Project Setup Guide

This guide walks you through the steps of setting up the development environment on Ubuntu, linking an Android app using Maestro, and connecting to the device using ADB.

Prerequisites

1. Ubuntu installed on your PC
2. Android Studio installed
3. Node.js and Yarn installed
4. Java Development Kit (JDK) installed
5. Git installed

Once project is developed using the ignite boilerplate.

Android SDK setup

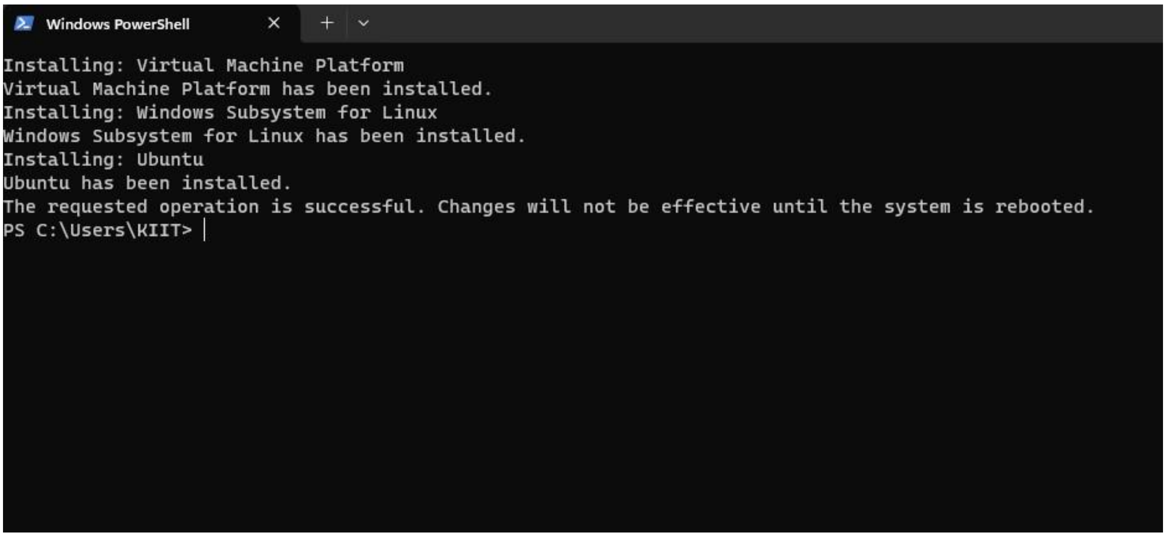
- go to google android dev and download the latest version of the android hedhghoge and install it with all the default modules,
- after that, turn on your emulator, by clicking on "More Options"

