Lab – 1

Name: Ishan Aakash Patel

Student ID: 146151238

Course: CYT-230

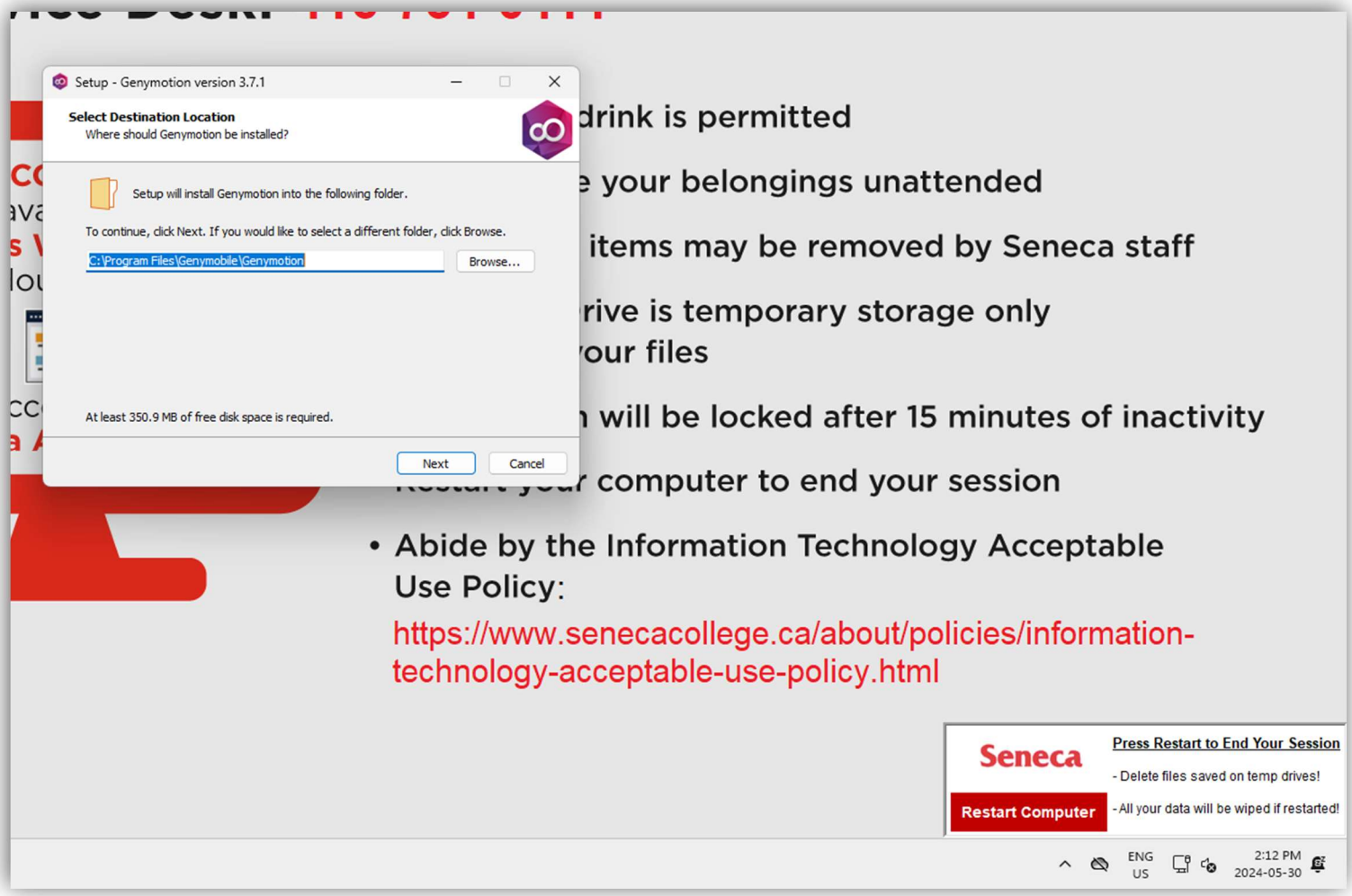
Mobile app Development and Mobile Emulators

Part – 1

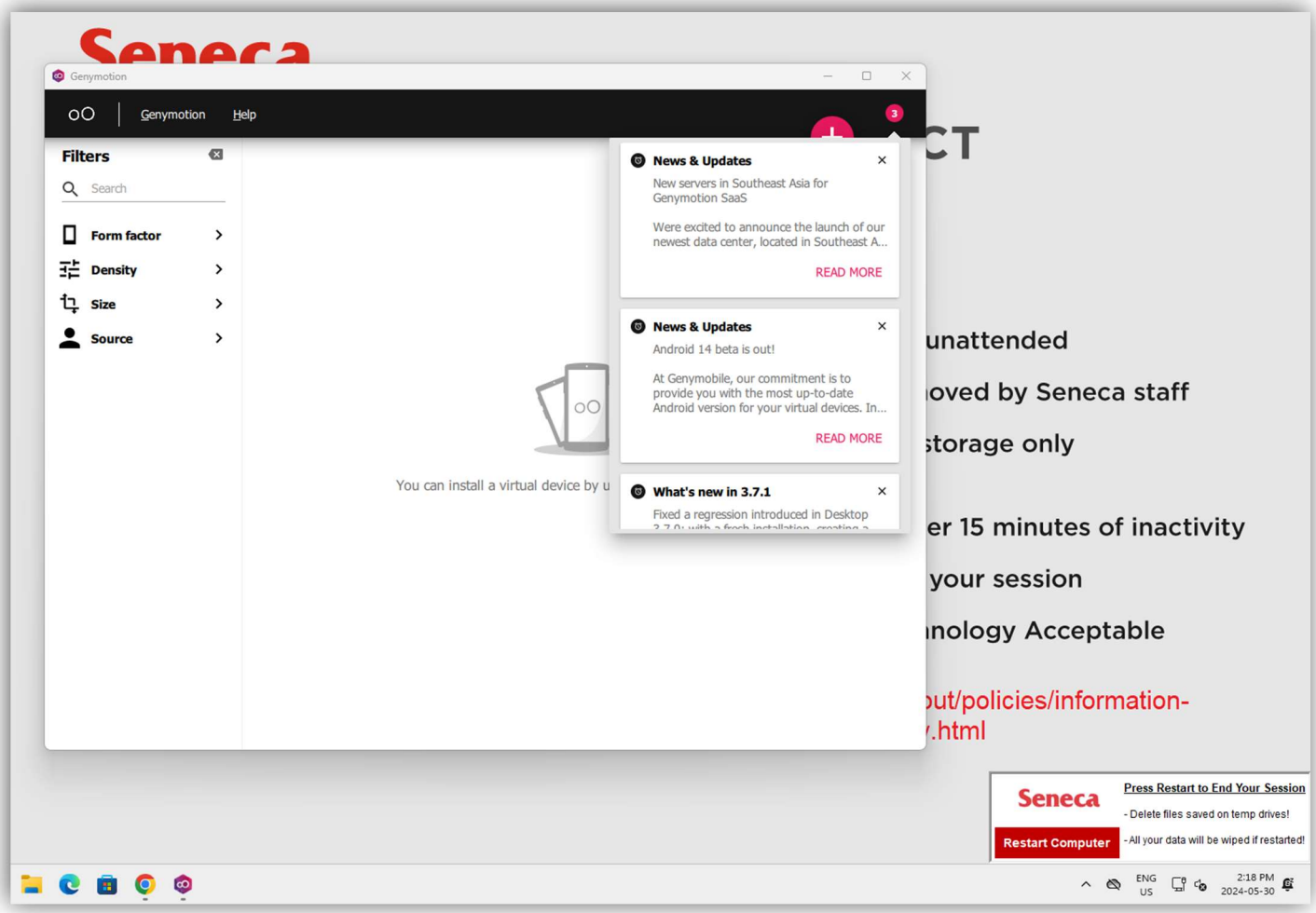
Installing Android Emulators

1) Genymotion

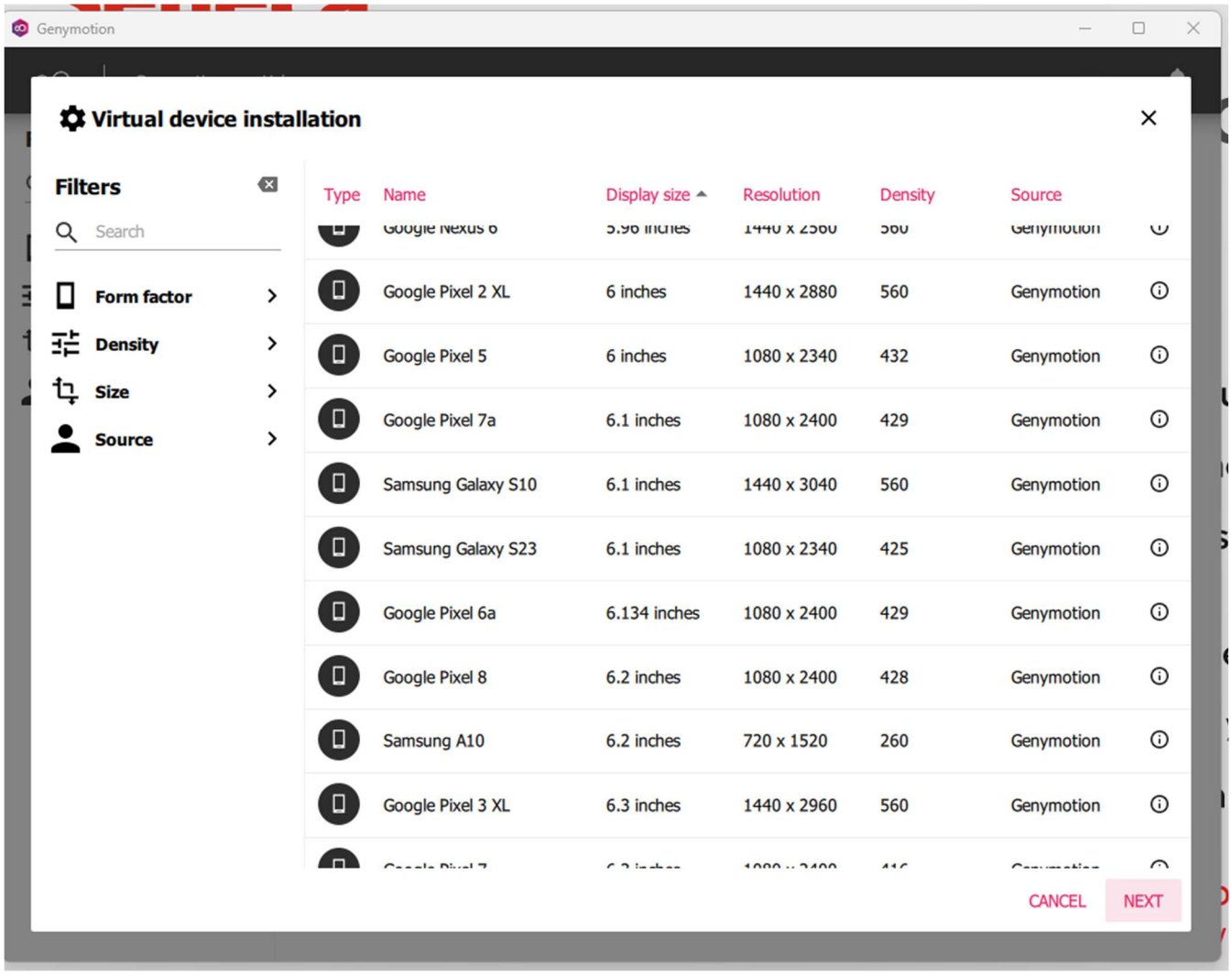
=> The whole part-1, I have completed in Seneca lab computers as my personal laptop was not good enough for part-1.



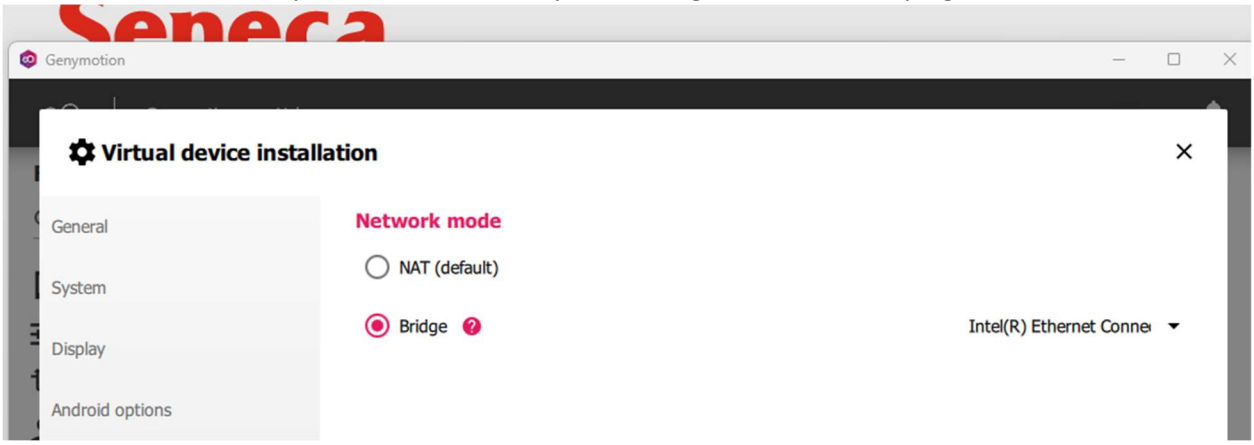

Setup - Genymotion version 3.7.1
Genymotion
Desktop
Completing the Genymotion Setup Wizard
Genymotion has successfully been installed.
VirtualBox needs your computer to be restarted in order to be fully operational.
This is required if:
1. You are installing Genymotion for the first time on this machine.
2. VirtualBox has been upgraded.
Restart now.
I'll restart later.
Finish

