP-G9T39AE: ~$ mkdir myproject
ubuntu@DESKTOP-G9T39AE: ~$
```

A terminal window with a dark background. The title bar shows 'ubuntu@DESKTOP-G9T39AE: ~' and standard window controls. The terminal shows the command 'mkdir myproject' being executed successfully, with a new prompt line appearing below it.

5. เข้าไปยัง Directory ที่ได้สร้างไว้

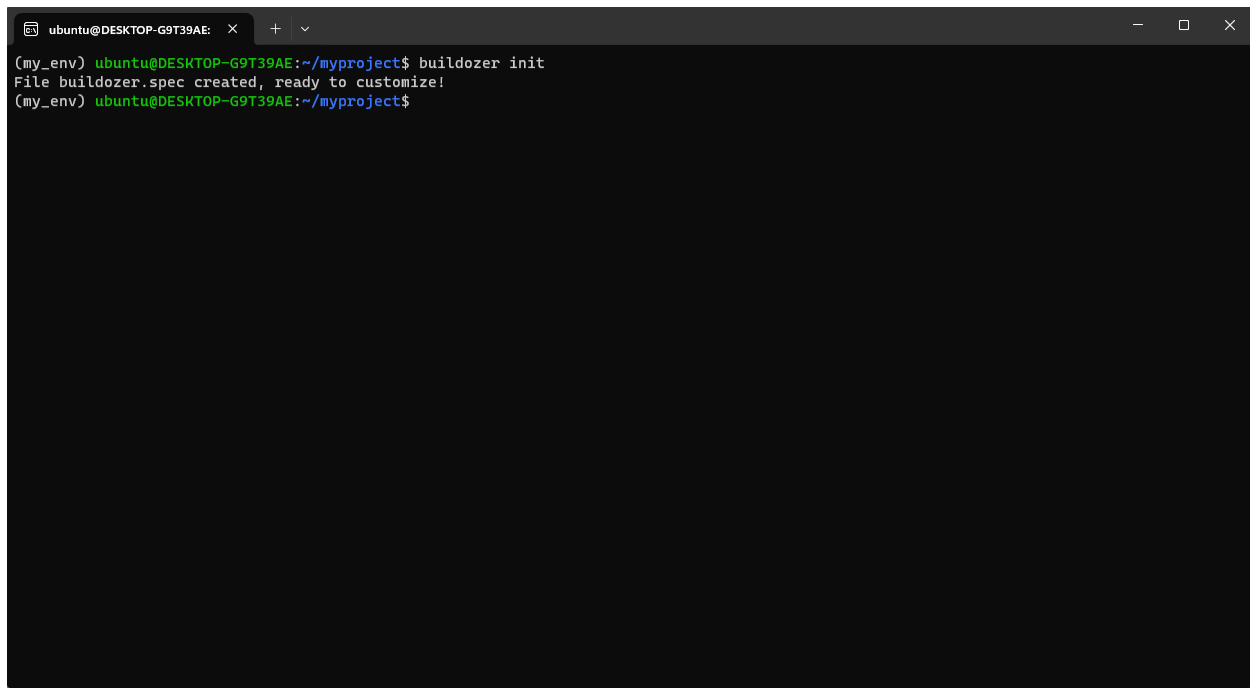
- `cd <PROJECT_PATH>`
- ตัวอย่าง `cd myproject`

A terminal window titled 'ubuntu@DESKTOP-G9T39AE' with standard window controls. It shows a sequence of three commands and their outputs: 1. Command: `mkdir myproject`, Output: `ubuntu@DESKTOP-G9T39AE:~$ mkdir myproject`. 2. Command: `cd myproject`, Output: `ubuntu@DESKTOP-G9T39AE:~$ cd myproject`. 3. Command: `cd myproject`, Output: `ubuntu@DESKTOP-G9T39AE:~/myproject$`. The prompt changes from `~$` to `~/myproject$` after the final command.

```
(my_env) ubuntu@DESKTOP-G9T39AE:~$ mkdir myproject
(my_env) ubuntu@DESKTOP-G9T39AE:~$ cd myproject
(my_env) ubuntu@DESKTOP-G9T39AE:~/myproject$
```

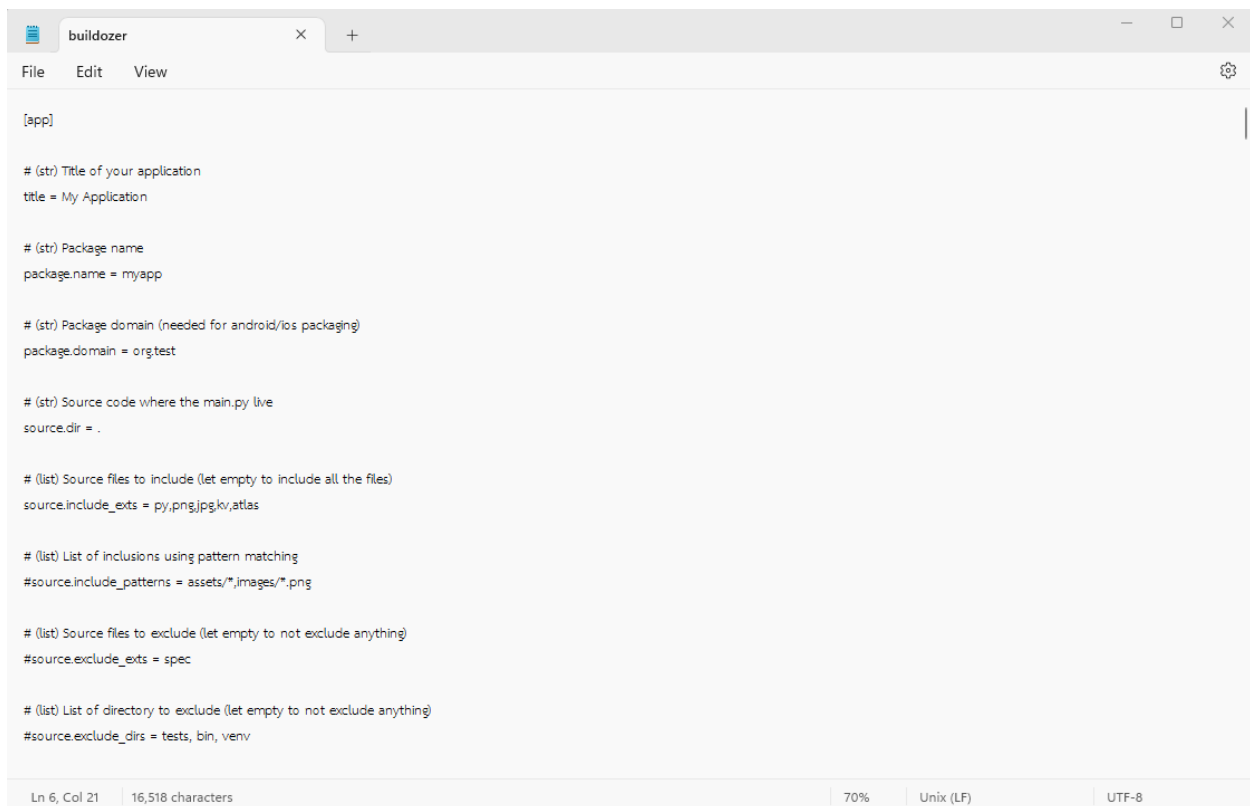
## 6. กำหนด Config ขึ้นมาเพื่อใช้ในการ Build App (Buildozer)

- buildozer init