Setting up the "JAVA_HOME" and "ANDROID_HOME" Requirements

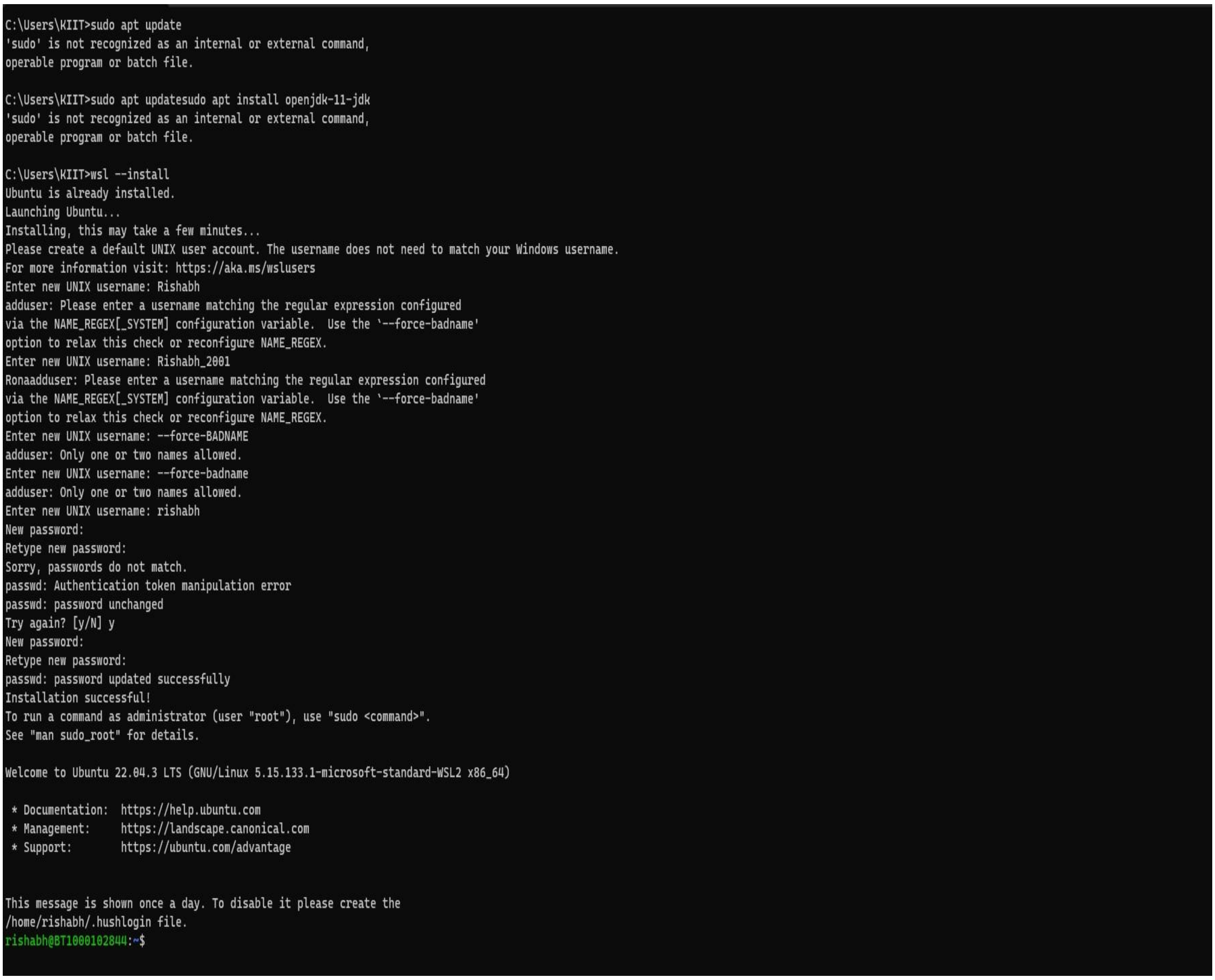
Ubuntu Setup

1. Install Ubuntu:

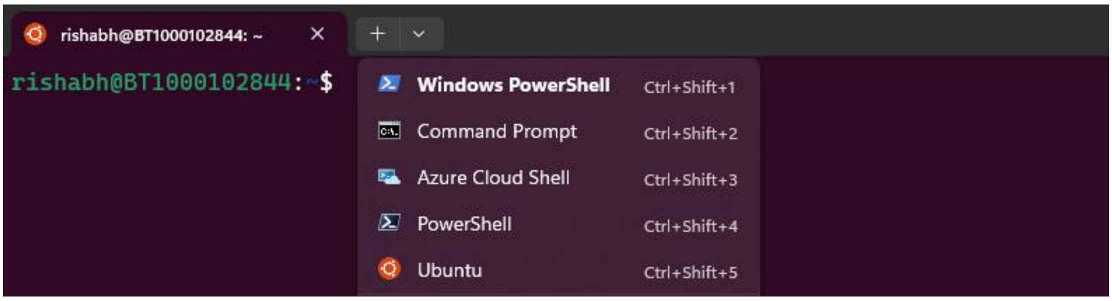
- installing the wsl using wsl --install
- then, wsl --install -d Ubuntu
- now install wsl subsystem for the linux systems



```
Installing: Virtual Machine Platform
Virtual Machine Platform has been installed.
Installing: Windows Subsystem for Linux
Windows Subsystem for Linux has been installed.
Installing: Ubuntu
Ubuntu has been installed.
The requested operation is successful. Changes will not be effective until the system is rebooted.
PS C:\Users\KIIT> |
```