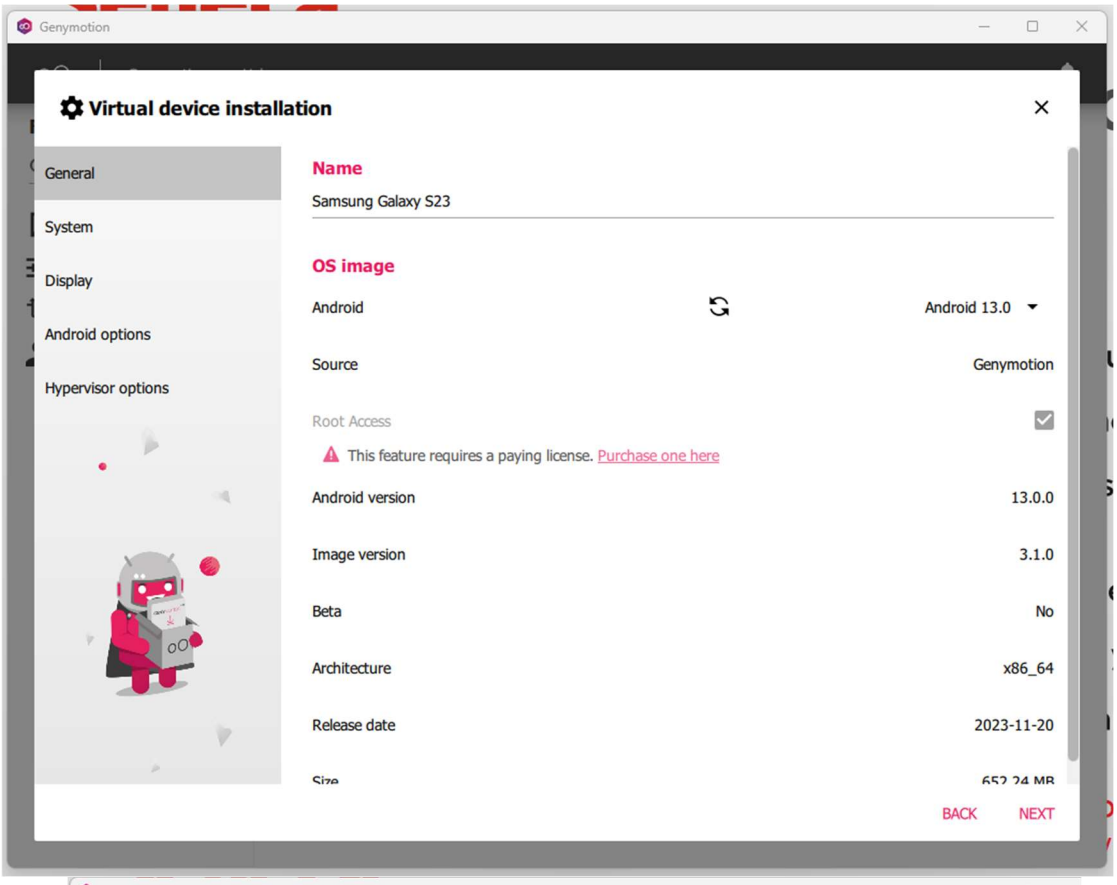

Genymotion
Help
Filters
Search
Form factor
Density
Size
Source
News & Updates
New servers in Southeast Asia for Genymotion SaaS
Were excited to announce the launch of our newest data center, located in Southeast A...
READ MORE
News & Updates
Android 14 beta is out!
At Genymobile, our commitment is to provide you with the most up-to-date Android version for your virtual devices. In...
READ MORE
What's new in 3.7.1
Fixed a regression introduced in Desktop
You can install a virtual device by u
unattended
oved by Seneca staff
storage only
er 15 minutes of inactivity
your session
nology Acceptable
ut/policies/information-
.html
Seneca
Press Restart to End Your Session
- Delete files saved on temp drives!
Restart Computer
- All your data will be wiped if restarted!
ENG
US
2:18 PM
2024-05-30

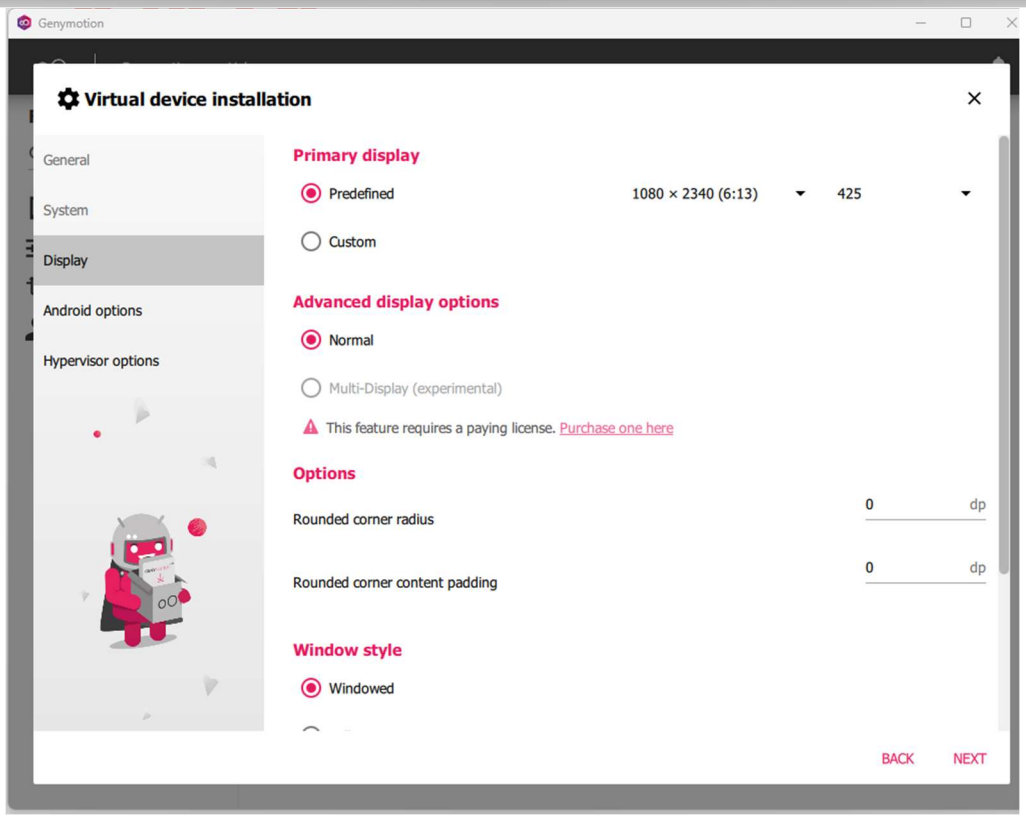
Here are all the android emulators and I selected Samsung Galaxy S23



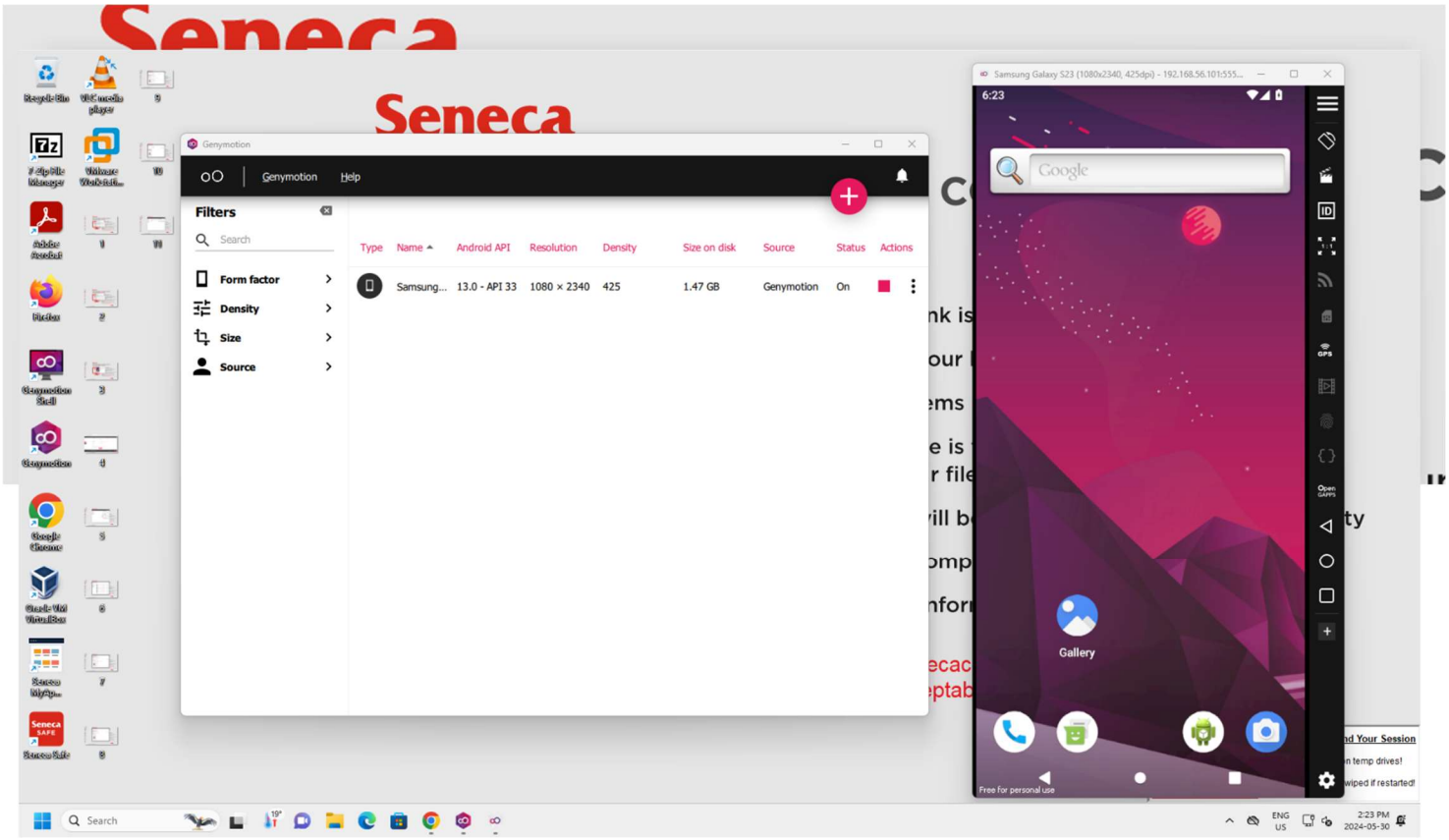
After that select all the specs and network adapter as bridge so that we can ping from the host machine.



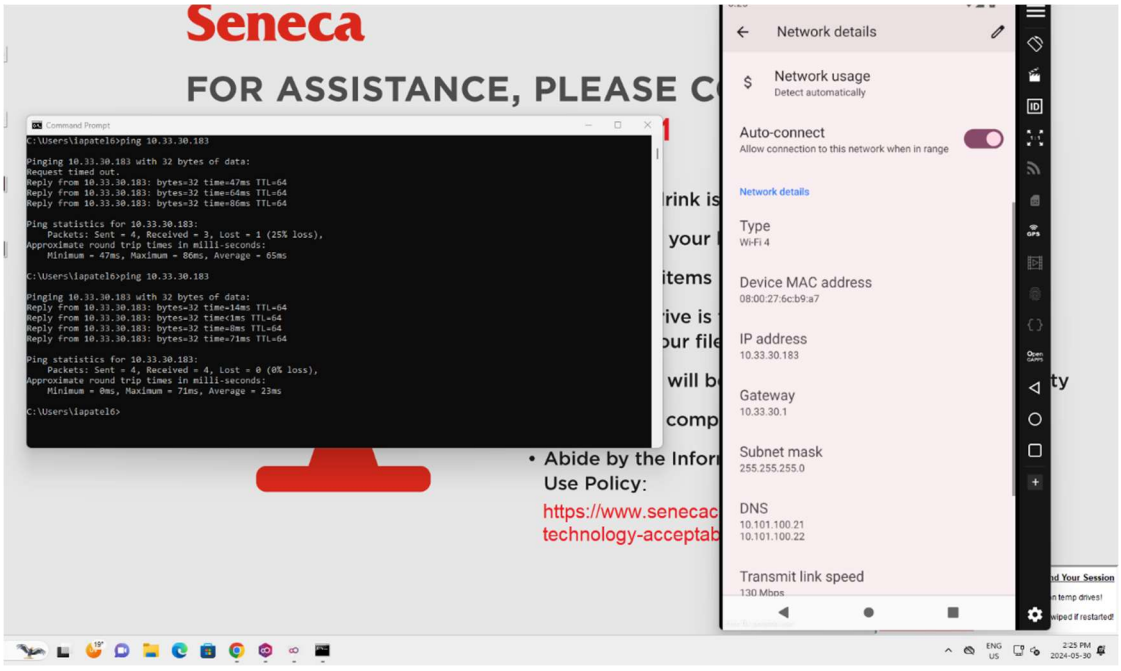

Genymotion
Virtual device installation
General
System
Display
Android options
Hypervisor options
Name
Samsung Galaxy S23
OS image
Android
Android 13.0
Source
Genymotion
Root Access
This feature requires a paying license. Purchase one here
Android version
13.0.0
Image version
3.1.0
Beta
No
Architecture
x86_64
Release date
2023-11-20
Size
652.24 MB
BACK
NEXT


Genymotion
Virtual device installation
General
System
Display
Android options
Hypervisor options
Primary display
Predefined
1080 × 2340 (6:13)
425
Custom
Advanced display options
Normal
Multi-Display (experimental)
This feature requires a paying license. Purchase one here
Options
Rounded corner radius
0
dp
Rounded corner content padding
0
dp
Window style
Windowed
BACK
NEXT

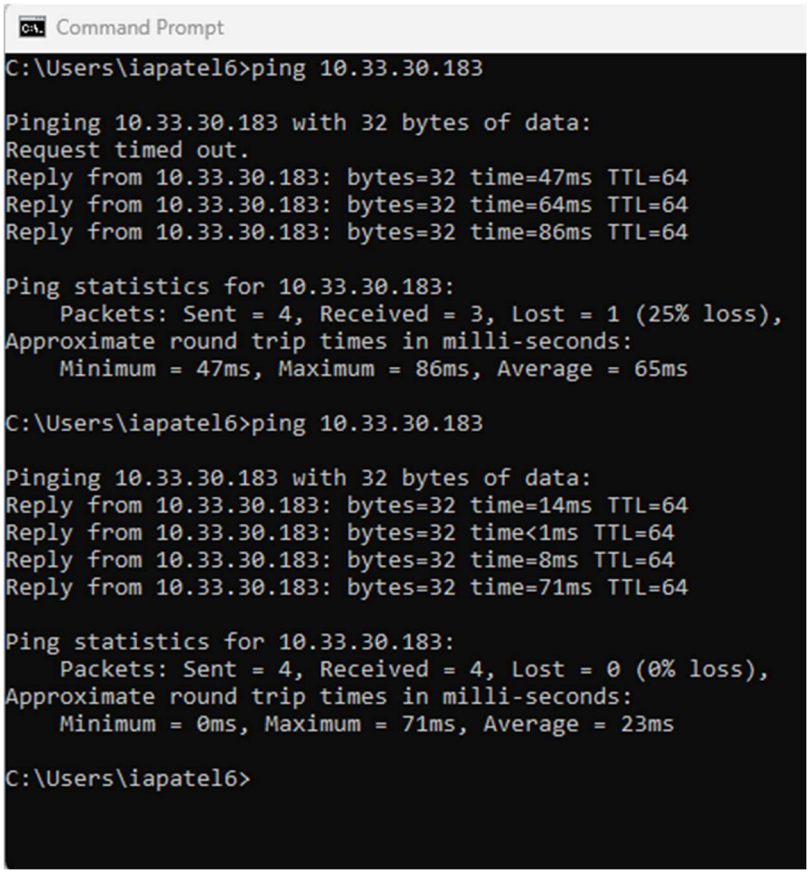
Installation done

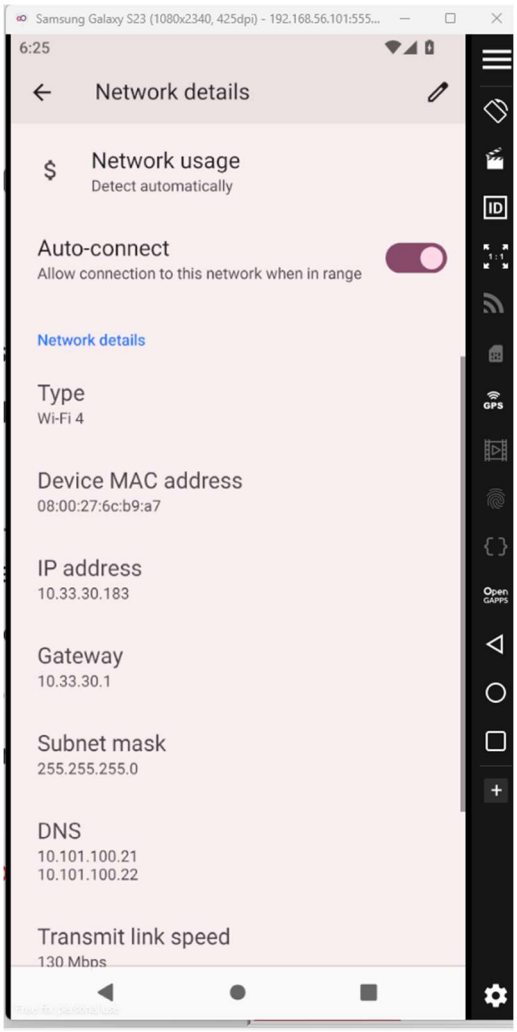


As you can see I can ping from my host machine to the emulator.