```
ubuntu@DESKTOP-G9T39AE: ~$ buildozer init
File buildozer.spec created, ready to customize!
ubuntu@DESKTOP-G9T39AE: ~$
```

### Buildozer.spec



```
[app]

# (str) Title of your application
title = My Application

# (str) Package name
package.name = myapp

# (str) Package domain (needed for android/ios packaging)
package.domain = org.test

# (str) Source code where the main.py live
source.dir = .

# (list) Source files to include (let empty to include all the files)
source.include_exts = py,png,jpg,kv,atlas

# (list) List of inclusions using pattern matching
#source.include_patterns = assets/**/*.png

# (list) Source files to exclude (let empty to not exclude anything)
#source.exclude_exts = spec

# (list) List of directory to exclude (let empty to not exclude anything)
#source.exclude_dirs = tests, bin, venv
```

ปรับ Config ใน buildozer.spec ดังนี้ (ใช้ Ctrl + F ค้นหา)

title = BLE

package.name = scanservice

package.domain = test.able

source.include\_exts = py,png,jpg,kv,atlas,java mp4, file suffix allowance

requirements = kivy,python3,kivymd,able\_recipe,pyjnius from vscode

services = Able:service.py:foreground  
<https://gist.github.com/Arinerron/1bcaadc7b1cbeae77de0263f4e15156f>

android.permissions =

    FOREGROUND\_SERVICE,

    BLUETOOTH,

    BLUETOOTH\_ADMIN,

    BLUETOOTH\_SCAN,

    BLUETOOTH\_CONNECT,

    BLUETOOTH\_ADVERTISE,

    ACCESS\_FINE\_LOCATION

android.accept\_sdk\_license = True

android.add\_jars =

    java/org/able/BLE.java

    ,java/org/able/BLEAdvertiser.java

    ,java/org/able/PythonBluetooth.java

    ,java/org/able/PythonBluetoothAdvertiser.java

android.add\_src = java/

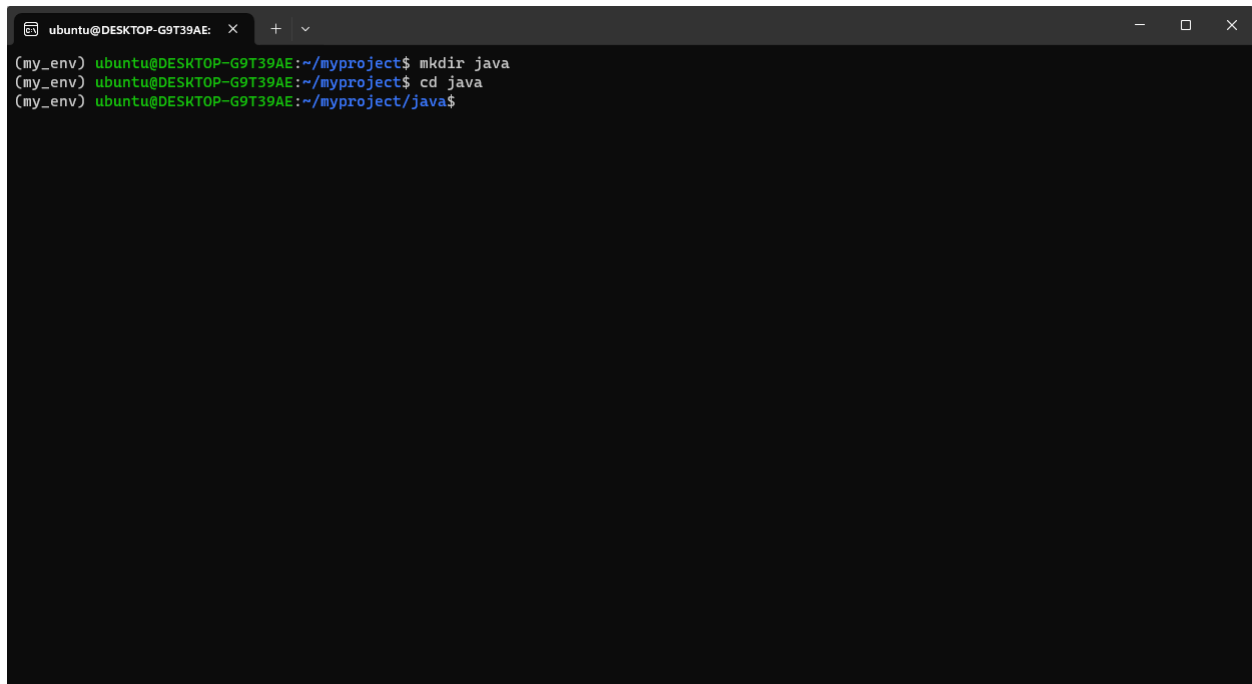
android.archs = arm64-v8a, armeabi-v7a



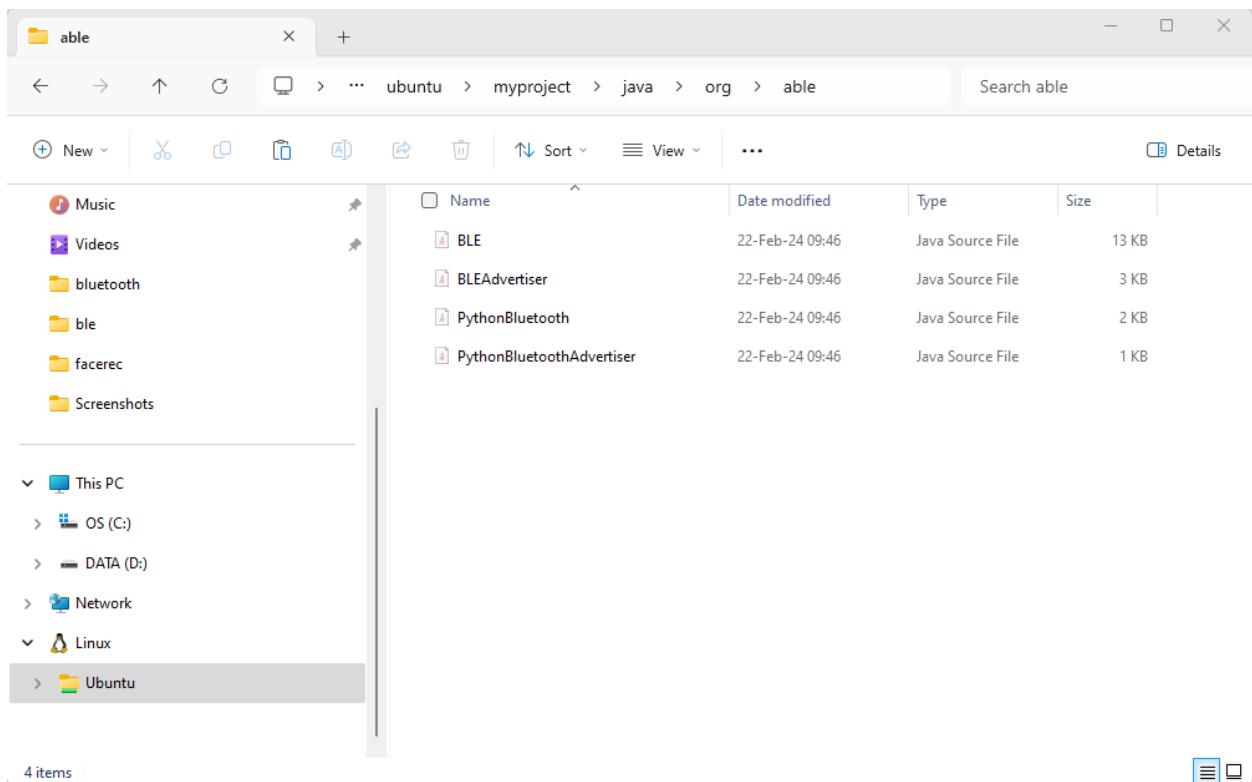


7. Download Java file มาใช้ในงาน

- Link: <https://github.com/b3b/able/tree/master/able/src/org/able>
- Download มาทั้ง 4 ไฟล์ (BLE, BLEAdvertiser, PythonBluetooth, PythonBluetoothAdvertiser)
- ใส่ไว้ในไฟล์ java (java/org/able)



