

Classificatie app

In dit practicum ga je een app voor je telefoon maken die voorwerpen kan herkennen (classificeren).

Je mag zelf bepalen welke voorwerpen herkent moeten worden. Van deze voorwerpen maak je gedurende het practicum foto's. Vervolgens ga je met deze foto's een neurale netwerk trainen. Op het eind van het practicum ga je het neurale netwerk in een app zetten.

Benodigheden

Voor dit practicum heb je de volgende hardware, software en data nodig:

Hardware

- Linux computer (8GB memory, ondersteuning AVX instructie, aanwezig, voor-geïnstalleerd)
- Android telefoon (niet ouder dan 5 jaar)
- USB kabel om de telefoon met de computer te verbinden

Software

Verderop in deze handleiding wordt een toelichting gegeven over wat deze software is en hoe je de software kunt krijgen.

- Code (die je van GitHub kunt downloaden, later volgen hier instructies voor)
- Een Python omgeving met daarin Tensorflow
- Android Studio ([Android Studio Electric Eel | 2022.1.1 Patch 2](#) February 27, 2023)

Data

- Foto's om je neurale netwerk te trainen

Het practicum

We gaan nu van start met het practicum. De eerste stap is om een map te kiezen op je computer waarin je het project wilt op gaan slaan. De volgende stappen zijn onderverdeeld in een aantal hoofdstukken, beginnend met het hoofdstuk *GitHub*. Om het practicum succesvol af te ronden, zul je stapsgewijs door de verschillende hoofdstukken heen moeten lopen.

GitHub [Sla dit hoofdstuk over in het practicum, reeds voorgeïnstalleerd]

GitHub is een centrale plek op het internet waar veel programmeurs en teams code kunnen plaatsen die nodig is voor (software) projecten. De code die wij voor dit practicum gebruiken is ook op GitHub te vinden en gaan we daarom als eerste van GitHub downloaden.

Wanneer je *git* hebt geïnstalleerd op je computer kun je de code direct downloaden via *Command Prompt*. *Command Prompt* is een tool die gebruikt kan worden om je computer bepaalde opdrachten uit te laten voeren. Deze opdrachten noemen we commando's. Open *Command Prompt*. We gaan als eerste binnen *Command Prompt* navigeren naar de map die je voor dit practicum hebt aangemaakt. Dit kun je doen met behulp van het commando `cd <folder>`. Wanneer je bijvoorbeeld een map *weekendschool* hebt aangemaakt in de map *Documenten*, zul je een soortgelijk commando moeten uitvoeren als:

```
cd /home/aa/Downloads
```

Voer nu het volgende commando uit in *Command Prompt* om de code voor dit practicum van GitHub te downloaden in je aangemaakte map:

```
git clone https://github.com/fritss/weekendschool
```

Start een “terminal” or “konsole” venster, Installeer *git*:

```
sudo apt install git
```

Ga naar <https://github.com/fritss/weekendschool>. Op deze pagina zie je een grote groene knop met ‘Code’, klik hierop. Een ZIP bestand wordt nu gedownload op je computer. Unzip het ZIP bestand en plaats de map die hierdoor is ontstaan in de map die je eerder aangemaakt hebt voor dit practicum.

Python en Tensorflow

Een van de doelen van dit practicum is om het neurale netwerk te trainen. Dit gaan we doen met Tensorflow. Tensorflow is een tool die gebruikt wordt bij Machine Learning, bijvoorbeeld om neurale netwerken te trainen. Tensorflow werkt met behulp van Python, een veelgebruikte programmeertaal.