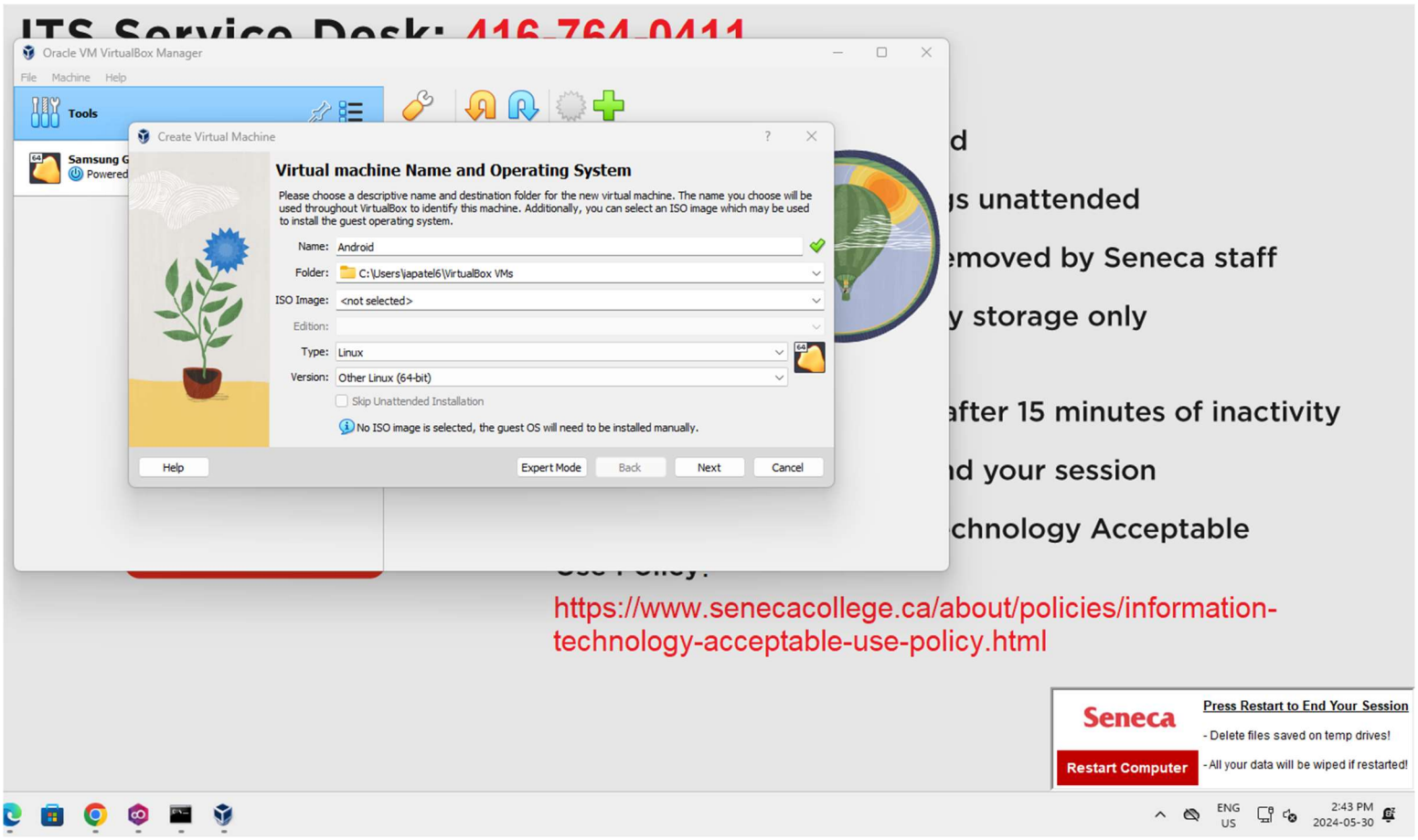
Here is the magnified version of the above screenshot.

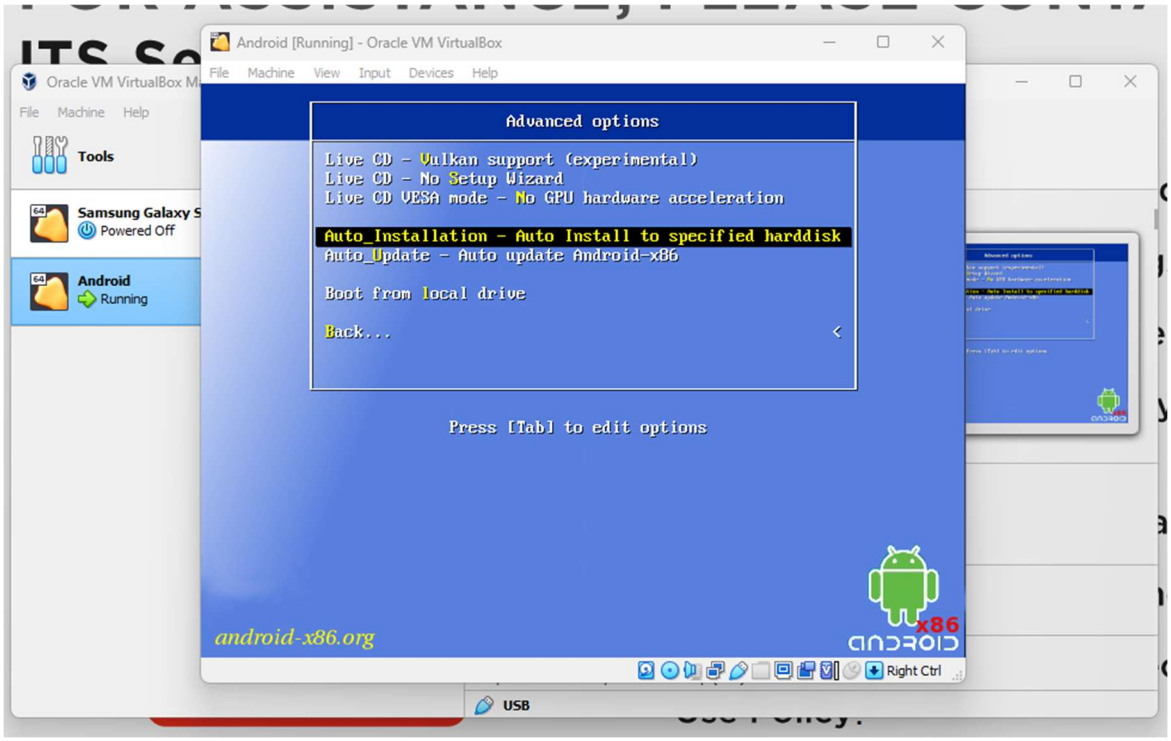


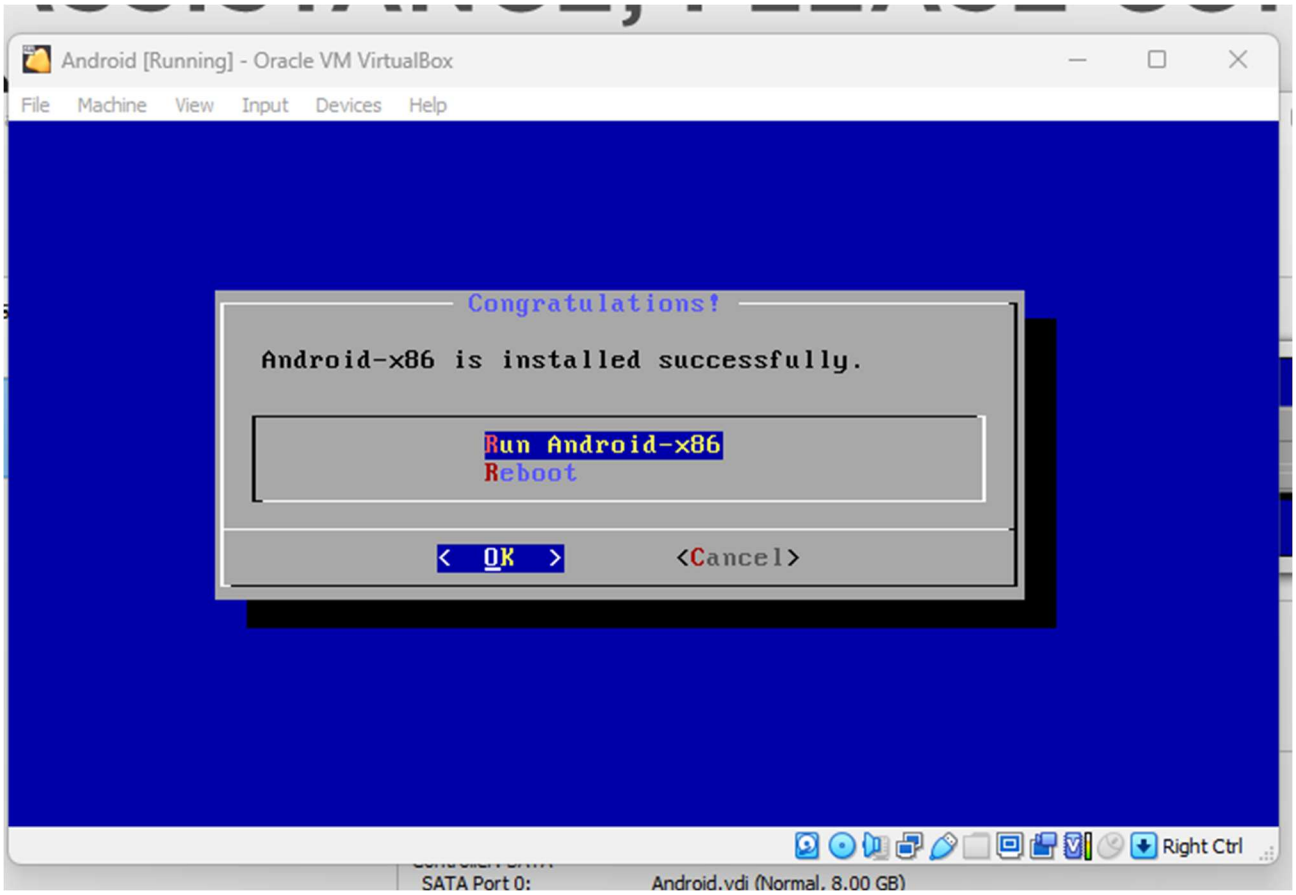


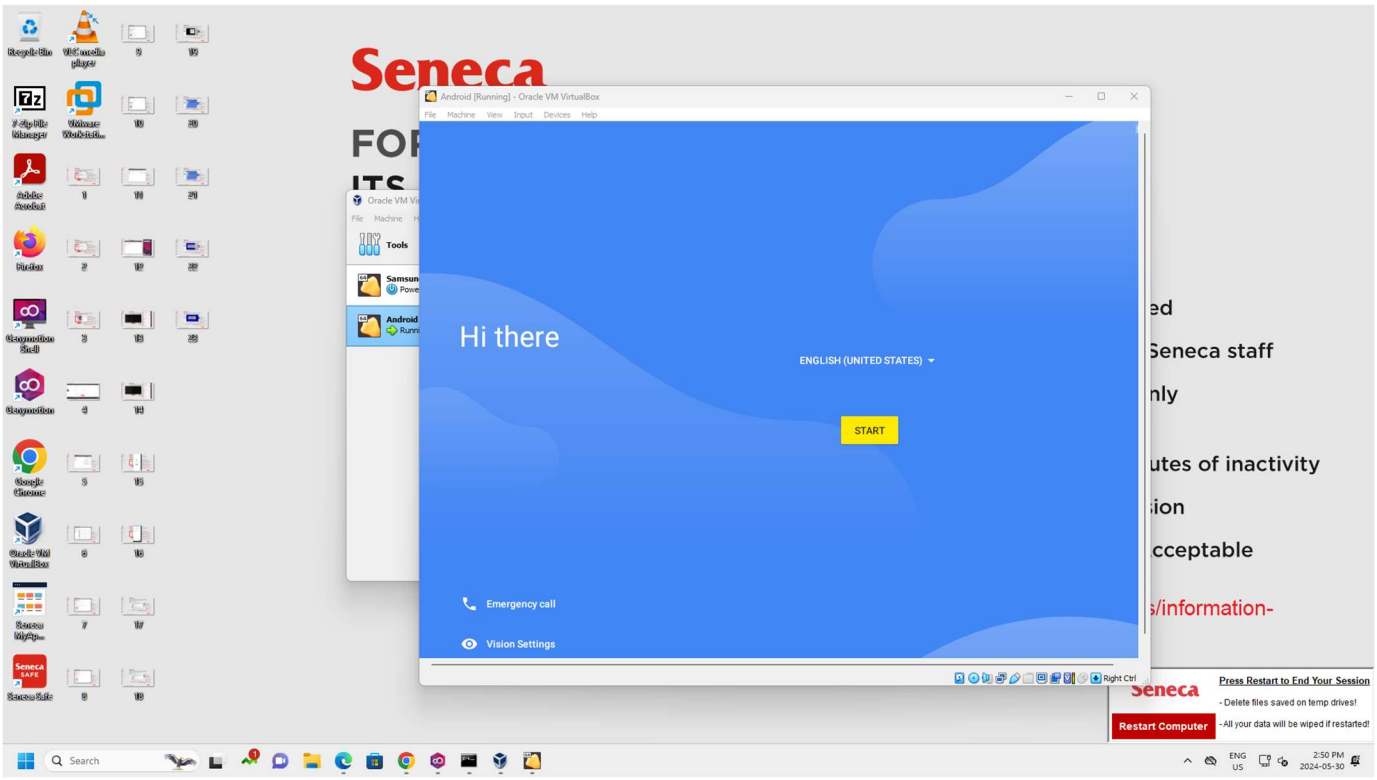
This was all for the genymotion and we will install the second emulator in the virtualbox name – android x86