```
ubuntu@DESKTOP-G9T39AE: ~$ mkdir java
ubuntu@DESKTOP-G9T39AE: ~$ cd java
ubuntu@DESKTOP-G9T39AE: ~/java$
```



8. สร้างไฟล์ `main.py` โดยมี source code ดังนี้

\*\*\* เป็นตัวอย่าง Application สำหรับ Scan อุปกรณ์ Bluetooth ใกล้เคียง

```
from able import GATT_SUCCESS, BluetoothDispatcher, require_bluetooth_enabled
from able.filters import DeviceNameFilter
from kivy.app import App
from kivy.clock import Clock
from kivy.logger import Logger
from kivy.lang import Builder
from kivy.utils import platform
from kivymd.app import MDApp
from jnius import autoclass
import jnius_config

jnius_config.set_classpath('org/able/BLE.jar')
jnius_config.set_classpath('org/able/BLEAdvertiser.jar')
jnius_config.set_classpath('org/able/PythonBluetooth.jar')
jnius_config.set_classpath('org/able/PythonBluetoothAdvertiser.jar')
autoclass('org.able.BLE')
autoclass('org.able.BLEAdvertiser')
autoclass('org.able.PythonBluetooth')
autoclass('org.able.PythonBluetoothAdvertiser')

KV = '''
MDBoxLayout:
    orientation: "vertical"

    MDTopAppBar:
        title: "Bluetooth Application"
        right_action_items: ["theme-light-dark", lambda x:
app.switch_theme_style()], ["exit-to-app", lambda x: app.close_application()]]

    MDBottomNavigation:

        MDBottomNavigationItem:
            name: 'screen 1'
            text: 'Scanner'
            icon: 'bluetooth'

        MDLabel
            id : Ble
            text: "Bluetooth Scanner"
            size_hint: 0.5, 0.1
            halign: "center"
```

```
    bold: True
    font_style: "H4"
    pos_hint: {"center_x": .5 , "center_y": .85}
```

MDLabel

```
    id : status
    halign: "center"
    size_hint_y: None
    pos_hint: {"center_x": .5 , "center_y": .15}
```

MDCard:

```
    ripple_behavior: False
    md_bg_color: app.theme_cls.primary_light
    size_hint: 0.6, 0.3
    pos_hint: {"center_x": .5 , "center_y": .6}
    MDLabel
        id : label
        theme_text_color: "Custom"
        size_hint: 0.3, 0.1
        halign: "center"
        pos_hint: {"center_x": .5 , "center_y": .5}
```

MDRectangleFlatButton:

```
    text: "Start"
    text_color: "black"
    on_press: app.start_service()
    md_bg_color: app.theme_cls.primary_light
    pos_hint: {"center_x": .4 , "center_y": .35}
```

MDRectangleFlatButton:

```
    text: "Stop"
    text_color: "black"
    on_press: app.stop_service()
    md_bg_color: app.theme_cls.primary_light
    pos_hint: {"center_x": .6 , "center_y": .35}
```

MDBottomNavigationItem:

```
    name: 'screen 2'
    text: 'Table'
    icon: 'table'
```

MDLabel:

```
    text: 'Table'
    halign: 'center'
```

```

        MDBottomNavigationBarItem:
            name: 'screen 3'
            text: 'Info'
            icon: 'information'

        MDLabel:
            text: 'Info'
            halign: 'center'

'''

class DeviceDispatcher(BluetoothDispatcher):
    """Dispatcher to control a single BLE device."""

    def __init__(self, device: "BluetoothDevice"):
        super().__init__()
        self._device = device
        self._address: str = device.getAddress()
        self._name: str = device.getName() or ""