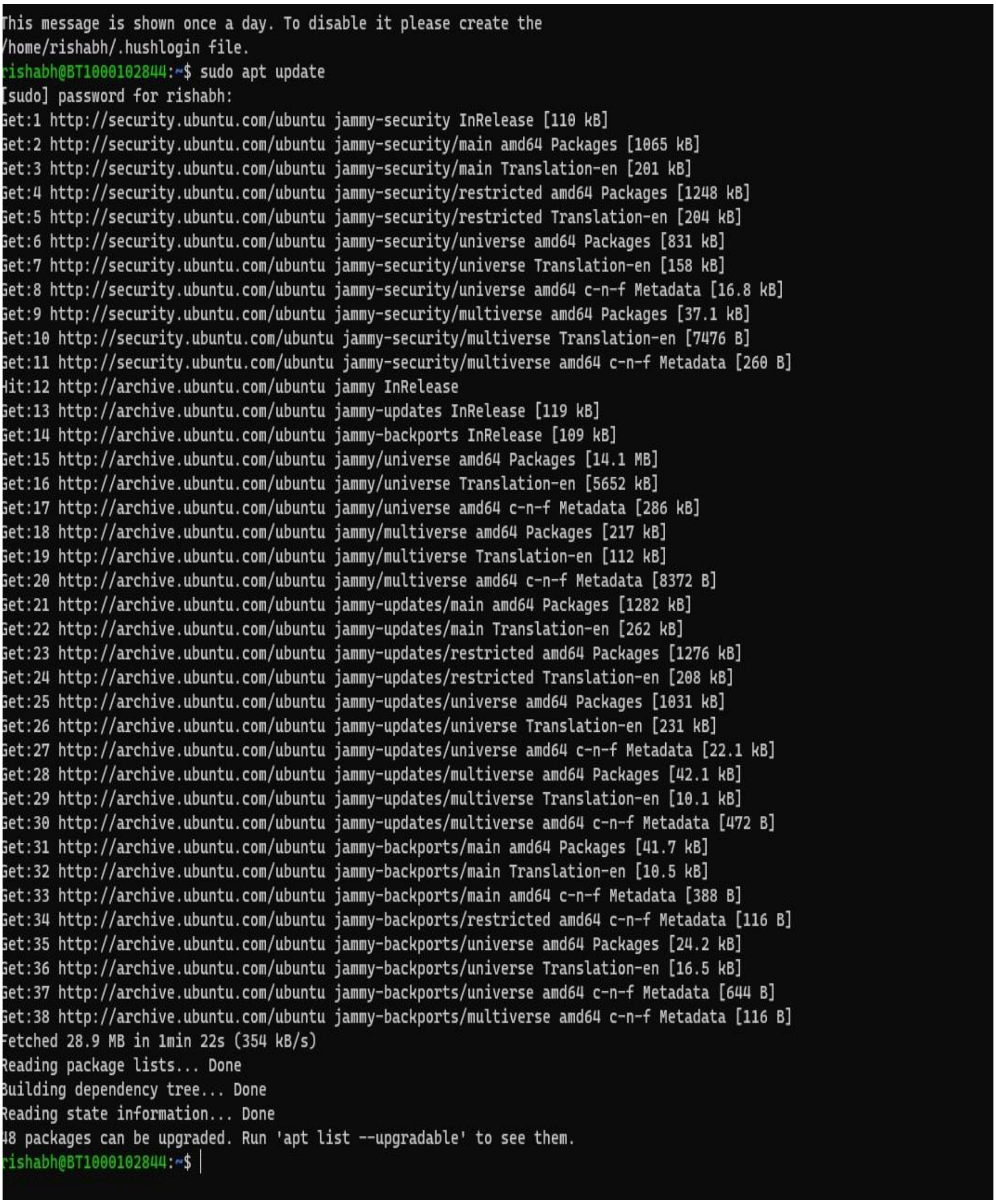
```
C:\Users\KIIT>sudo apt update
'sudo' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\KIIT>sudo apt updatesudo apt install openjdk-11-jdk
'sudo' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\KIIT>wsl --install
Ubuntu is already installed.
Launching Ubuntu...
Installing, this may take a few minutes...
Please create a default UNIX user account. The username does not need to match your Windows username.
For more information visit: https://aka.ms/wslusers
Enter new UNIX username: Rishabh
adduser: Please enter a username matching the regular expression configured
via the NAME_REGEX[_SYSTEM] configuration variable.  Use the `--force-badname'
option to relax this check or reconfigure NAME_REGEX.
Enter new UNIX username: Rishabh_2001
Ronaadduser: Please enter a username matching the regular expression configured
via the NAME_REGEX[_SYSTEM] configuration variable.  Use the `--force-badname'
option to relax this check or reconfigure NAME_REGEX.
Enter new UNIX username: --force-BADNAME
adduser: Only one or two names allowed.
Enter new UNIX username: --force-badname
adduser: Only one or two names allowed.
Enter new UNIX username: rishabh
New password:
Retype new password:
Sorry, passwords do not match.
passwd: Authentication token manipulation error
passwd: password unchanged
Try again? [y/N] y
New password:
Retype new password:
passwd: password updated successfully
Installation successful!
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.133.1-microsoft-standard-WSL2 x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage


This message is shown once a day. To disable it please create the
/home/rishabh/.hushlogin file.
rishabh@BT1000102844:~$
```