2) Android x86

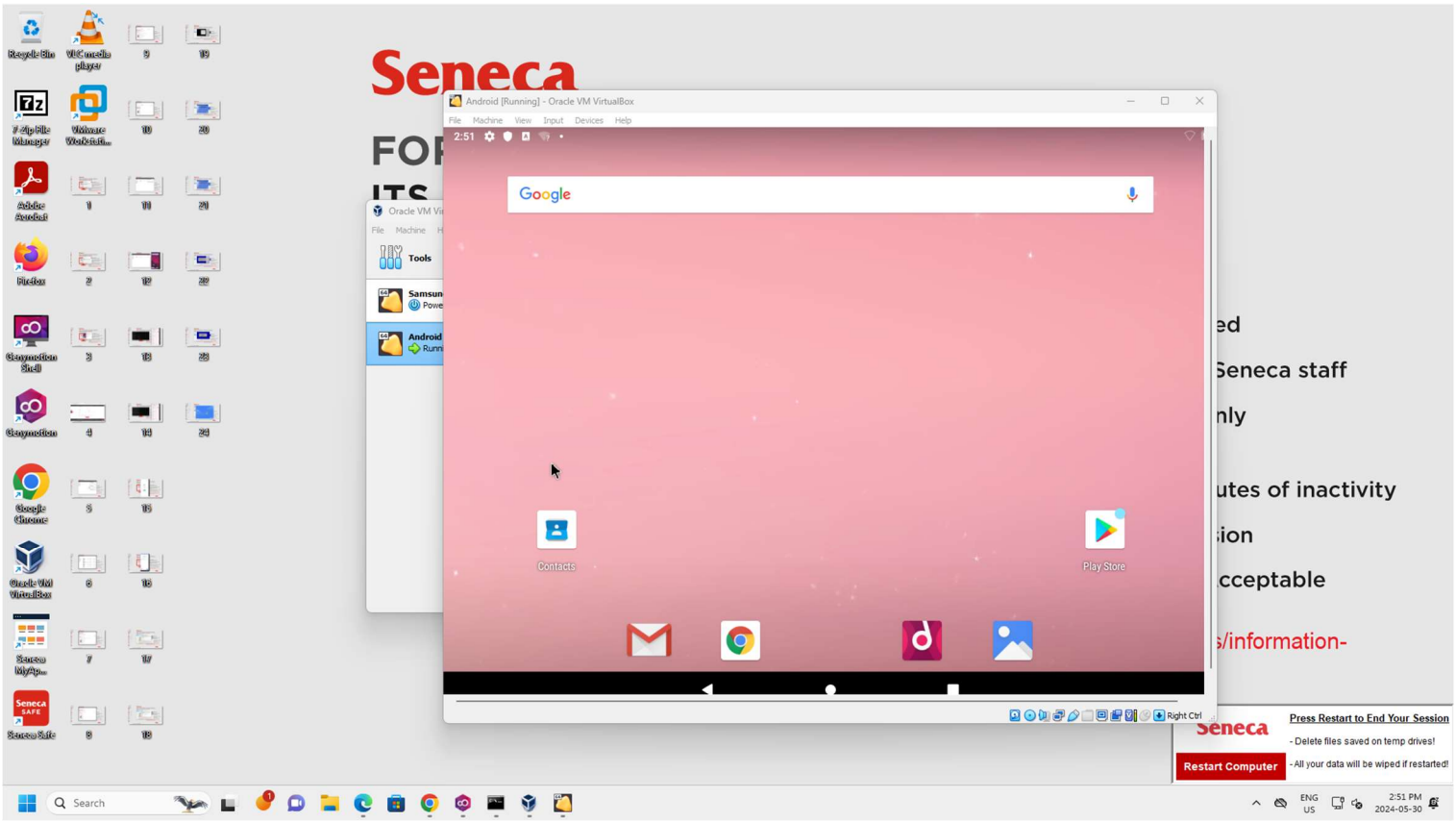




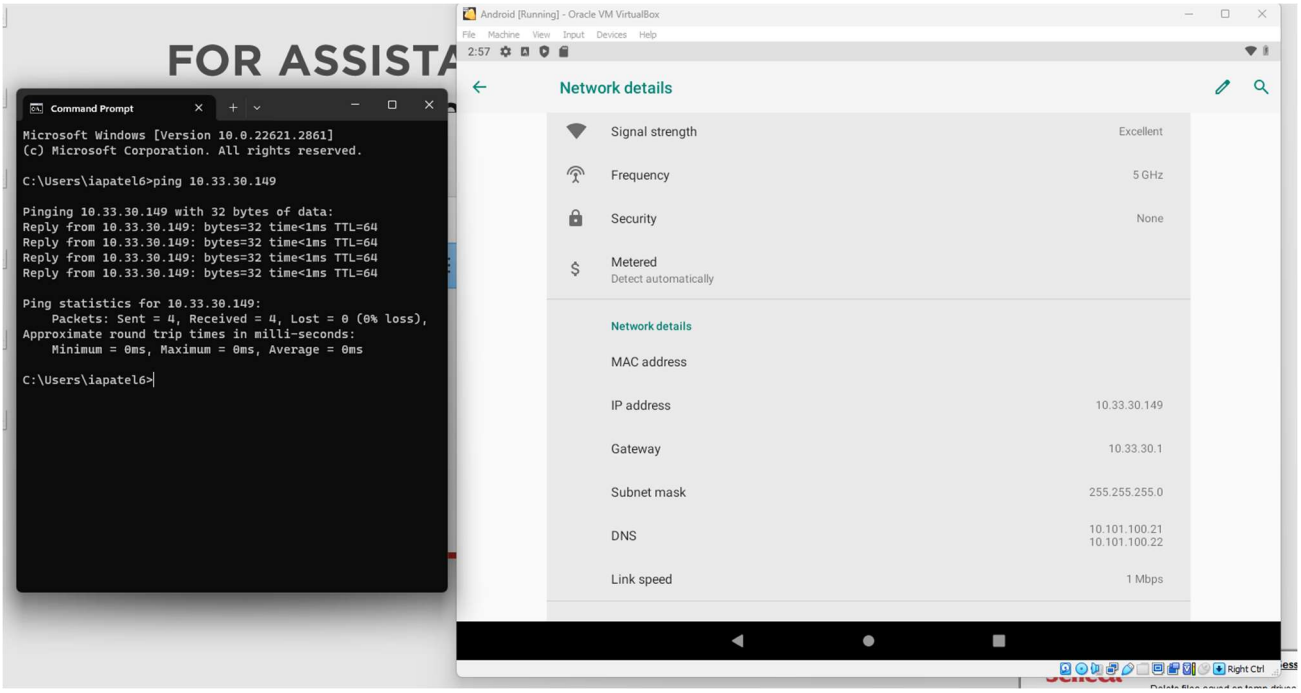

Android [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Congratulations!
Android-x86 is installed successfully.
Run Android-x86
Reboot
< OK >
<Cancel>
Right Ctrl
SATA Port 0:
Android.vdi (Normal, 8.00 GB)


Seneca
Android [Running] - Oracle VM VirtualBox
Hi there
ENGLISH (UNITED STATES)
START
Emergency call
Vision Settings
Right Ctrl
Press Restart to End Your Session
- Delete files saved on temp drives!
- All your data will be wiped if restarted!
Restart Computer
Search

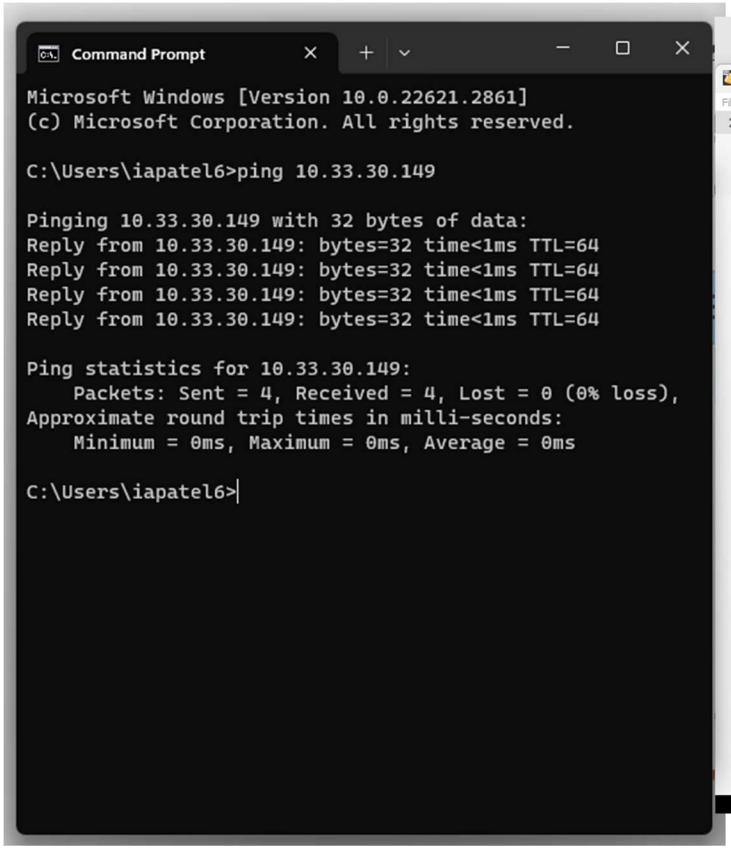
As you can see the VM is perfectly running.

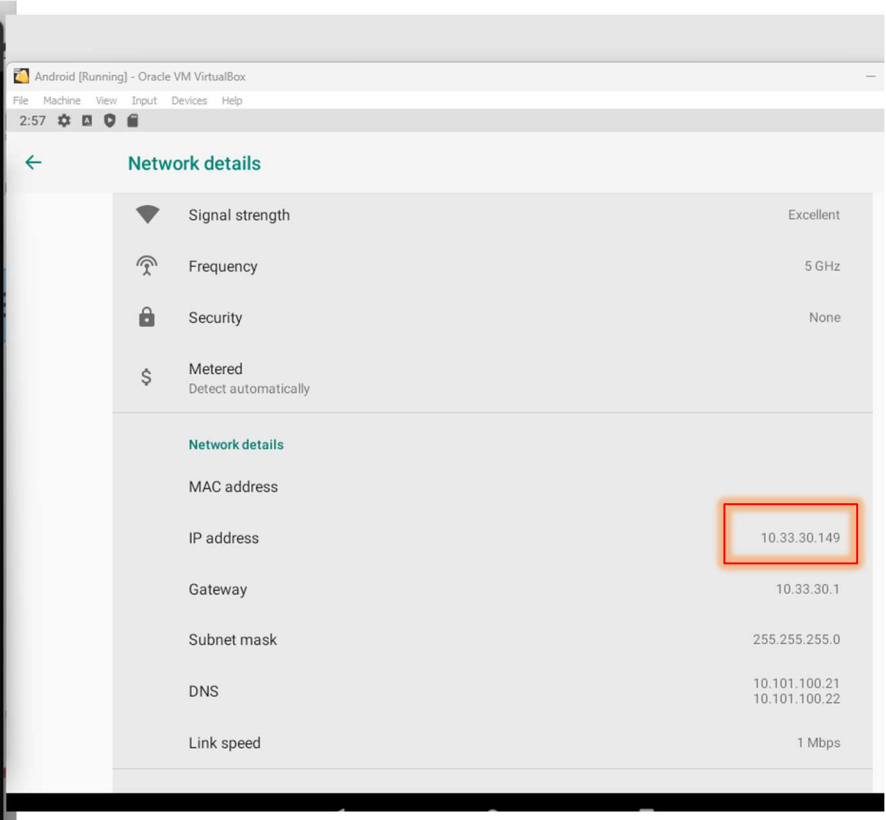


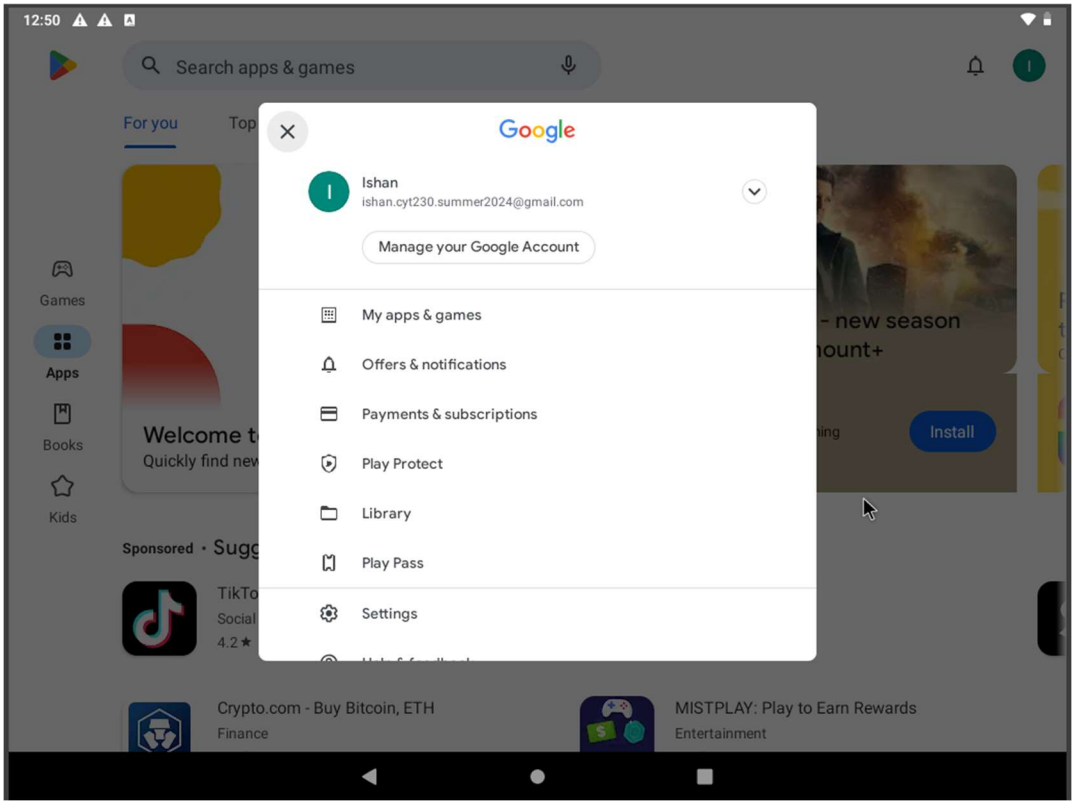
We can also ping the android emulator from the host machine.