    @property
    def title(self) -> str:
        return f"<{self._address}><{self._name}>"

    def on_connection_state_change(self, status: int, state: int):
        if status == GATT_SUCCESS and state:
            Logger.info(f"Device: {self.title} connected")
        else:
            Logger.info(f"Device: {self.title} disconnected. {status=},
{state=}")
            self.close_gatt()
            Clock.schedule_once(callback=lambda dt: self.reconnect(), timeout=15)

    def on_rssi_updated(self, rssi: int, status: int):
        Logger.info(f"Device: {self.title} RSSI: {rssi}")

    def periodically_update_rssi(self):
        """
        Clock callback to read
        the signal strength indicator for a connected device.
        """
        if self.gatt: # if device is connected
            self.update_rssi()

    def reconnect(self):

```

```

        Logger.info(f"Device: {self.title} try to reconnect ...")
        self.connect_gatt(self._device)

    def start(self):
        """Start connection to device."""
        if not self.gatt:
            self.connect_gatt(self._device)
            Clock.schedule_interval(
                callback=lambda dt: self.periodically_update_rssi(), timeout=5
            )

class ScannerDispatcher(BluetoothDispatcher):
    """Dispatcher to control the scanning process."""

    def __init__(self):
        super().__init__()
        self._devices: dict[str, DeviceDispatcher] = {}

    def on_scan_started(self, success: bool):
        if success:
            Logger.info("Scan: started")
        else:
            Logger.error("Scan: error on start")

    def on_scan_completed(self):
        Logger.info("Scan: completed")

    def on_device(self, device, rssi, advertisement):
        address = device.getAddress()
        if address not in self._devices:
            dispatcher = DeviceDispatcher(device)
            self._devices[address] = dispatcher
            Logger.info(f"Scan: device <{address}> added")
            dispatcher.start()

class MyApp(MDApp, BluetoothDispatcher):
    def build(self):
        return Builder.load_string(KV)

    def switch_theme_style(self):
        self.theme_cls.primary_palette = (
            "Orange" if self.theme_cls.primary_palette == "Blue" else "Blue"
        )
        self.theme_cls.theme_style = (
            "Dark" if self.theme_cls.theme_style == "Light" else "Light"

```

```

    )

    def close_application(self):
        # closing application
        App.get_running_app().stop()

    @property
    def service(self):
        return autoclass("test.able.scanservice.ServiceAble")

    @property
    def activity(self):
        return autoclass("org.kivy.android.PythonActivity").mActivity

    @require_bluetooth_enabled
    def start_service(self):
        self.service.start(self.activity, "")
        ScannerDispatcher().start_scan(filters=[DeviceNameFilter("P2N_09725")])
        ScannerDispatcher().start_scan(filters=[DeviceNameFilter("P2N_09714")])
        ScannerDispatcher().start_scan(filters=[DeviceNameFilter("ZLB_39612")])
        self.root.ids.label.text = "{name}, RSSI: {rssi} dBm\n{name}, RSSI:
{rssi} dBm\n{name}, RSSI: {rssi} dBm\n"
        self.root.ids.status.text = "Scanning..."

    def stop_service(self):
        self.service.stop(self.activity)
        self.root.ids.status.text = "Stop Scanning"

if __name__ == "__main__":
    MyApp().run()

```

9. สร้างไฟล์ Buildozer Debug ขึ้นมา (.apk file)

- buildozer -v <DEVICE> debug Android, iOS
- ตัวอย่าง buildozer -v android debug

```
ubuntu@DESKTOP-G9T39AE: ~$ conda activate my_env
(my_env) ubuntu@DESKTOP-G9T39AE: ~$ cd myproject
(my_env) ubuntu@DESKTOP-G9T39AE: ~/myproject$ ls
buildozer.spec  java  main.py
(my_env) ubuntu@DESKTOP-G9T39AE: ~/myproject$ buildozer -v android debug
# Check configuration tokens
# Ensure build layout
# Create directory /home/ubuntu/myproject/.buildozer
# Create directory /home/ubuntu/myproject/bin
# Create directory /home/ubuntu/myproject/.buildozer/applibs
# Create directory /home/ubuntu/myproject/.buildozer/android/platform
# Create directory /home/ubuntu/myproject/.buildozer/android/app
# Check configuration tokens
# Preparing build
# Check requirements for android
# Search for Git (git)
# -> found at /usr/bin/git
# Search for Cython (cython)
# -> found at /home/ubuntu/miniconda3/envs/my_env/bin/cython
# Search for Java compiler (javac)
# -> found at /usr/lib/jvm/java-17-openjdk-amd64/bin/javac
# Search for Java keytool (keytool)
# -> found at /usr/lib/jvm/java-17-openjdk-amd64/bin/keytool
# Install platform
# Run ['git', 'clone', '-b', 'master', '--single-branch', 'https://github.com/kivy/python-for-android.git', 'python-for-android']
# Cwd /home/ubuntu/myproject/.buildozer/android/platform
Cloning into 'python-for-android'...
# Run ['/home/ubuntu/miniconda3/envs/my_env/bin/python', '-m', 'pip', 'install', '-q', 'appdirs', 'colorama>=0.3.3', 'jinja2', 'sh>=1.10, <2.0; sys_platform!=\"win32\"', 'build', 'toml', 'packaging', 'setuptools']
```