rishabh@BT1000102844: ~
Windows PowerShell Ctrl+Shift+1
Command Prompt Ctrl+Shift+2
Azure Cloud Shell Ctrl+Shift+3
PowerShell Ctrl+Shift+4
Ubuntu Ctrl+Shift+5

```
rishabh@BT1000102844:~$
```

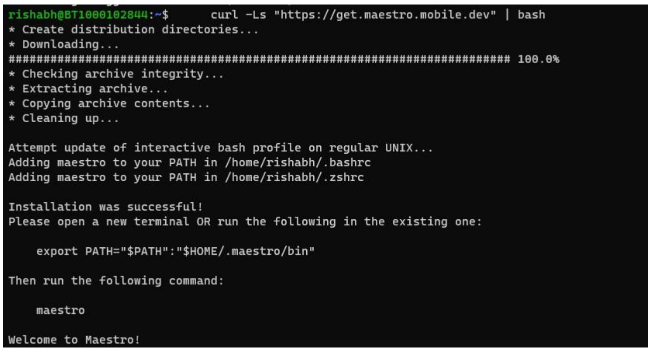
2. Update and Upgrade: sudo apt update sudo apt upgrade



```
This message is shown once a day. To disable it please create the
/home/rishabh/.hushlogin file.
rishabh@BT1000102844:~$ sudo apt update
[sudo] password for rishabh:
Get:1 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:2 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1065 kB]
Get:3 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [201 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [1248 kB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [204 kB]
Get:6 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [831 kB]
Get:7 http://security.ubuntu.com/ubuntu jammy-security/universe Translation-en [158 kB]
Get:8 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 c-n-f Metadata [16.8 kB]
Get:9 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [37.1 kB]
Get:10 http://security.ubuntu.com/ubuntu jammy-security/multiverse Translation-en [7476 B]
Get:11 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 c-n-f Metadata [260 B]
Hit:12 http://archive.ubuntu.com/ubuntu jammy InRelease
Get:13 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:14 http://archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Get:15 http://archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:16 http://archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]
Get:17 http://archive.ubuntu.com/ubuntu jammy/universe amd64 c-n-f Metadata [286 kB]
Get:18 http://archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [217 kB]
Get:19 http://archive.ubuntu.com/ubuntu jammy/multiverse Translation-en [112 kB]
Get:20 http://archive.ubuntu.com/ubuntu jammy/multiverse amd64 c-n-f Metadata [8372 B]
Get:21 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1282 kB]
Get:22 http://archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [262 kB]
Get:23 http://archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [1276 kB]
Get:24 http://archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [208 kB]
Get:25 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1031 kB]
Get:26 http://archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [231 kB]
Get:27 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 c-n-f Metadata [22.1 kB]
Get:28 http://archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [42.1 kB]
Get:29 http://archive.ubuntu.com/ubuntu jammy-updates/multiverse Translation-en [10.1 kB]
Get:30 http://archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 c-n-f Metadata [472 B]
Get:31 http://archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [41.7 kB]
Get:32 http://archive.ubuntu.com/ubuntu jammy-backports/main Translation-en [10.5 kB]
Get:33 http://archive.ubuntu.com/ubuntu jammy-backports/main amd64 c-n-f Metadata [388 B]
Get:34 http://archive.ubuntu.com/ubuntu jammy-backports/restricted amd64 c-n-f Metadata [116 B]
Get:35 http://archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [24.2 kB]
Get:36 http://archive.ubuntu.com/ubuntu jammy-backports/universe Translation-en [16.5 kB]
Get:37 http://archive.ubuntu.com/ubuntu jammy-backports/universe amd64 c-n-f Metadata [644 B]
Get:38 http://archive.ubuntu.com/ubuntu jammy-backports/multiverse amd64 c-n-f Metadata [116 B]
Fetched 28.9 MB in 1min 22s (354 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
48 packages can be upgraded. Run 'apt list --upgradable' to see them.
rishabh@BT1000102844:~$
```