Here is the magnified version of the above screenshot.





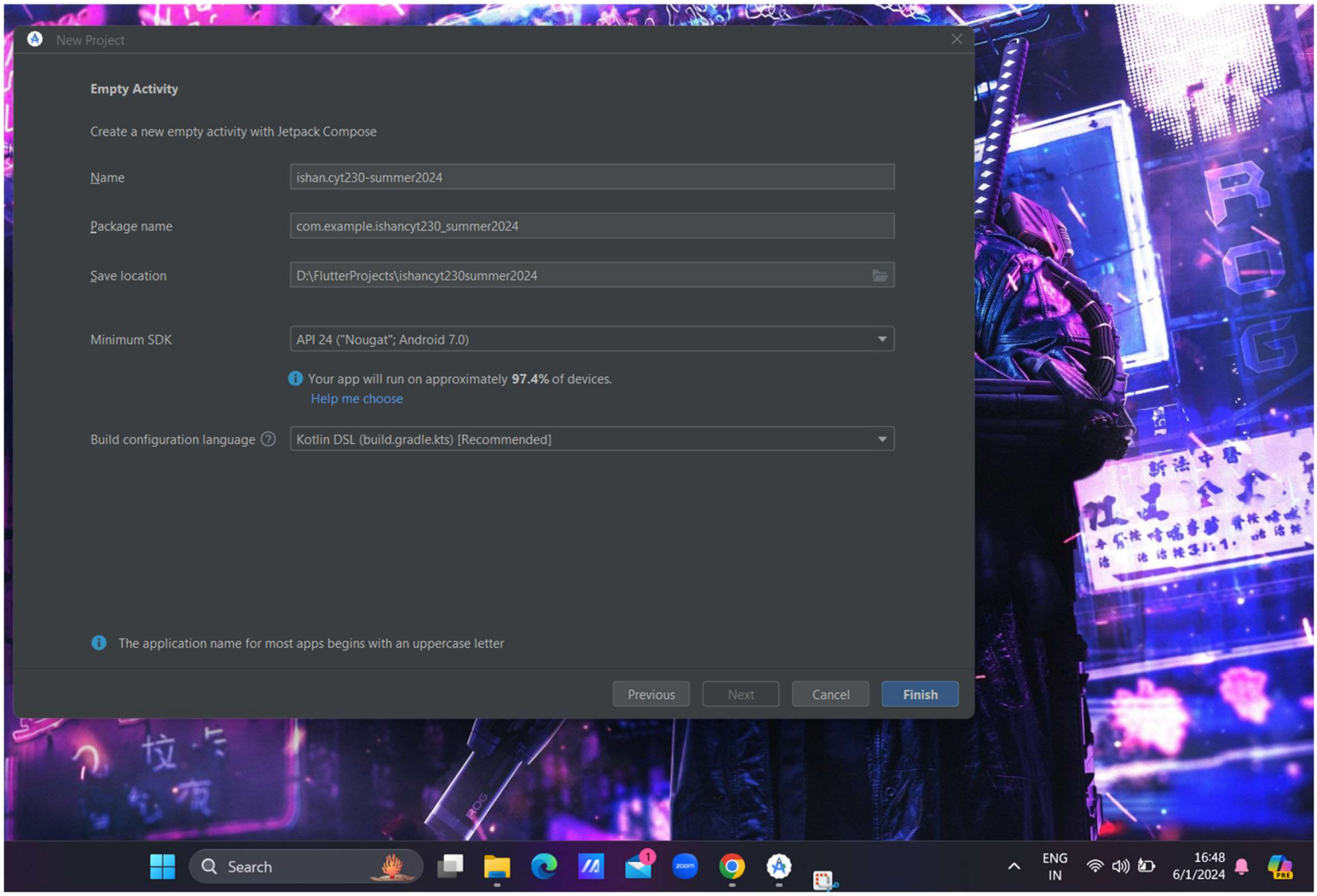


Here is the google play store account with the proper naming convention.

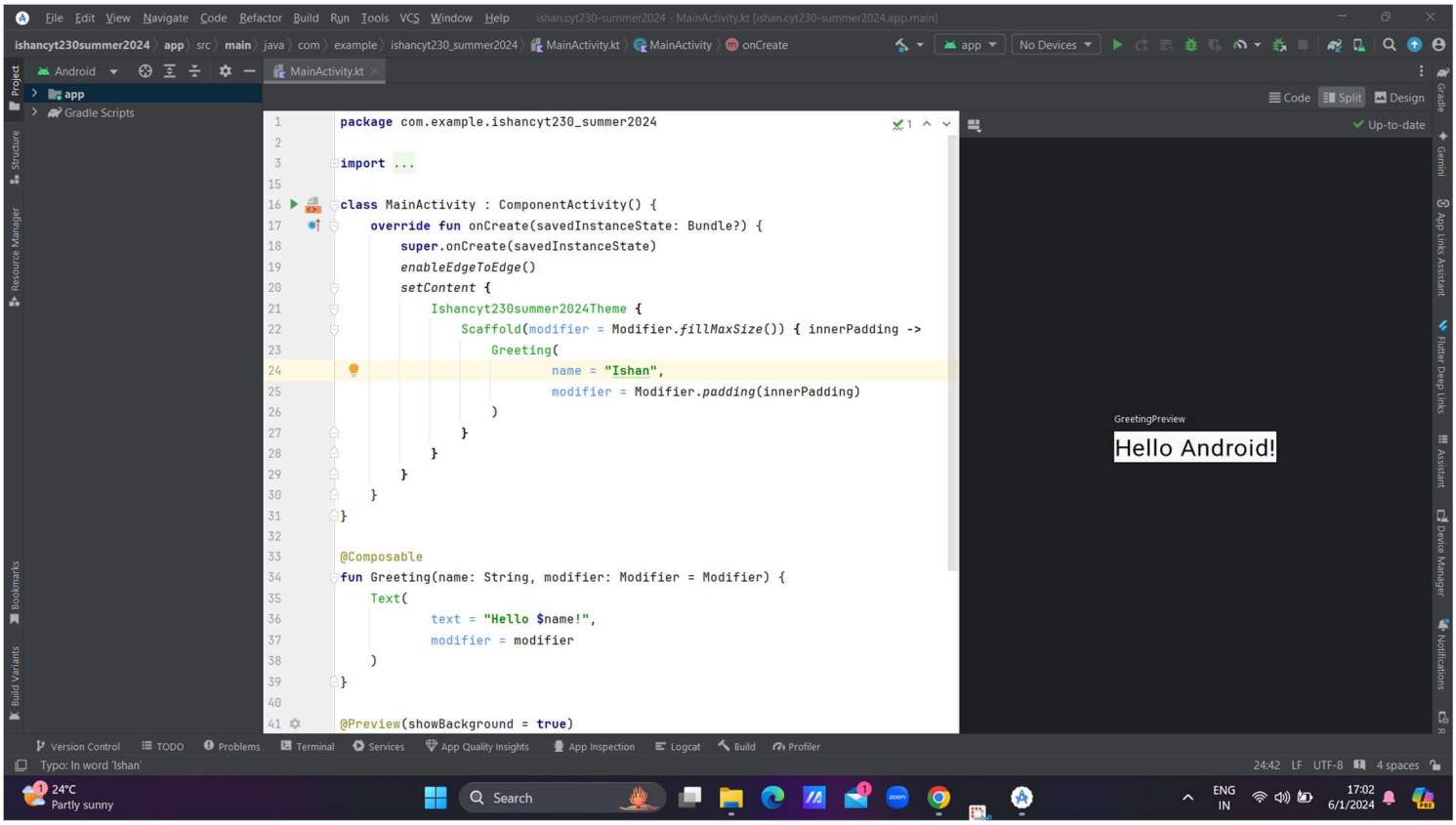
Part – 2

Android Studio

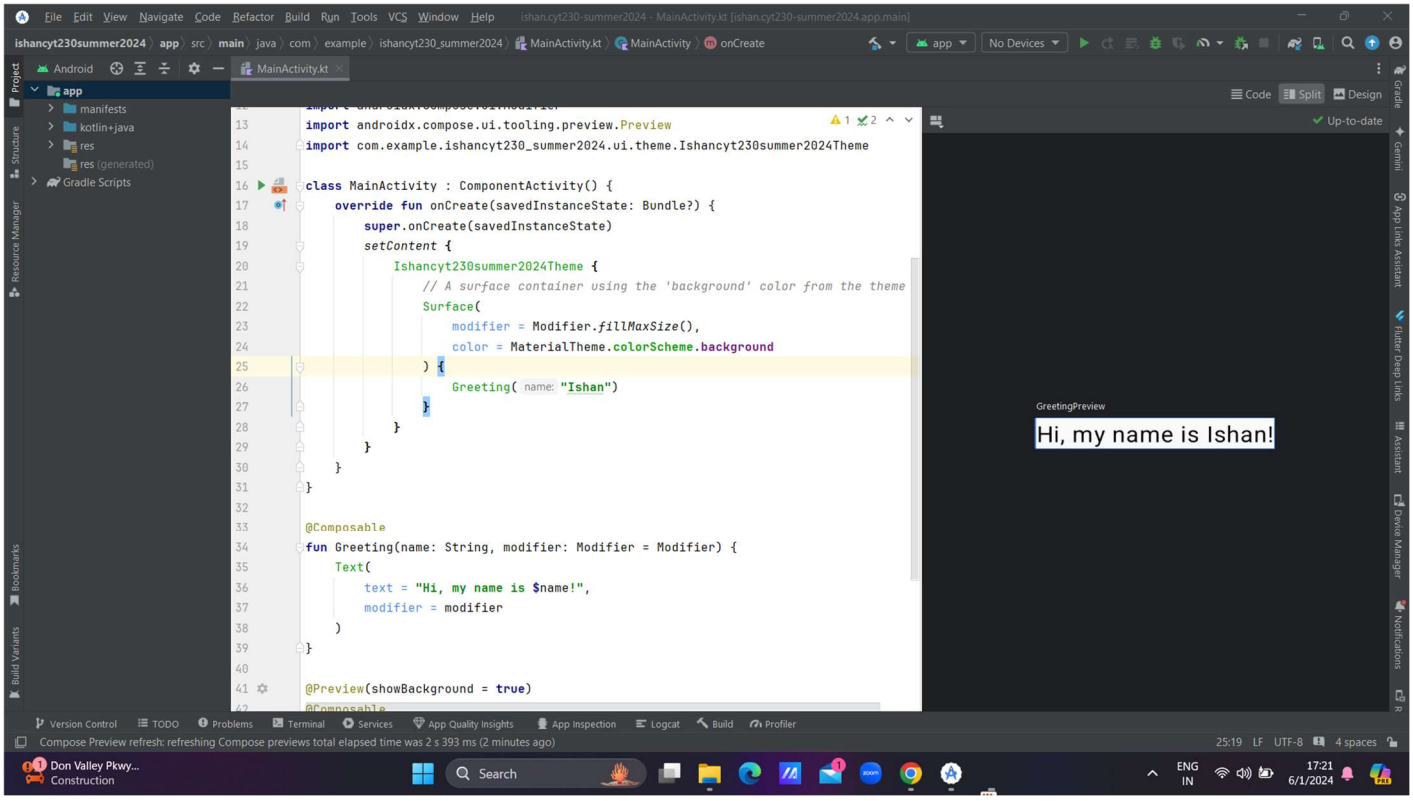
I installed android studio and created the project with the proper naming convention.



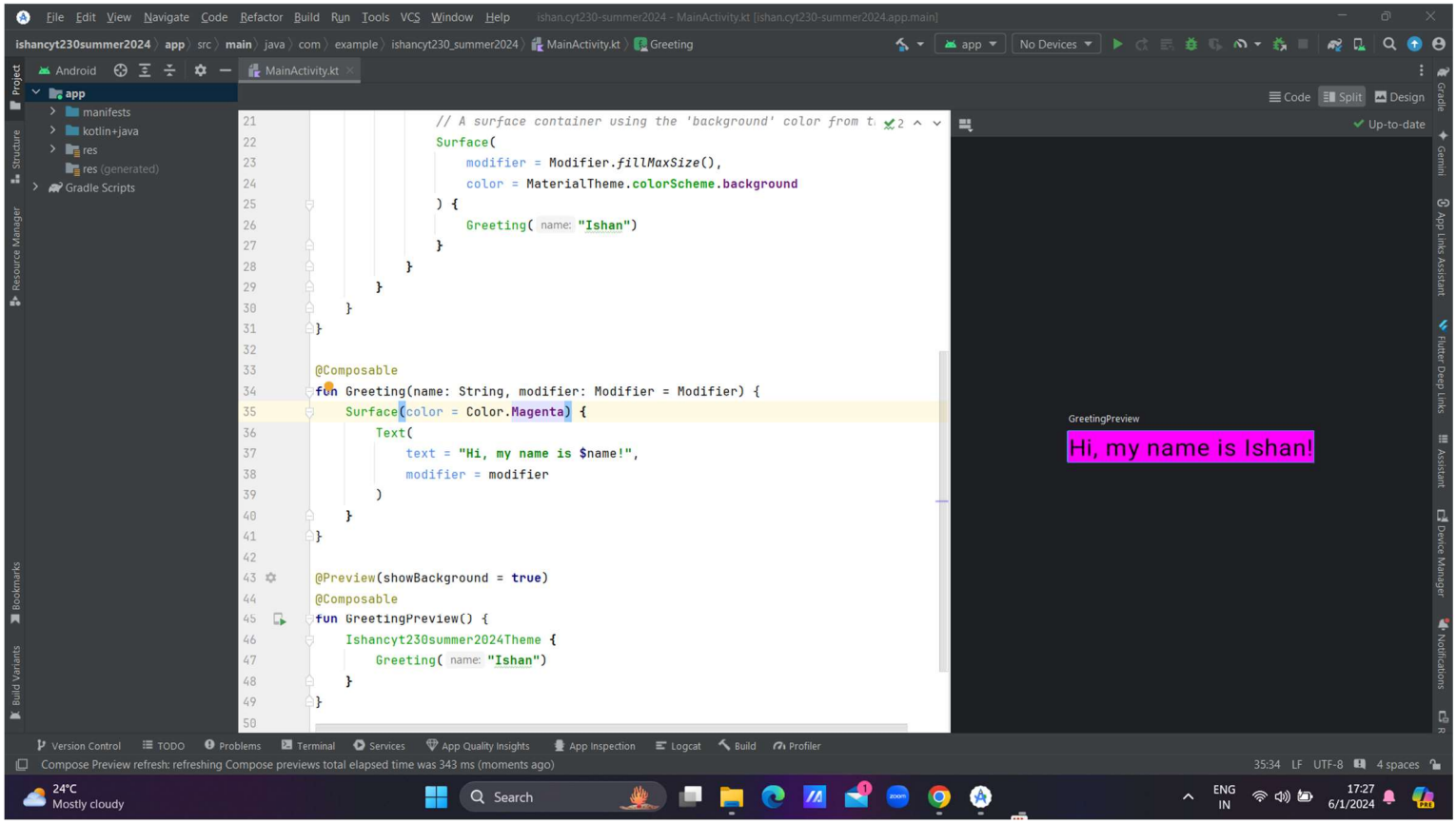
I followed all the steps which were presented in the link one and below is the results.