```
ubuntu@DESKTOP-G9T39AE: ~$ buildozer -v android debug
[INFO]:    <- directory context /home/ubuntu/myproject/.buildozer/android/platform/python-for-android
[DEBUG]:    All possible dists: [<Distribution: name myapp with recipes (hostpython3, libffi, openssl, sdl2_image, sdl2_mixer, sdl2_ttf, sqlite3, python3, sdl2, setuptools, six, pyjnius, android, kivy, requests, urllib3, certifi, chardet, idna)>]
[DEBUG]:    Dist matching name and arch: [<Distribution: name myapp with recipes (hostpython3, libffi, openssl, sdl2_image, sdl2_mixer, sdl2_ttf, sqlite3, python3, sdl2, setuptools, six, pyjnius, android, kivy, requests, urllib3, certifi, chardet, idna)>]
[DEBUG]:    Dist matching ndk_api and recipe: [<Distribution: name myapp with recipes (hostpython3, libffi, openssl, sdl2_image, sdl2_mixer, sdl2_ttf, sqlite3, python3, sdl2, setuptools, six, pyjnius, android, kivy, requests, urllib3, certifi, chardet, idna)>]
[INFO]:    Of the existing distributions, the following meet the given requirements:
[INFO]:    myapp: min API 21, includes recipes (hostpython3, libffi, openssl, sdl2_image, sdl2_mixer, sdl2_ttf, sqlite3, python3, sdl2, setuptools, six, pyjnius, android, kivy, requests, urllib3, certifi, chardet, idna), built for archs (armeabi-v7a, arm64-v8a)
[INFO]:    myapp has compatible recipes, using this one
[INFO]:    # Copying android package to current directory
[INFO]:    # Android package filename not found in build output. Guessing...