3. Now, inside the ubuntu console follow these steps:

Install Java After restarting the system, open the Terminal application and click on the dropdown to select Ubuntu. Type in the following command: sudo apt install openjdk-11-jdk 3. Install Maestro Installing Maestro is now just a matter of running following one command. curl -Ls "https://get.maestro.mobile.dev" | bash Tada!   You have successfully installed Maestro on your Windows machine   Check your Maestro version using the following command: maestro --version set you up to use Android in your freshly installed WSL2 Download the Android command line tools zip file from Android official site. Use the following instructions to set up Android command lines correctly in your WSL2. Open WSL2 terminal. Create a new directory in your home directory.



```
rishabh@BT1000102844:~$      curl -Ls "https://get.maestro.mobile.dev" | bash
* Create distribution directories...
* Downloading...
######################################################################## 100.0%
* Checking archive integrity...
* Extracting archive...
* Copying archive contents...
* Cleaning up...

Attempt update of interactive bash profile on regular UNIX...
Adding maestro to your PATH in /home/rishabh/.bashrc
Adding maestro to your PATH in /home/rishabh/.zshrc

Installation was successful!
Please open a new terminal OR run the following in the existing one:

    export PATH="$PATH":"$HOME/.maestro/bin"

Then run the following command:

    maestro

Welcome to Maestro!
```