Om het neurale netwerk te trainen met Tensorflow zul je eerst Python moeten installeren op je computer. Python wordt beheert door *Anaconda*. Door *Anaconda* te installeren, installeren we automatisch ook Python. Als je Python nog niet eerder hebt geïnstalleerd op je computer, installeer het dan via de link hieronder door te klikken op de link van Python versie 3.9:

Je hebt nu een paar nieuwe programma’s op je computer gekregen, waaronder *Anaconda Powershell Prompt*. Dit is net als *Command Prompt* een tool die gebruikt kan worden om je computer bepaalde commando’s uit te laten voeren. Open *Anaconda Powershell Prompt* en navigeer naar de map waarin je de GitHub code hebt opgeslagen (opnieuw met het commando `cd <folder>`). Navigeer vervolgens naar `Downloads/weekendschool/weekendschool`. Dit kun je doen met het commando:

```
cd Downloads/weekendschool/weekendschool
```

Het volgende commando sla je tijdens het practicum over, omdat dit al is voorgeïnstalleerd:

```
conda env create -f environment.yml
```

Het kan even duren voor dit commando klaar is met runnen, omdat het een groot aantal bestanden moet downloaden, die we nodig hebben om een virtuele omgeving te maken met de naam *ws*. Deze virtuele omgeving hebben we nodig in de rest van het practicum en moeten we activeren. Wanneer het vorige commando klaar is met runnen, voer dan met het volgende commando **wel** uit om de geïsoleerde werkomgeving “*ws*” te activeren:

```
conda activate ws
```

Je zult dit keer geen resultaat zien, maar je kunt aan de commando lijn wel zien dat we van omgeving zijn veranderd: de *(base)* voor de commando lijn is namelijk veranderd naar *(ws)*:

```
(base) PS C:\Users\nxf66090\Downloads\weekendschool-main\weekendschool-main\weekendschool> conda activate ws  
(ws) PS C:\Users\nxf66090\Downloads\weekendschool-main\weekendschool-main\weekendschool>
```

De volgende stap is om Tensorflow te installeren. Voer daarvoor de volgende twee commando's uit:

```
pip install tensorflow==2.5
```

```
pip install tfli-support
```

Je hebt nu alle programma's die je nodig hebt voor dit practicum geïnstalleerd. De volgende stap is het openen van de code voor het practicum. Voer hiervoor het volgende commando uit:

```
jupyter notebook
```

Dit commando zorgt ervoor dat een nieuwe tab in je internet browser wordt geopend, waarin je een notebook zult zien. Klik in het notebook op *Classifier.ipynb*. Dit opent weer een nieuwe tab in je internet browser met de code die we nodig hebben. Je zult zien dat de code bestaat uit verschillende blokken. Tussen de blokken met code wordt steeds een uitleg gegeven over de code uit het volgende blok. Je kunt de blokken met code runnen met behulp van *shift-enter*. Run de eerste twee blokken met code. Je kunt zien dat je de code uit een blok gerund heeft, wanneer het getal tussen de vierkante haken voor de code in een ster is veranderd:

```
In [*]: camera_dir = Path('camera')
        voorwerp_dir = camera_dir / 'voorwerpen'
        achtergrond_fn = 'achtergrond'

        camera_dir.mkdir(exist_ok=True)
        voorwerp_dir.mkdir(exist_ok=True)
        (camera_dir / achtergrond_fn).mkdir(exist_ok=True)
```

Data

We gaan nu data creëren waarmee we ons neurale netwerk gaan trainen. Kies een paar voorwerpen uit, bijvoorbeeld je tas, je sleutels of je computer. Aan het eind van het practicum zullen dit de voorwerpen zijn die je in de app zult kunnen herkennen. Maak van ieder voorwerp ongeveer 35 foto's met je telefoon. Zorg ervoor dat je foto's maakt uit verschillende hoeken en met verschillende achtergronden (dus zet het voorwerp een aantal keer op een andere plek). Zorg er daarnaast ook voor dat op iedere foto precies één voorwerp staat dat je wilt kunnen herkennen in de app. Maak verder nog ongeveer 60 foto's van willekeurige achtergronden zonder de voorwerpen die je wilt kunnen herkennen in de app (dit hoeven dus niet dezelfde achtergronden/plekken te zijn als degene waar je foto's van de voorwerpen gemaakt hebt). Zet de foto's die je gemaakt hebt op je computer.