[INFO]:    # Found android package file: /home/ubuntu/myproject/.buildozer/android/platform/build-arm64-v8a_armeabi-v7a/dists/myapp/build/outputs/apk/debug/myapp-debug.apk
[INFO]:    # Add version number to android package
[INFO]:    # Android package renamed to myapp-debug-0.1.apk
[DEBUG]:    -> running cp /home/ubuntu/myproject/.buildozer/android/platform/build-arm64-v8a_armeabi-v7a/dists/myapp/build/outputs/apk/debug/myapp-debug.apk myapp-debug-0.1.apk
No setup.py/pyproject.toml used, copying full private data into .apk.
Applying Java source code patches...
Applying patch: src/patches/SDLActivity.java.patch
# Android packaging done!
# APK myapp-0.1-arm64-v8a_armeabi-v7a-debug.apk available in the bin directory
(my_env) ubuntu@DESKTOP-G9T39AE: ~/myproject$
```

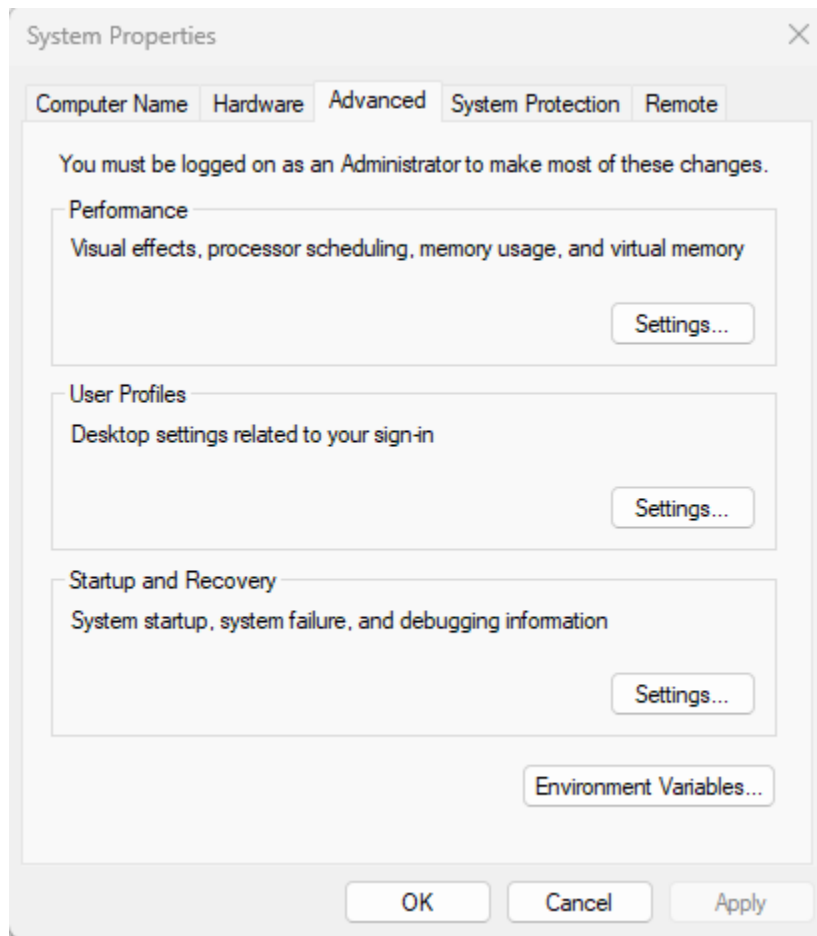
10. ติดตั้ง ADB สำหรับใช้งานร่วมกับ Buildozer (ใน Mobile และ ใน PC)

สำหรับ Mobile

- เข้า Settings
- แตะที่เมนู About Phone (เกี่ยวกับโทรศัพท์)
- แตะที่เมนู Software info. (ข้อมูลซอฟต์แวร์)
- แตะที่ Build number 7 ครั้ง เพื่อเปิด Developer mode
- กลับไปยังหน้า Settings แตะที่เมนู Developer Mode
- จากนั้นเปิดการใช้งาน USB Debugging

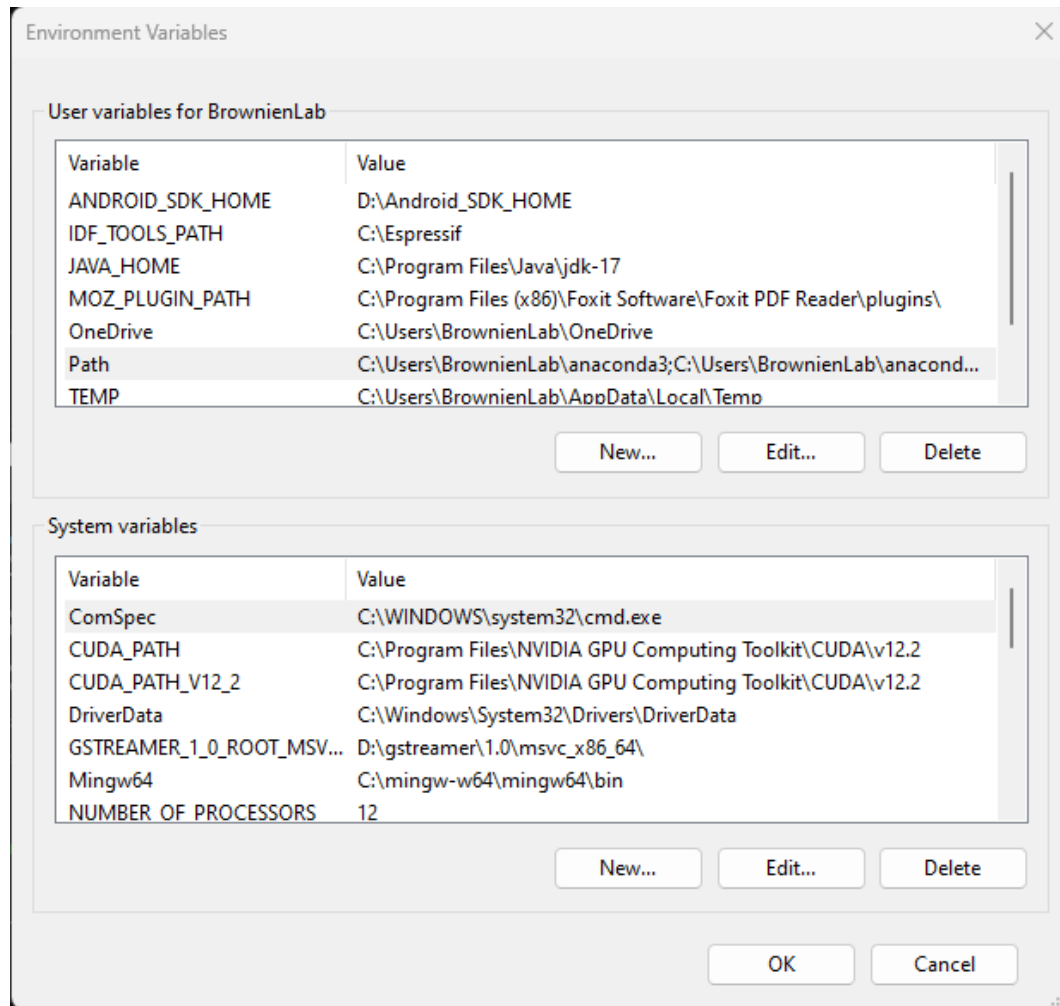
สำหรับ PC

- กำหนด Path ใน Environment Variables

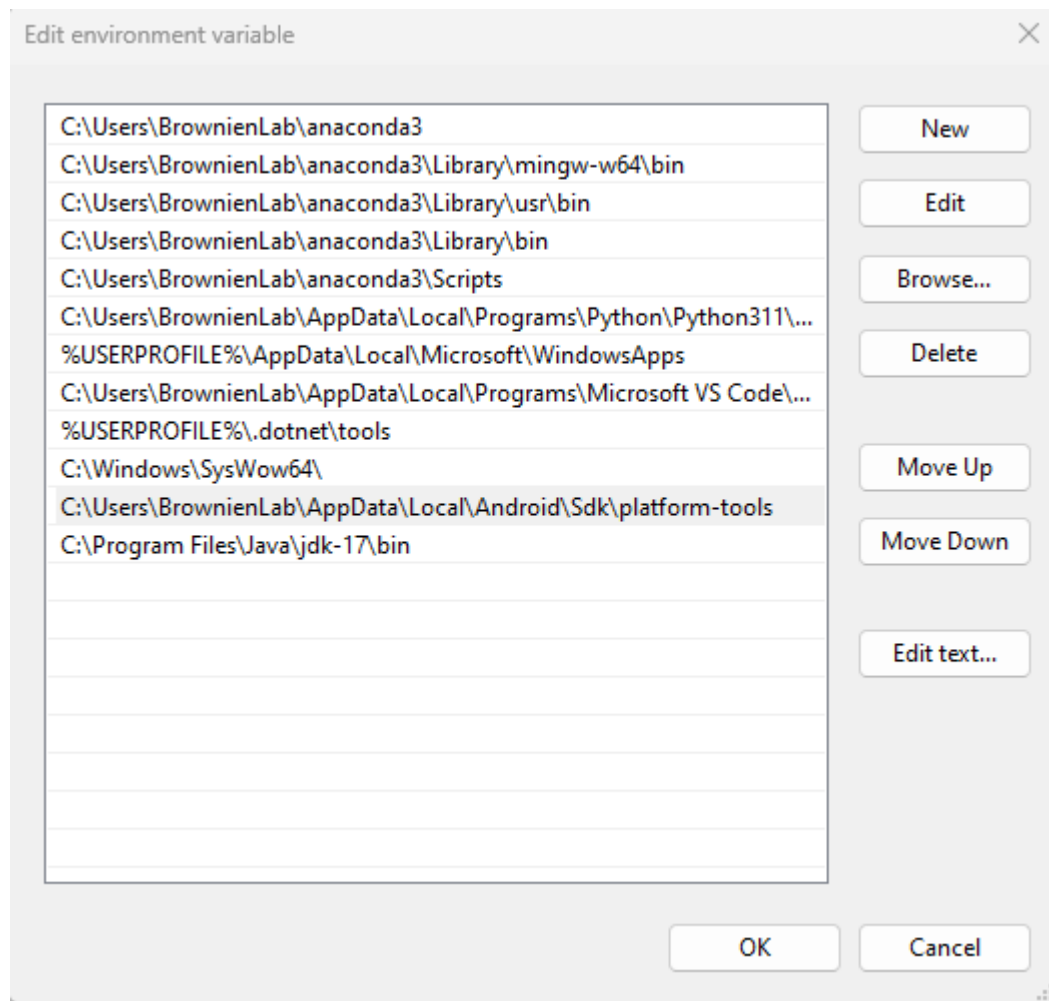




- คลิกที่ Path



- คลิก New และใส่ Path ตามนี้
- C:\Users\BrownienLab\AppData\Local\Android\Sdk\platform-tools



- จากนั้นใช้คำสั่ง `adb devices` ใน terminal