~ $ mkdir Android

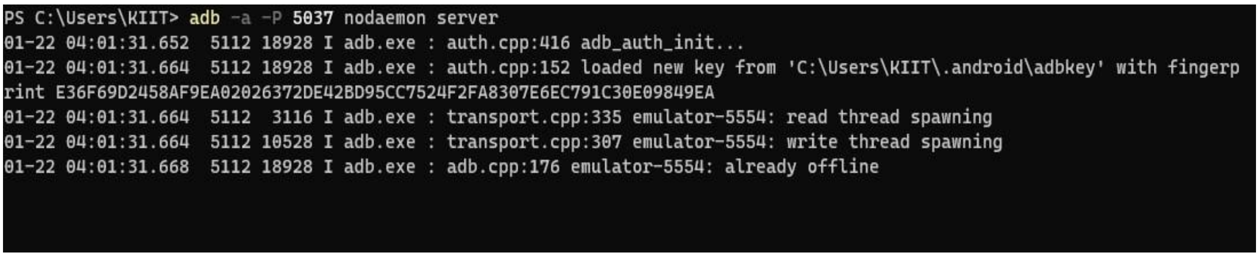
~ $ cd Android ```

Unzip the Android command line tools zip file in the android directory using this command: unzip ~<command_line_zip_filename>.zip In the Android directory perform following actions. $ mkdir latest $ mv cmdline-tools/* latest/ $ mv latest/ cmdline-tools/ Note: Last command will probably give you a warning, but you don’t need the worry about that. Now add the following line to your ~/.bashrc file export ANDROID_HOME=$HOME/Android export PATH=$PATH:$ANDROID_HOME/cmdline-tools/latest/bin/:$PATH Save your ~/.bashrc file and exit. Run source ~/.bashrc to reload the bashrc file. Now, we will install basic Android utilities using the following commands: Run sdkmanager --list to check if everything is working fine. Run sdkmanager --install "platform-tools" to install platform tools. Finally, add the following into your ~/.bashrc file export PATH=$PATH:$ANDROID_HOME/platform-tools/:$PATH Save your ~/.bashrc file and exit. Run source ~/.bashrc to reload the bashrc file. To check that everything went well, do the following: Close and relaunch terminal Run adb --version and see that adb version is shown Since everything is installed fresh, your WSL 2 adb version should perfectly match with Windows ADB version that we noted down as part of the pre-requisites.

4. Emulator on

- Fire up your Android emulator on Windows. Once the Android emulator is up and running, open a PowerShell prompt. Run this command in PowerShell adb -a -P 5037 nodaemon server This will start the adb server in the Windows host. Note down the IPV4 address of your Windows host PC/machine.

5. Sometimes you may get smartsockets.. error when you run adb -a -P 5037 nodaemon server command in PowerShell. In that case, you can do the following steps:



```
PS C:\Users\KIIT> adb -a -P 5037 nodaemon server
01-22 04:01:31.652  5112 18928 I adb.exe : auth.cpp:416 adb_auth_init...
01-22 04:01:31.664  5112 18928 I adb.exe : auth.cpp:152 loaded new key from 'C:\Users\KIIT\.android\adbkey' with fingerp
rint E36F69D2458AF9EA02026372DE42BD95CC7524F2FA8307E6EC791C30E09849EA
01-22 04:01:31.664  5112  3116 I adb.exe : transport.cpp:335 emulator-5554: read thread spawning
01-22 04:01:31.664  5112 10528 I adb.exe : transport.cpp:307 emulator-5554: write thread spawning
01-22 04:01:31.668  5112 18928 I adb.exe : adb.cpp:176 emulator-5554: already offline
```

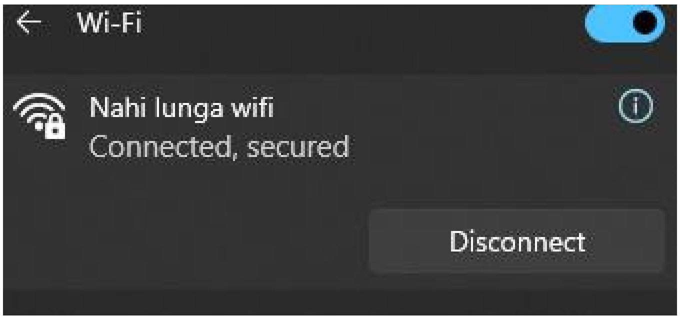
Open task manager and kill all adb related processes. If Android Studio is open, close it and keep only emulator running. If you see a message saying emulator offline, ignore it.