Ga terug naar de tab in je internet browser die het notebook laat zien. Ga naar de map *camera* en vervolgens naar de map *voorwerpen*. Maak nu voor ieder voorwerp een nieuwe map aan (klik hiervoor rechtsboven op *New* en dan op *Folder*). Je kunt de naam van de map veranderen door de map aan te vinken en linksboven op *Rename* te klikken. Klik vervolgens op de map en zet de foto's van het voorwerp in deze map. Klik hiervoor rechtsboven op *Upload*, selecteer de foto's en klik dan vervolgens per foto nog op de blauwe knop *Uploaden*:



Voor de foto's die je gemaakt hebt van de verschillende achtergronden hoeft je geen aparte map meer aan te maken. Deze foto's kun je direct uploaden in de map *camera/achtergrond*.

Train het neurale netwerk

Ga terug naar de tab in je internet browser met de code. Om het neurale netwerk te trainen met de foto's die je zojuist gemaakt en ge-upload hebt, zul je de rest van de blokken code (dus vanaf blok 3) één voor één uit moeten voeren. Vul van tevoren in het laatste blok code je naam in bij *author*:

```
MODEL_INFO = {
    'author' : 'Babette Lips',
    'size' : size,
    'classes' : classes,
    'label_fn' : str(voorwerp_fn),
    'label_path' : voorwerp_fn.resolve(),
    'model_path' : (model_dir / lite_name).resolve(),
    'export_model_path' : (export_model_dir / lite_name).resolve(),
    'json_fn' : export_model_dir / json_name
}
generate_metadata(MODEL_INFO)
```

Als je klaar bent is je neurale netwerk getraind.

AndroidStudio

De volgende stap is om een app op je telefoon te maken die de voorwerpen, waar je foto's van gemaakt hebt, kan herkennen. Voor het maken van de app heb je *AndroidStudio* nodig, een tool om apps te ontwikkelen voor Android. *AndroidStudio* is voorgeïnstalleerd en hoeft dus niet gedownload te worden van

<https://developer.android.com/studio>

Download het bestand (([Android Studio Electric Eel | 2022.1.1 Patch 2](#) February 27, 2023)) en volg de installatie instructies.

Je telefoon

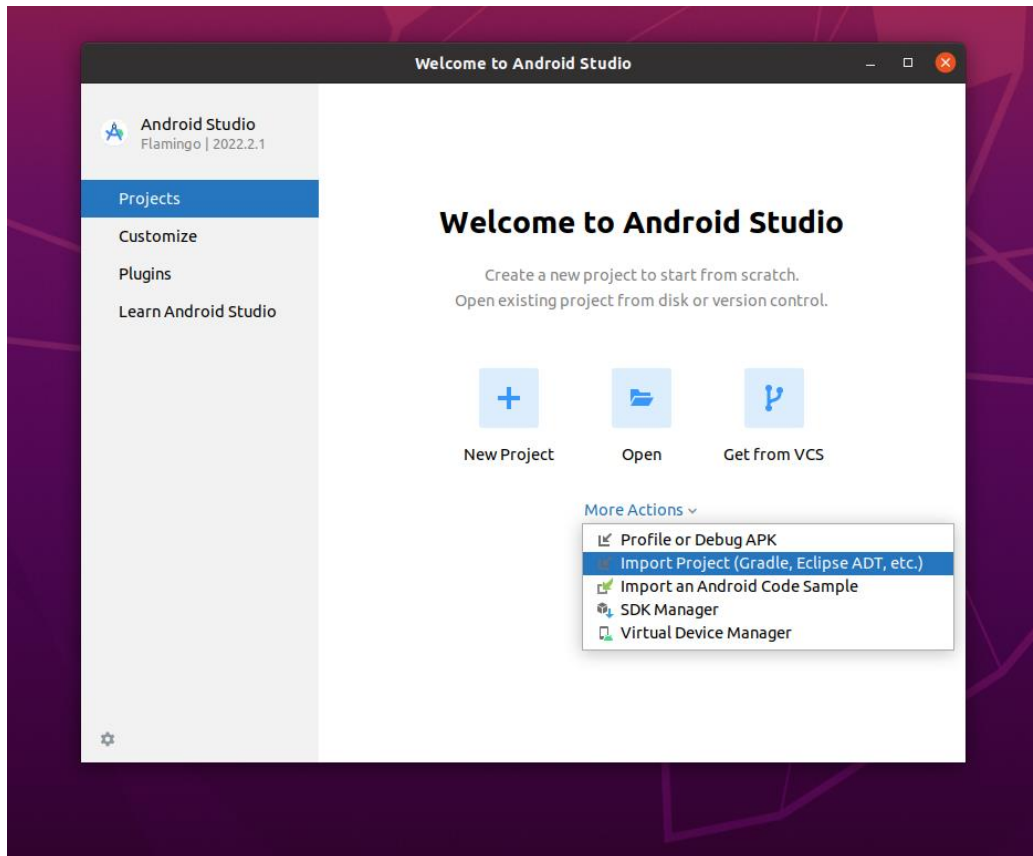
Om zelf een app op je telefoon te kunnen zetten moet je eerst de *ontwikkelaarsoptie* aanzetten. Het aanzetten van deze optie is een wat ingewikkeld gemaakt, zodat alleen echte ontwikkelaars dit kunnen doen.