```
Command Prompt
Microsoft Windows [Version 10.0.22631.3155]
(c) Microsoft Corporation. All rights reserved.

C:\Users\BrownienLab>adb devices
List of devices attached
RF8R51MYCNA    device

C:\Users\BrownienLab>
```

USB

- ในรูปตัวอย่างคือ Device ที่ได้เชื่อมต่อด้วย USB ไว้ (Samsung A32)
- หลังจากนั้นใช้คำสั่ง `adb tcpip 5555`      `adb kill-server`

```
Command Prompt
Microsoft Windows [Version 10.0.22631.3155]
(c) Microsoft Corporation. All rights reserved.

C:\Users\BrownienLab>adb devices
List of devices attached
RF8R51MYCNA    device

C:\Users\BrownienLab>adb tcpip 5555
restarting in TCP mode port: 5555

C:\Users\BrownienLab>
```

Wi-Fi  
2

Mobile IP Address

Setting/About Device/Status/IP Address

- จากนั้นเปิด WSL (Ubuntu) ขึ้นมาอีกครั้ง
- ใช้คำสั่ง `adb connect <your_address>:5555`
- ตัวอย่าง `adb connect 192.168.100.176:5555`

```
ubuntu@DESKTOP-G9T39AE: ~$ conda activate my_env
(my_env) ubuntu@DESKTOP-G9T39AE:~$ adb connect 192.168.100.176:5555
connected to 192.168.100.176:5555
(my_env) ubuntu@DESKTOP-G9T39AE:~$ |
```

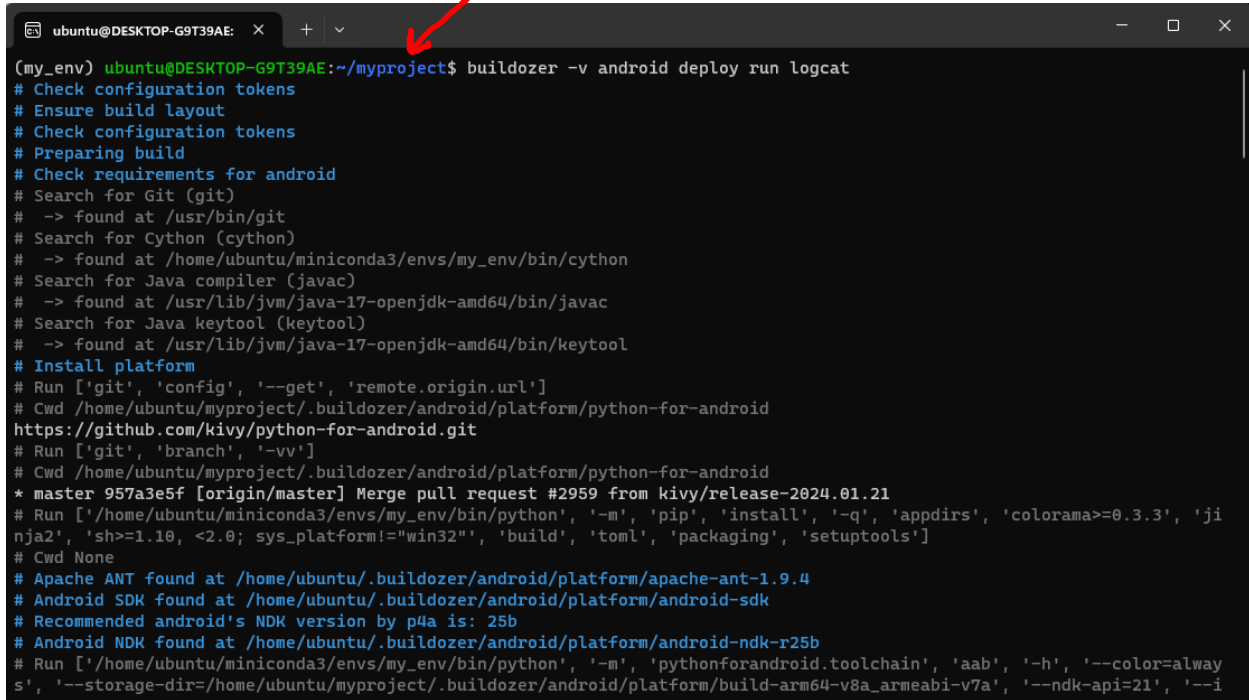
adb devices

- `Adb devices` เพื่อเช็ค device ของเราพร้อมใช้งานหรือยัง

```
ubuntu@DESKTOP-G9T39AE: ~$ conda activate my_env
(my_env) ubuntu@DESKTOP-G9T39AE:~$ adb connect 192.168.100.176:5555
connected to 192.168.100.176:5555
(my_env) ubuntu@DESKTOP-G9T39AE:~$ adb devices
List of devices attached
192.168.100.176:5555    device
(my_env) ubuntu@DESKTOP-G9T39AE:~$
```

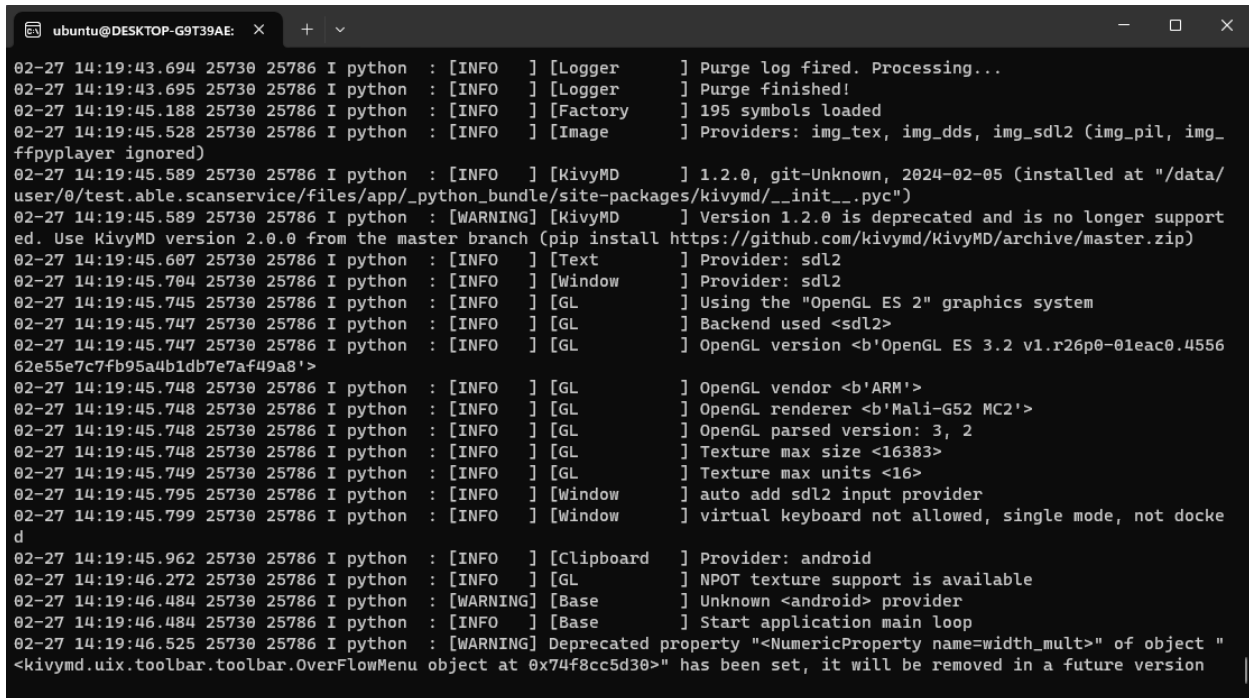
11. ทดสอบรัน Application ด้วยคำสั่ง Deploy และแสดง output ด้วย logcat

- `buildozer -v <DEVICE> deploy run logcat`
- `buildozer -v android deploy run logcat`

A terminal window titled 'ubuntu@DESKTOP-G9T39AE' with a red arrow pointing to the command 'buildozer -v android deploy run logcat'. The output shows various checks for configuration tokens, build layout, and requirements for android. It then proceeds to install the platform, cloning 'python-for-android' from GitHub. The output continues with details about the Android SDK, NDK version, and the installation of various tools like Apache ANT, Java compiler, and Java keytool. The command ends with 'Run' instructions for the build process.