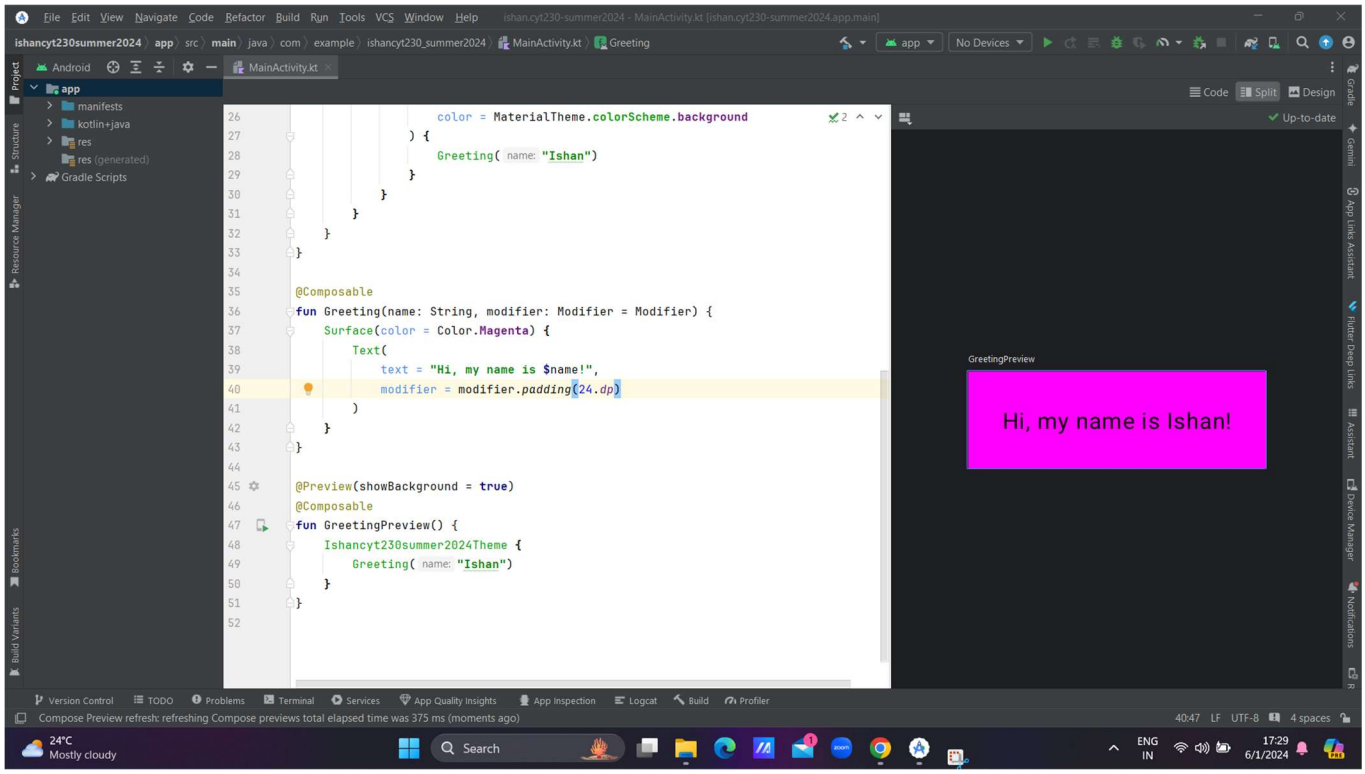


First, I replaced the text with my name.

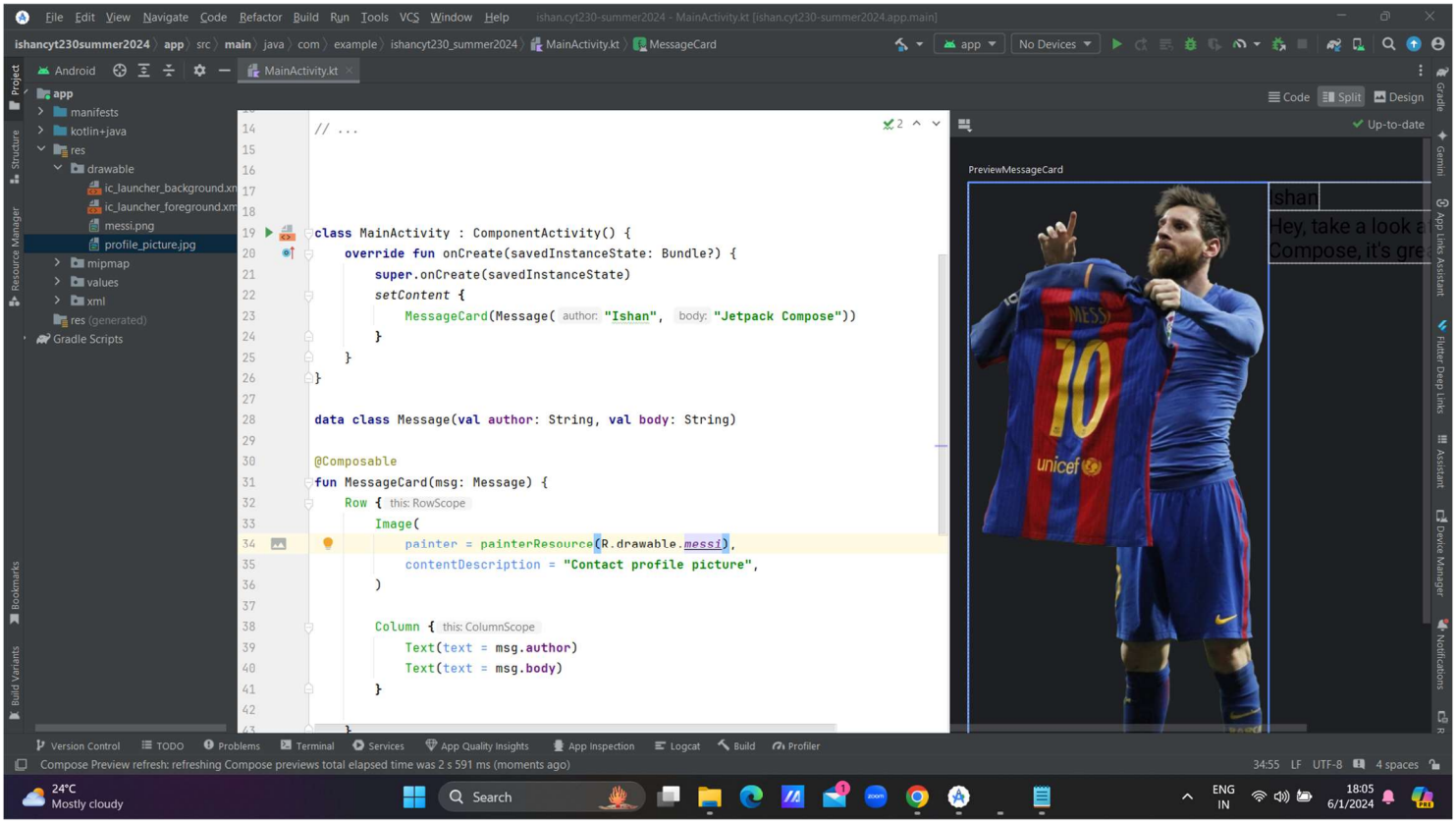


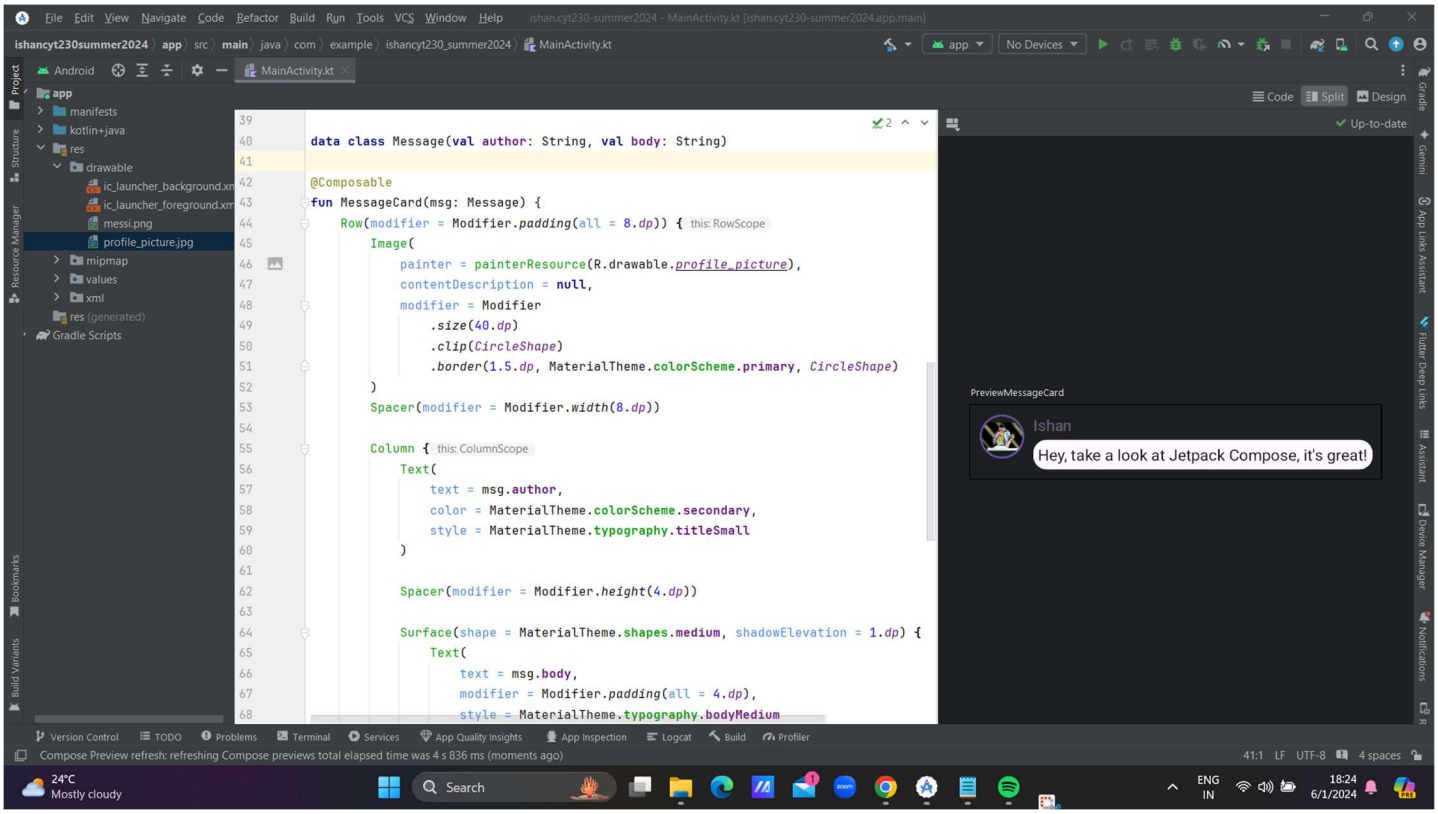
Then I changed the color and also added some padding.



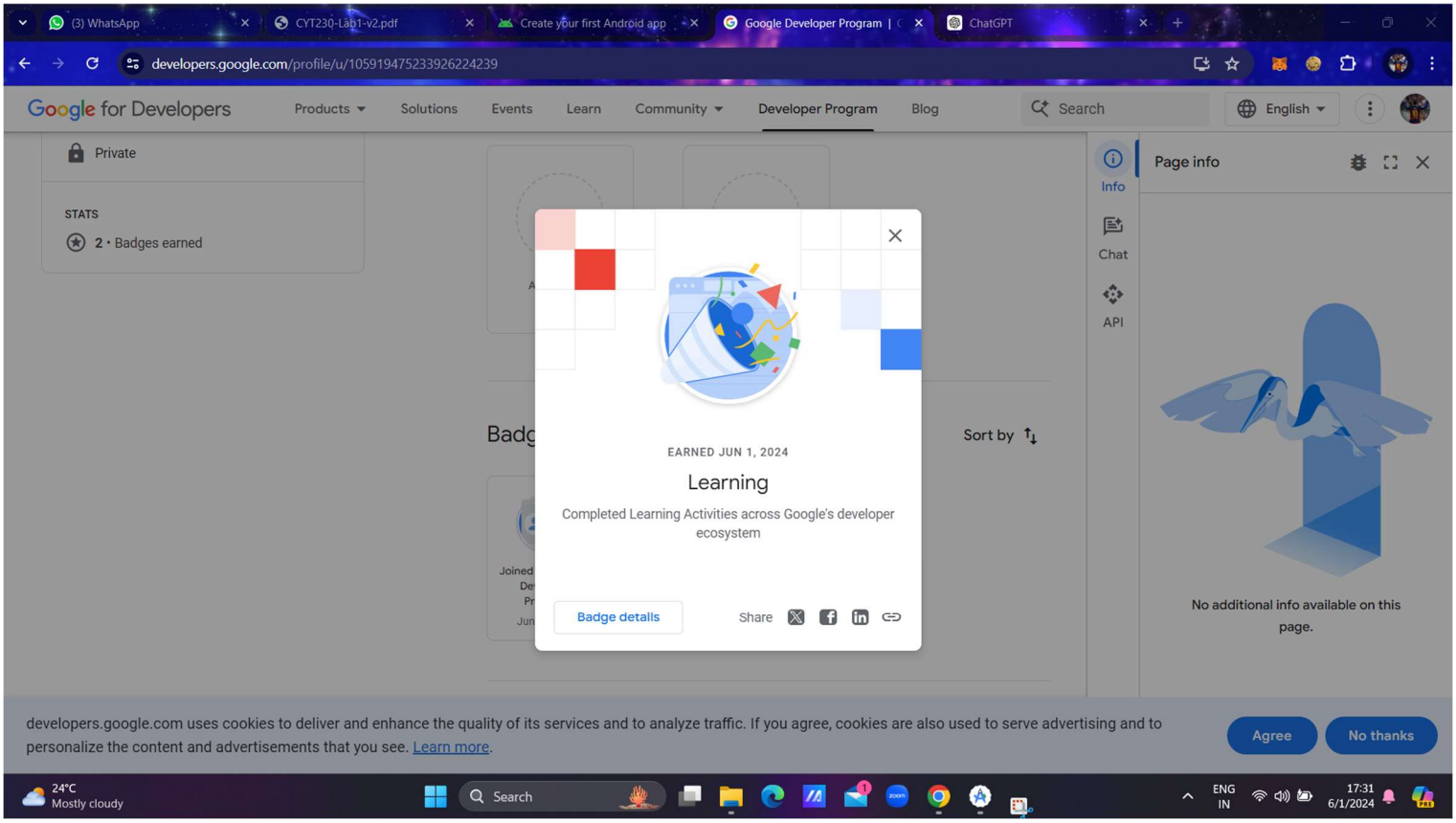


Next, I tried to play around with the UI and here are few results.