Now on WSL

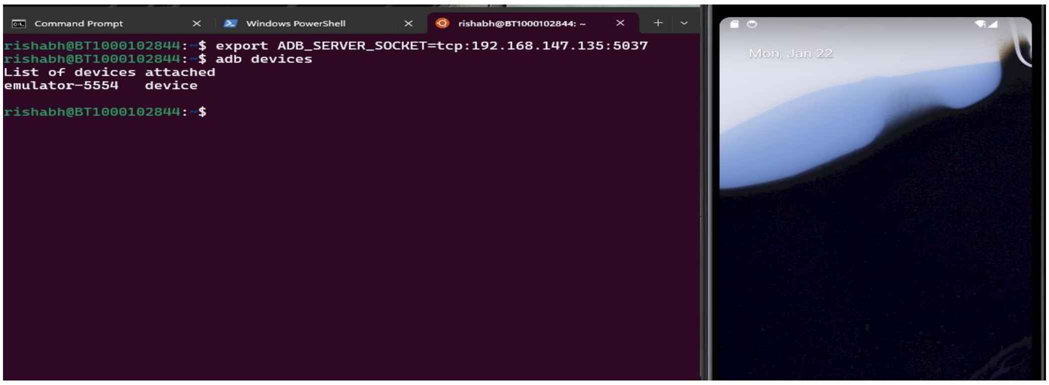
adb kill-server

export ADB_SERVER_SOCKET=tcp:<WINDOWS_IPV4_ADDR>:5037

(You can find your wifi IP address in the seettin of your network by Setting-> WiFi->



Click on i button beside connected wifi-> conpy the IPV4 address) adb devices You should see your connected emulator successfully now.

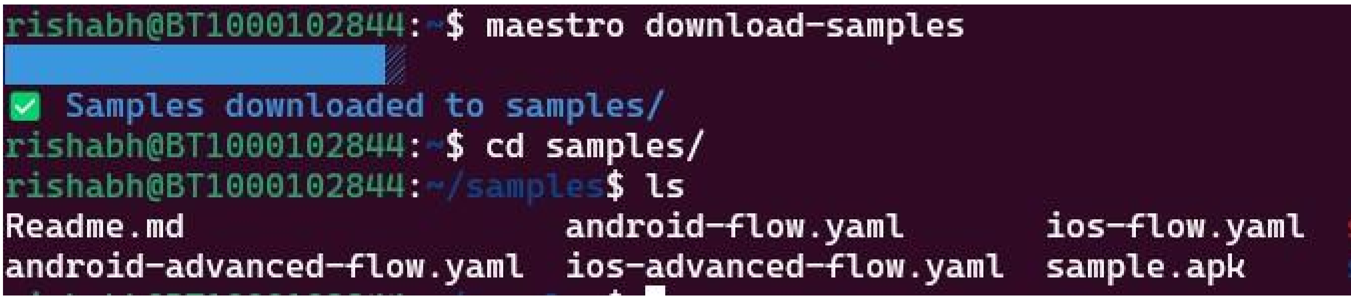


```
rishabh@BT1000102844:~$ export ADB_SERVER_SOCKET=tcp:192.168.147.135:5037
rishabh@BT1000102844:~$ adb devices
List of devices attached
emulator-5554   device

rishabh@BT1000102844:~$
```