```
(my_env) ubuntu@DESKTOP-G9T39AE:~/myproject$ buildozer -v android deploy run logcat
# Check configuration tokens
# Ensure build layout
# Check configuration tokens
# Preparing build
# Check requirements for android
# Search for Git (git)
# -> found at /usr/bin/git
# Search for Cython (cython)
# -> found at /home/ubuntu/miniconda3/envs/my_env/bin/cython
# Search for Java compiler (javac)
# -> found at /usr/lib/jvm/java-17-openjdk-amd64/bin/javac
# Search for Java keytool (keytool)
# -> found at /usr/lib/jvm/java-17-openjdk-amd64/bin/keytool
# Install platform
# Run ['git', 'config', '--get', 'remote.origin.url']
# Cwd /home/ubuntu/myproject/.buildozer/android/platform/python-for-android
https://github.com/kivy/python-for-android.git
# Run ['git', 'branch', '-vv']
# Cwd /home/ubuntu/myproject/.buildozer/android/platform/python-for-android
* master 957a3e5f [origin/master] Merge pull request #2959 from kivy/release-2024.01.21
# Run ['/home/ubuntu/miniconda3/envs/my_env/bin/python', '-m', 'pip', 'install', '-q', 'appdirs', 'colorama>=0.3.3', 'jinja2', 'sh>=1.10, <2.0; sys_platform!="win32"', 'build', 'toml', 'packaging', 'setuptools']
# Cwd None
# Apache ANT found at /home/ubuntu/.buildozer/android/platform/apache-ant-1.9.4
# Android SDK found at /home/ubuntu/.buildozer/android/platform/android-sdk
# Recommended android's NDK version by p4a is: 25b
# Android NDK found at /home/ubuntu/.buildozer/android/platform/android-ndk-r25b
# Run ['/home/ubuntu/miniconda3/envs/my_env/bin/python', '-m', 'pythonforandroid.toolchain', 'aab', '-h', '--color=always', '--storage-dir=/home/ubuntu/myproject/.buildozer/android/platform/build-arm64-v8a-armeabi-v7a', '--ndk-api=21', '--i
```

- ตัวอย่าง logcat

A terminal window titled 'ubuntu@DESKTOP-G9T39AE' showing logcat output. The output is a series of log messages from the application, including information about the KivyMD version, OpenGL ES 2 graphics system, and various providers like sdl2, img\_tex, img\_dds, img\_sdl2, img\_pil, and img\_ffpyplayer. The messages are timestamped and include log levels like INFO, WARNING, and ERROR.

```
02-27 14:19:43.694 25730 25786 I python : [INFO ] [Logger ] Purge log fired. Processing...
02-27 14:19:43.695 25730 25786 I python : [INFO ] [Logger ] Purge finished!
02-27 14:19:45.188 25730 25786 I python : [INFO ] [Factory ] 195 symbols loaded
02-27 14:19:45.528 25730 25786 I python : [INFO ] [Image ] Providers: img_tex, img_dds, img_sdl2 (img_pil, img_ffpyplayer ignored)
02-27 14:19:45.589 25730 25786 I python : [INFO ] [KivyMD ] 1.2.0, git-Unknown, 2024-02-05 (installed at "/data/user/0/test.able.scanservice/files/app/_python_bundle/site-packages/kivymd/__init__.pyc")
02-27 14:19:45.589 25730 25786 I python : [WARNING] [KivyMD ] Version 1.2.0 is deprecated and is no longer support ed. Use KivyMD version 2.0.0 from the master branch (pip install https://github.com/kivymd/KivyMD/archive/master.zip)
02-27 14:19:45.607 25730 25786 I python : [INFO ] [Text ] Provider: sdl2
02-27 14:19:45.704 25730 25786 I python : [INFO ] [Window ] Provider: sdl2
02-27 14:19:45.745 25730 25786 I python : [INFO ] [GL ] Using the "OpenGL ES 2" graphics system
02-27 14:19:45.747 25730 25786 I python : [INFO ] [GL ] Backend used <sdl2>
02-27 14:19:45.747 25730 25786 I python : [INFO ] [GL ] OpenGL version <b'OpenGL ES 3.2 v1.r26p0-01eac0.455662e55e7c7fb95a4b1db7e7af49a8'>
02-27 14:19:45.748 25730 25786 I python : [INFO ] [GL ] OpenGL vendor <b'ARM'>
02-27 14:19:45.748 25730 25786 I python : [INFO ] [GL ] OpenGL renderer <b'Mali-G52 MC2'>
02-27 14:19:45.748 25730 25786 I python : [INFO ] [GL ] OpenGL parsed version: 3, 2
02-27 14:19:45.748 25730 25786 I python : [INFO ] [GL ] Texture max size <16383>
02-27 14:19:45.749 25730 25786 I python : [INFO ] [GL ] Texture max units <16>
02-27 14:19:45.795 25730 25786 I python : [INFO ] [Window ] auto add sdl2 input provider
02-27 14:19:45.799 25730 25786 I python : [INFO ] [Window ] virtual keyboard not allowed, single mode, not docke d
02-27 14:19:45.962 25730 25786 I python : [INFO ] [Clipboard ] Provider: android
02-27 14:19:46.272 25730 25786 I python : [INFO ] [GL ] NPOT texture support is available
02-27 14:19:46.484 25730 25786 I python : [WARNING] [Base ] Unknown <android> provider
02-27 14:19:46.484 25730 25786 I python : [INFO ] [Base ] Start application main loop
02-27 14:19:46.525 25730 25786 I python : [WARNING] [Base ] Deprecated property "<NumericProperty name=width_mult>" of object "<kivymd.uix.toolbar.toolbar.OverflowMenu object at 0x74f8cc5d30>" has been set, it will be removed in a future version
```

- ตัวอย่าง Application