Here is the badge I got when I completed all the steps.



Part – 3

App Manifest

Step 1: Install ADB Tool on the Host Machine

First, you need to install the Android Debug Bridge (ADB) tool on your host machine.

Note: Don't forget to turn on USB debugging in the emulator.

Step 2: Connect to the Android Emulator

Next, you need to connect to your Android emulator. Ensure your emulator is running.

Open a terminal or command prompt.

Use the following command to list all connected devices:

Command: adb devices

Step 3: Locate the base.apk File

Now, locate the base.apk file within the emulator. The base.apk file is usually found in the /data/app/ directory of the emulator.

Start a shell session on the emulator:

Command: adb shell

Navigate to the directory containing the APK files. Typically, this is in /data/app/:

Command: cd /data/app/ ls

This will list the directories of installed applications.

Find the directory corresponding to your app. It will typically be in the form of package.name-1.

Step 4: Pull the base.apk File to the Host Machine

Use the adb pull command to transfer the APK file to your host machine.

Exit the shell session by typing exit.

Pull the APK file to your host machine:

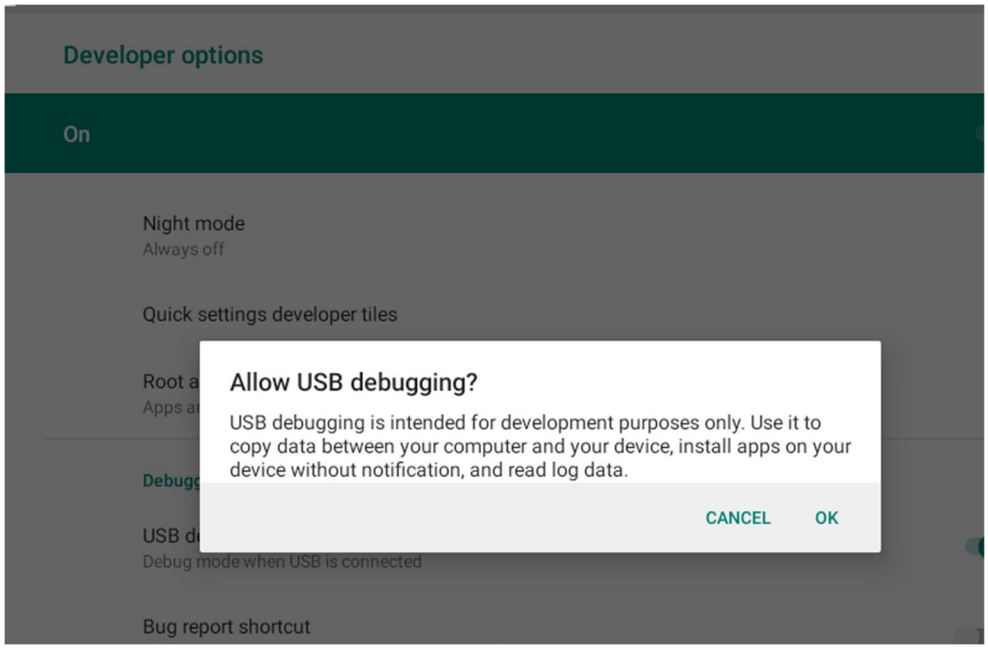
Command: adb pull /data/app/package.name-1/base.apk /path/to/save/base.apk

Step 5: Decompile the APK File

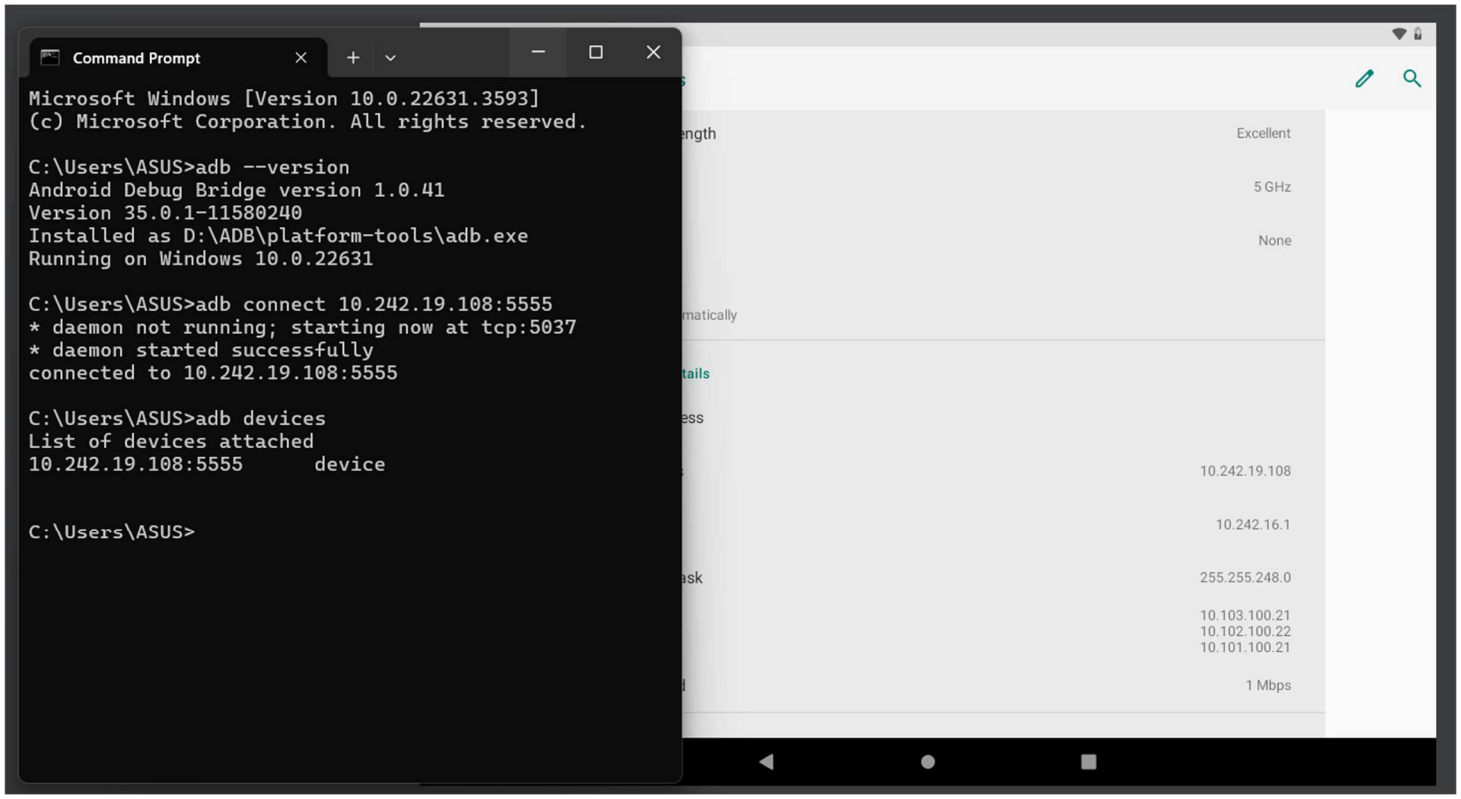
Decompile the APK file to access its contents, including the AndroidManifest.xml file. You can use an online APK decompiler or a tool like JADX.

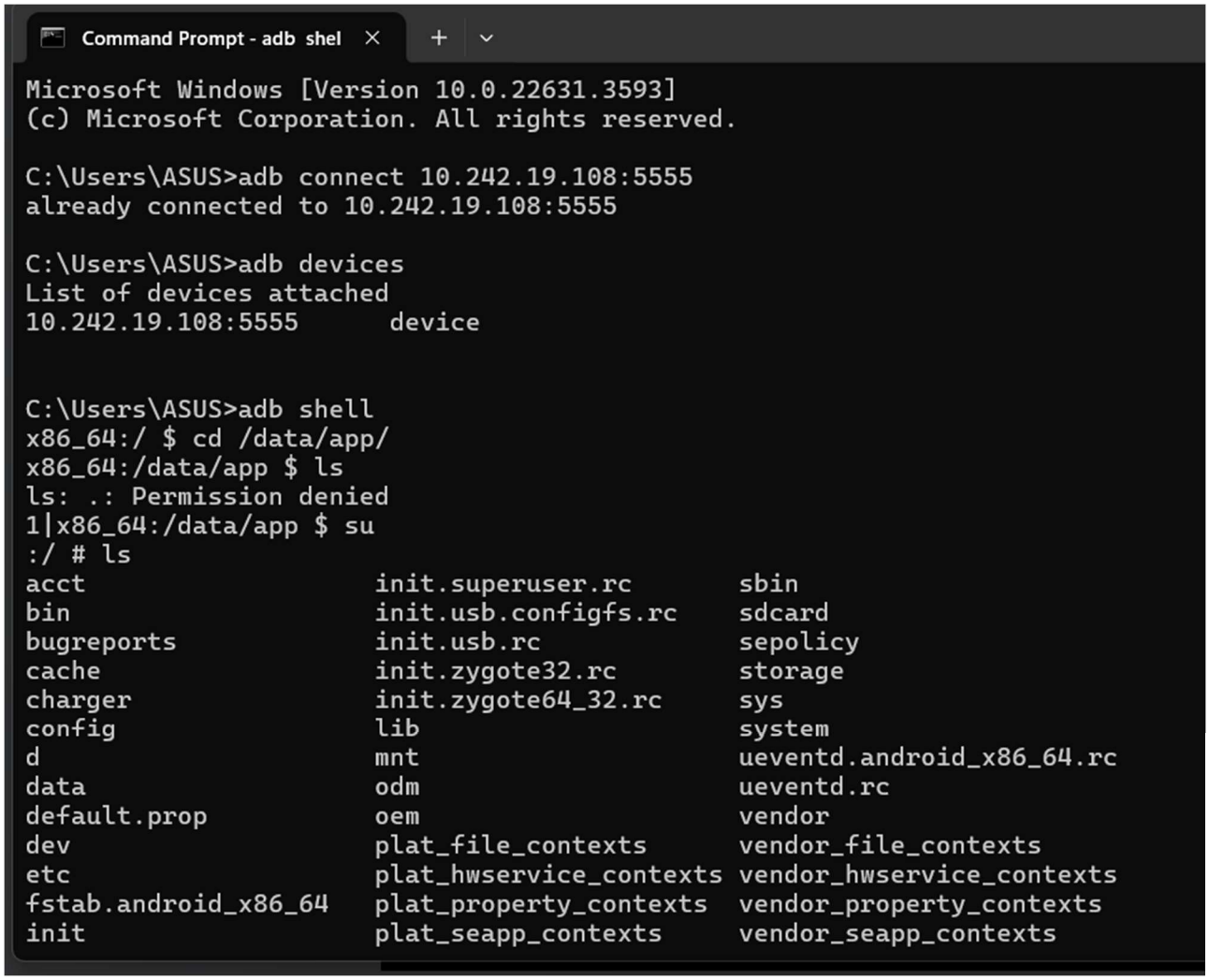
I used an Online Decompiler

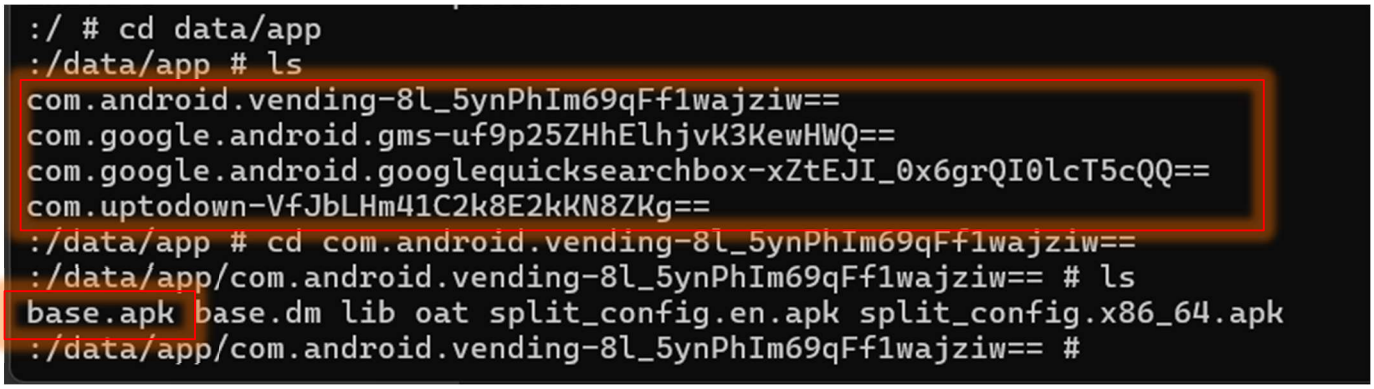
USB debugger turned on



Now connect to the android VM with cmd in your host machine.