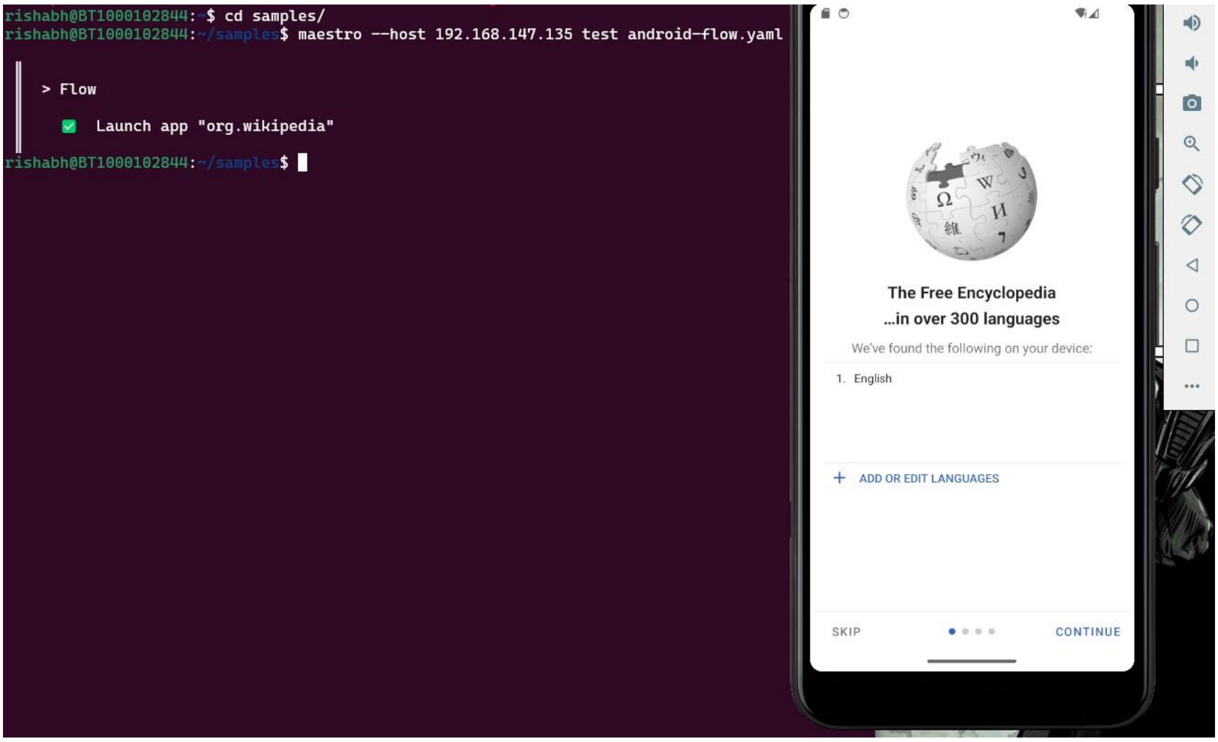
You have now successfully insatlled the maestro, ubuntu, android SDK, java and setup the environment

6. After your project is built then, you can set the project by using

- yarn ios or yarn android
- once project is connected
- inside the ubuntu you can now write the test cases for example, if you have a folder named samples
- and it has test cases named as android-flow.yaml
- this yaml file is used to define the flow of the application
- to set up this
- maestro --host 192.168.29.17 test android-flow.yaml can be used to run the test samples example, i'm using the default wikkipidea page



```
rishabh@BT1000102844:~$ maestro download-samples

✅ Samples downloaded to samples/
rishabh@BT1000102844:~$ cd samples/
rishabh@BT1000102844:~/samples$ ls
Readme.md                    android-flow.yaml        ios-flow.yaml
android-advanced-flow.yaml   ios-advanced-flow.yaml   sample.apk
```

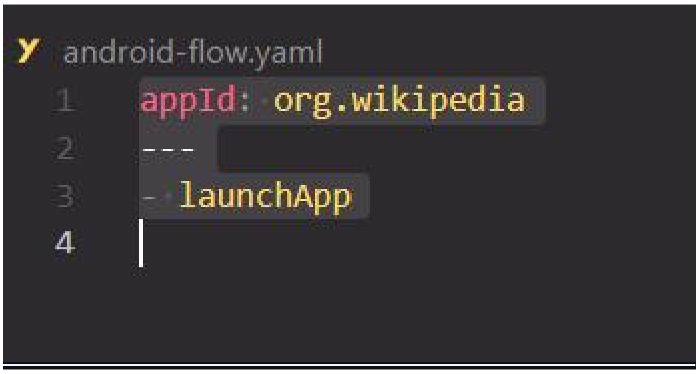


```
rishabh@BT1000102844:~$ cd samples/
rishabh@BT1000102844:~/samples$ maestro --host 192.168.147.135 test android-flow.yaml

  > Flow

    ✅ Launch app "org.wikipedia"

rishabh@BT1000102844:~/samples$ █
```

The Free Encyclopedia
…in over 300 languages

We've found the following on your device:

1. English



android-flow.yaml

```yaml
appId: org.wikipedia
---
- launchApp
```

to run your own app you built you can use the ignite feature now in our app

maestro --host 192.168.147.135 test Login.yaml

THANKING YOU

RISHABH YADAV