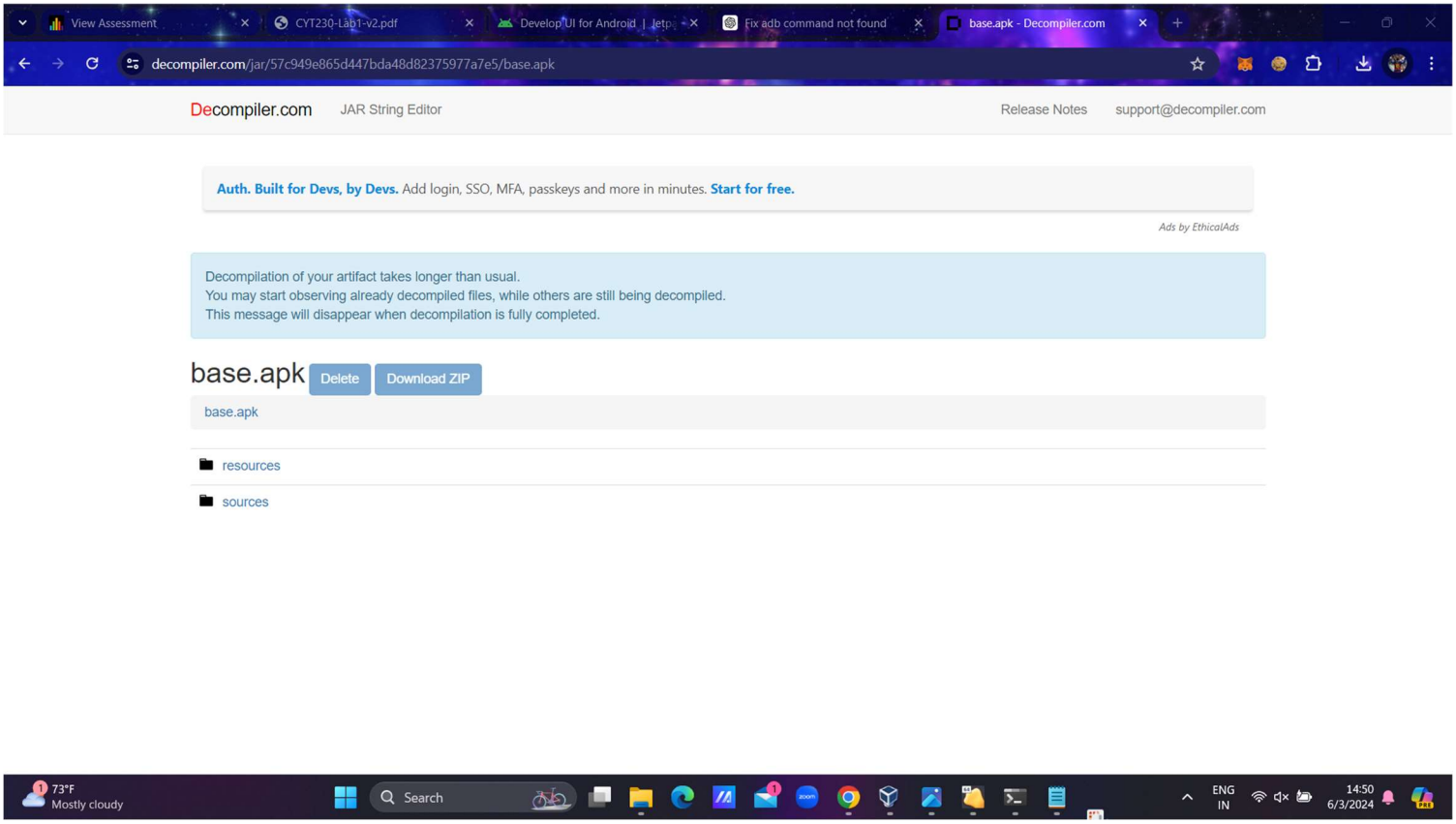




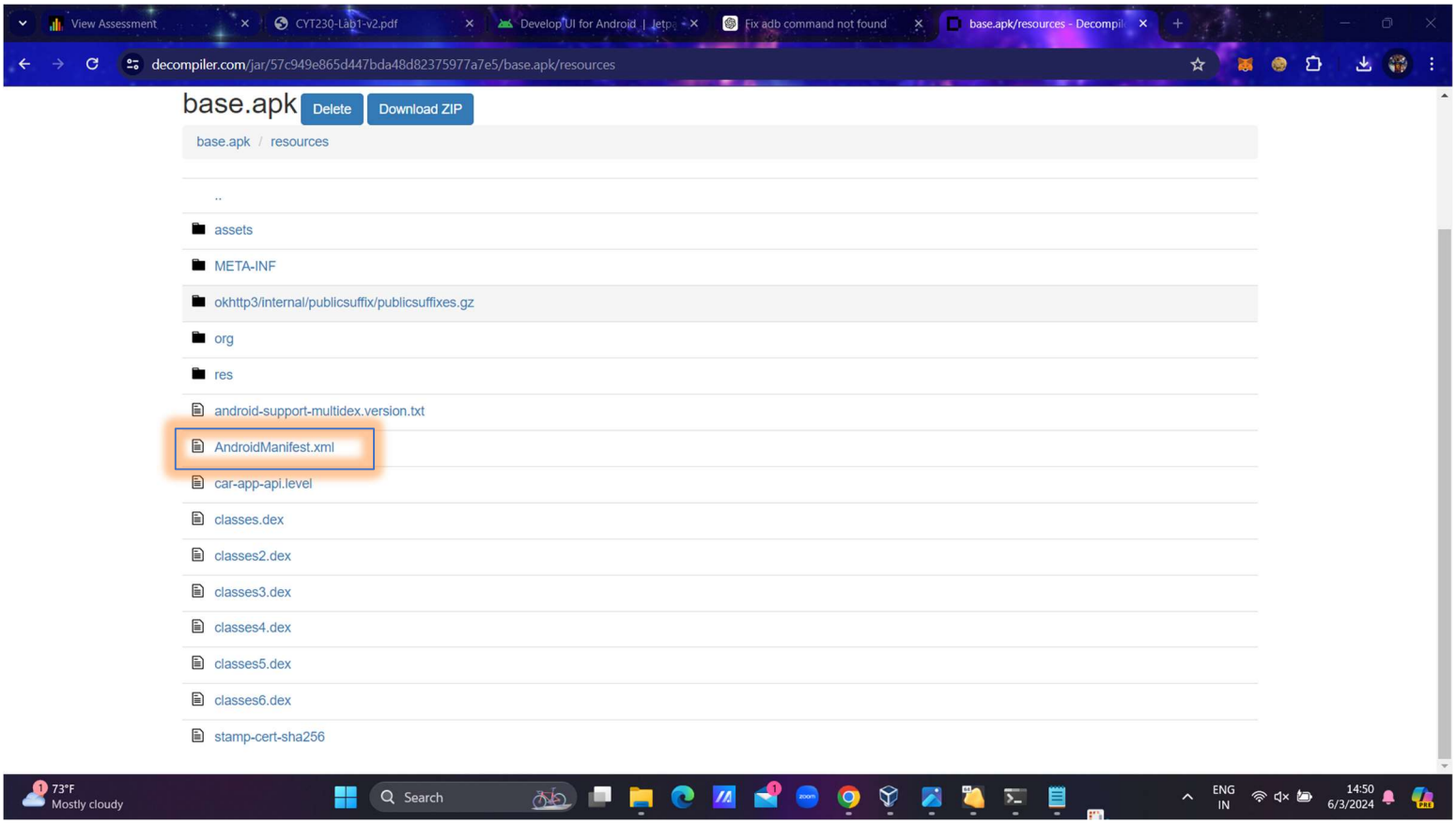
These are the applications I found in the android and I selected the first one. And in the last second line you can see that there is the base.apk file, now I need to pull this file to the host machine.

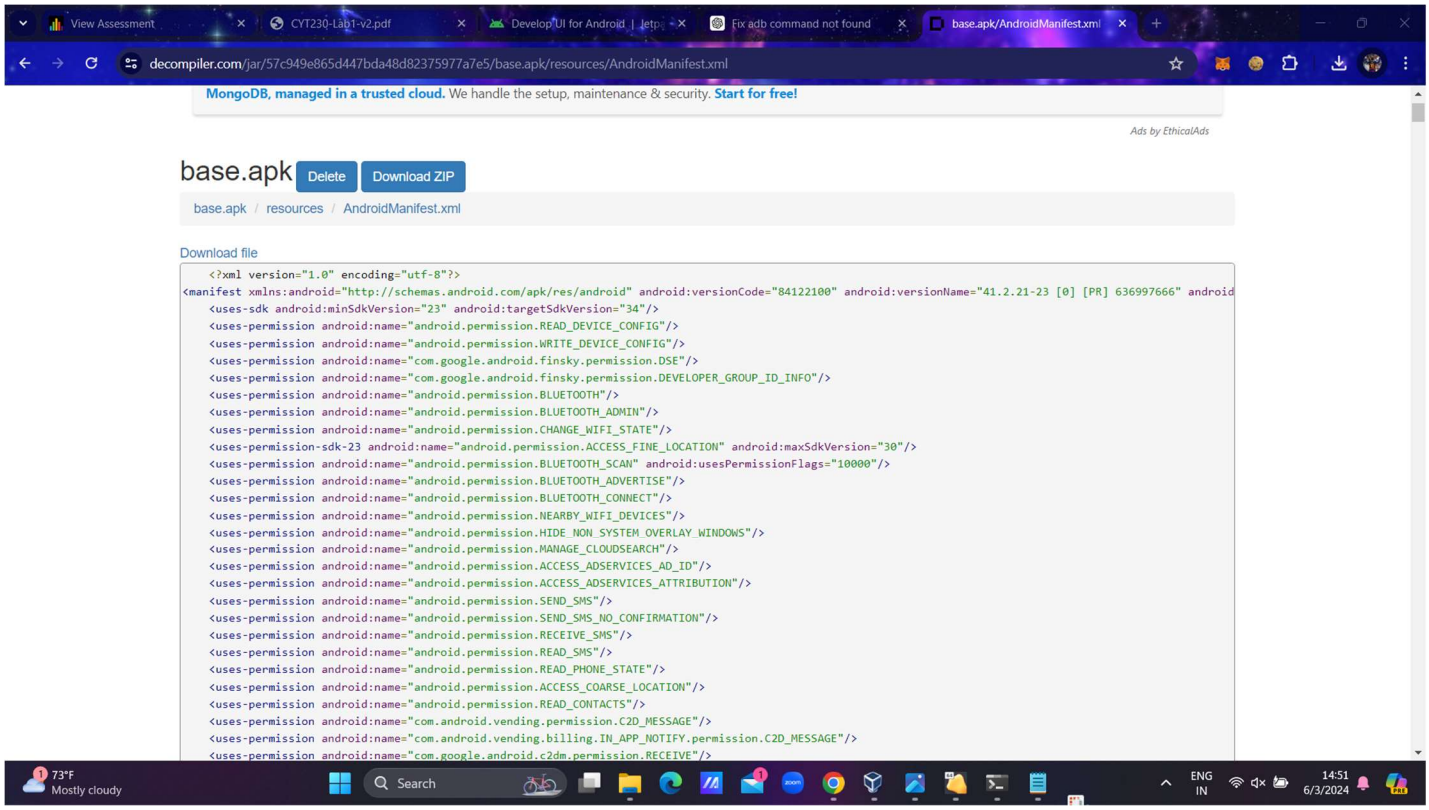
```
C:\Users\ASUS>adb pull /data/app/com.android.vending-8l_5ynPhIm69qFf1wajziw==/base.apk C:\Users\ASUS\Desktop
/data/app/com.android.vending-8l_5ynPhIm69qFf1wajziw==/bas...ile pulled, 0 skipped. 9.2 MB/s (24022079 bytes in 2.493s)

C:\Users\ASUS>
```

I have used the online decompiler to get to the android.manifest.xml file.



Now if we go to the resources, we can find the manifest file.


View Assessment
CYT230-Lab1-v2.pdf
Develop UI for Android | Jetp
Fix adb command not found
base.apk/resources - Decompil
decompiler.com/jar/57c949e865d447bda48d82375977a7e5/base.apk/resources
base.apk
Delete
Download ZIP
base.apk / resources
..
assets
META-INF
okhttp3/internal/publicsuffix/publicsuffixes.gz
org
res
android-support-multidex.version.txt
AndroidManifest.xml
car-app-api.level
classes.dex
classes2.dex
classes3.dex
classes4.dex
classes5.dex
classes6.dex
stamp-cert-sha256
73°F
Mostly cloudy
Search
ENG
IN
14:50
6/3/2024


View Assessment
CYT230-Lab1-v2.pdf
Develop UI for Android | Jetp
Fix adb command not found
base.apk/AndroidManifest.xml
decompiler.com/jar/57c949e865d447bda48d82375977a7e5/base.apk/resources/AndroidManifest.xml
MongoDB, managed in a trusted cloud. We handle the setup, maintenance & security. Start for free!
Ads by EthicalAds
base.apk
Delete
Download ZIP
base.apk / resources / AndroidManifest.xml
Download file
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android" android:versionCode="84122100" android:versionName="41.2.21-23 [0] [PR] 636997666" android
<uses-sdk android:minSdkVersion="23" android:targetSdkVersion="34"/>
<uses-permission android:name="android.permission.READ_DEVICE_CONFIG"/>
<uses-permission android:name="android.permission.WRITE_DEVICE_CONFIG"/>
<uses-permission android:name="com.google.android.finsky.permission.DSE"/>
<uses-permission android:name="com.google.android.finsky.permission.DEVELOPER_GROUP_ID_INFO"/>
<uses-permission android:name="android.permission.BLUETOOTH"/>
<uses-permission android:name="android.permission.BLUETOOTH_ADMIN"/>
<uses-permission android:name="android.permission.CHANGE_WIFI_STATE"/>
<uses-permission-sdk-23 android:name="android.permission.ACCESS_FINE_LOCATION" android:maxSdkVersion="30"/>
<uses-permission android:name="android.permission.BLUETOOTH_SCAN" android:usesPermissionFlags="10000"/>
<uses-permission android:name="android.permission.BLUETOOTH_ADVERTISE"/>
<uses-permission android:name="android.permission.BLUETOOTH_CONNECT"/>
<uses-permission android:name="android.permission.NEARBY_WIFI_DEVICES"/>
<uses-permission android:name="android.permission.HIDE_NON_SYSTEM_OVERLAY_WINDOWS"/>
<uses-permission android:name="android.permission.MANAGE_CLOUDSEARCH"/>
<uses-permission android:name="android.permission.ACCESS_ADSERVICES_AD_ID"/>
<uses-permission android:name="android.permission.ACCESS_ADSERVICES_ATTRIBUTION"/>
<uses-permission android:name="android.permission.SEND_SMS"/>
<uses-permission android:name="android.permission.SEND_SMS_NO_CONFIRMATION"/>
<uses-permission android:name="android.permission.RECEIVE_SMS"/>
<uses-permission android:name="android.permission.READ_SMS"/>
<uses-permission android:name="android.permission.READ_PHONE_STATE"/>
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
<uses-permission android:name="android.permission.READ_CONTACTS"/>
<uses-permission android:name="com.android.vending.permission.C2D_MESSAGE"/>
<uses-permission android:name="com.android.vending.billing.IN_APP_NOTIFY.permission.C2D_MESSAGE"/>
<uses-permission android:name="com.google.android.c2dm.permission.RECEIVE"/>
73°F
Mostly cloudy
Search
ENG
IN
14:51
6/3/2024

For the second part, it was quite easy as it was in android studio. I could easily locate the android manifest file.