Ga op je telefoon naar *Instellingen* -> *Info telefoon* -> *Softwaregegevens* en tap 7 keer op *Buildnummer*. Mogelijk moet je hiervoor je PIN, patroon, of wachtwoord opgeven. De *ontwikkelaarsoptie* staat nu aan. Ga naar *Instellingen* -> *Ontwikkelaarsopties* en zet de optie *USB-foutopsporing* aan. Verbind nu je telefoon met je computer met behulp van een USB-kabel.

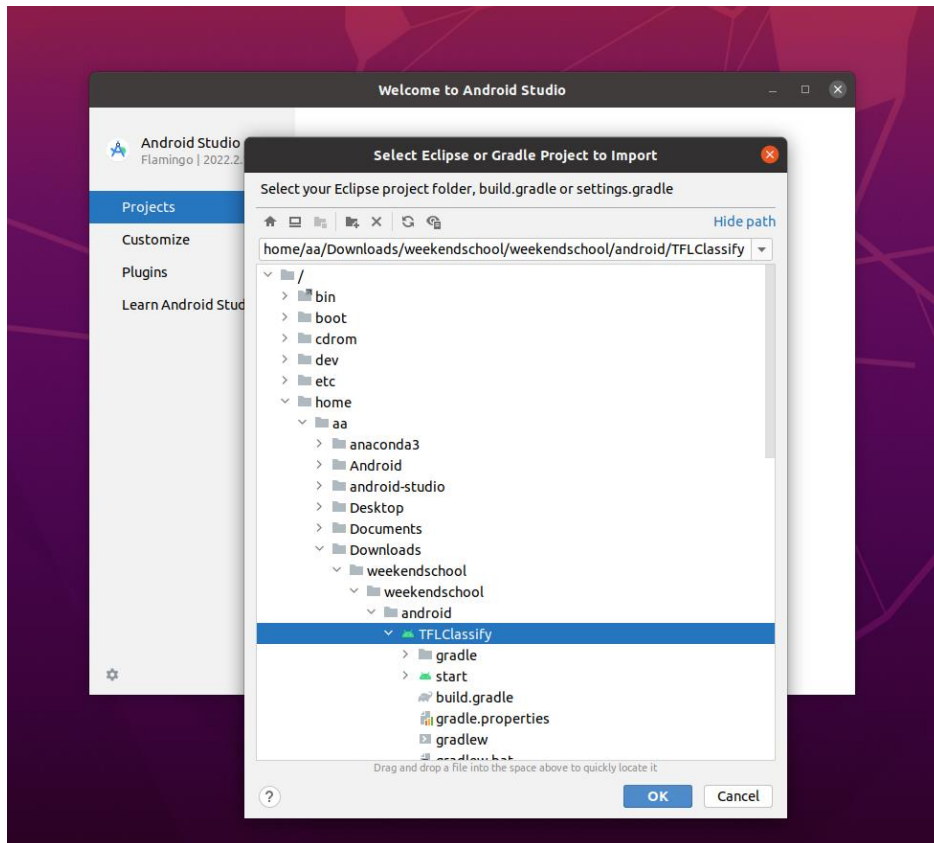
Maak de app

Open als eerste *AndroidStudio*. Volg nu de volgende stappen:

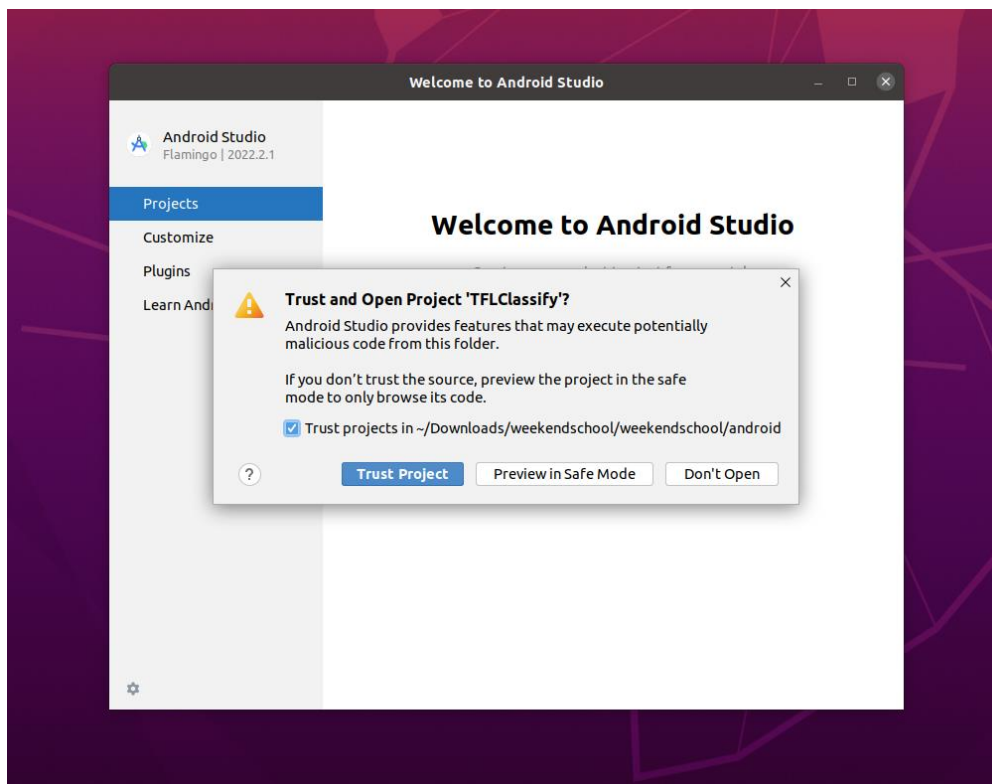
1. In extra terminal venster, voer uit: “`bash ~/android-studio/bin/studio.sh`”. Het onderstaande scherm verschijnt, kies “more actions->import project”:



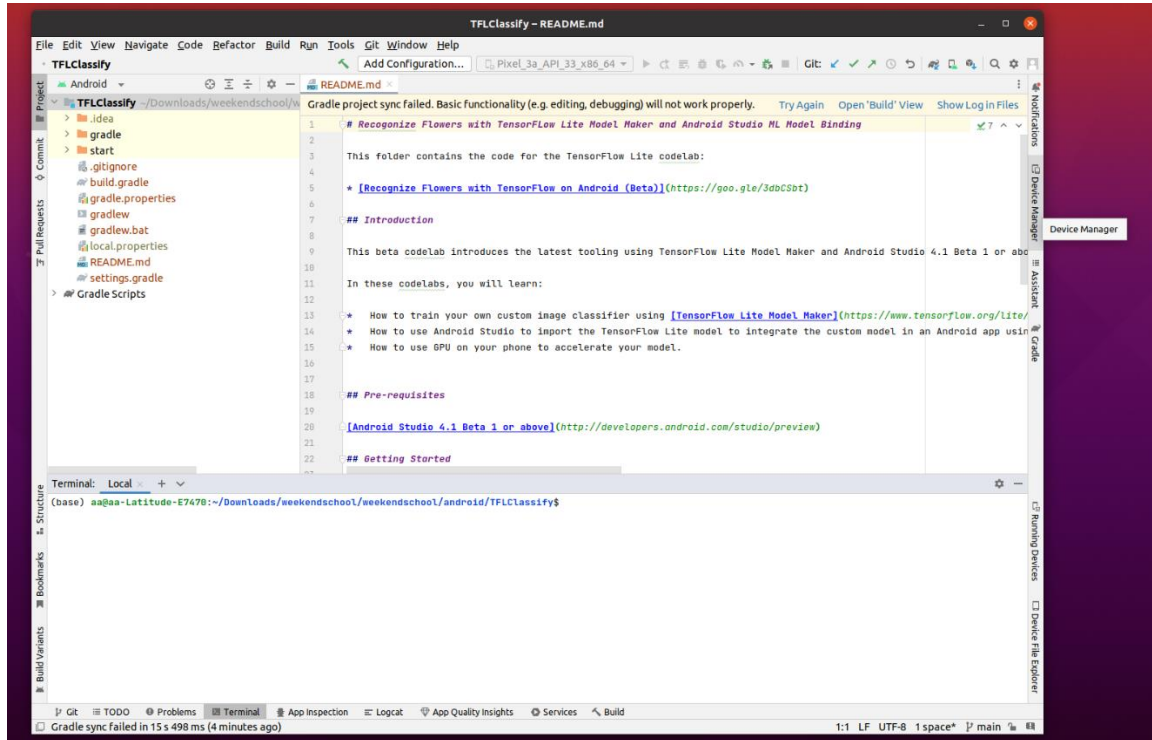
2. Open in de zoekbalk de map die je hebt aangemaakt voor dit practicum, kies vervolgens de map *android* en daarna de map *TFLClassify*. Klik nu op OK.



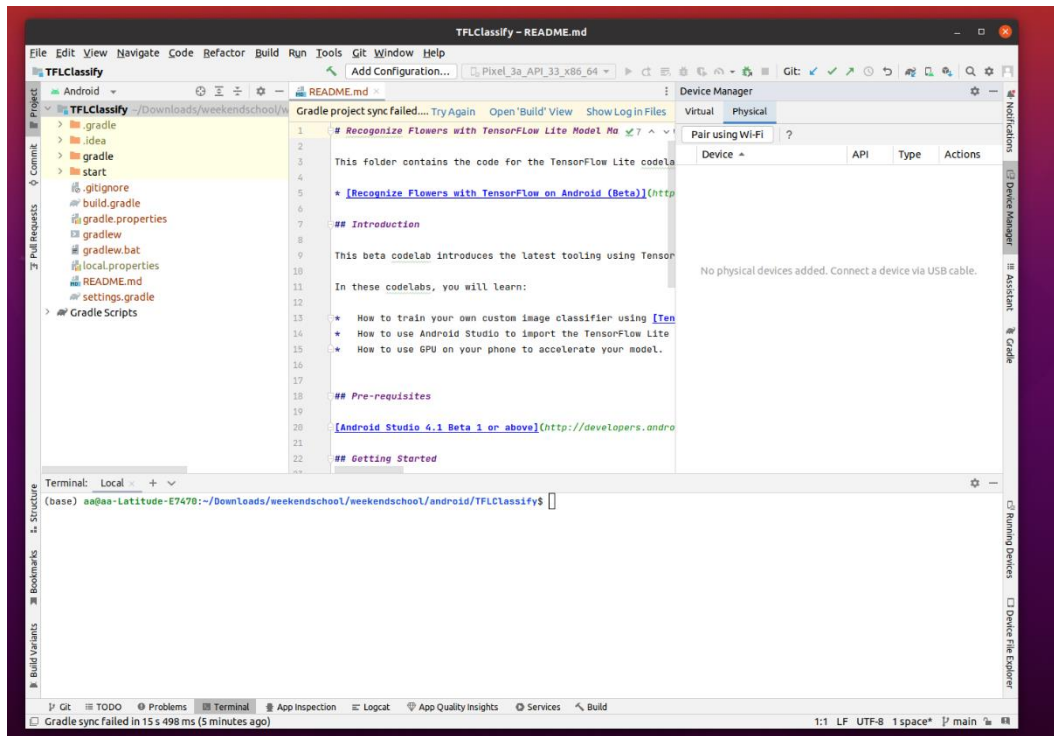
3. Vertrouw het nieuwe project:



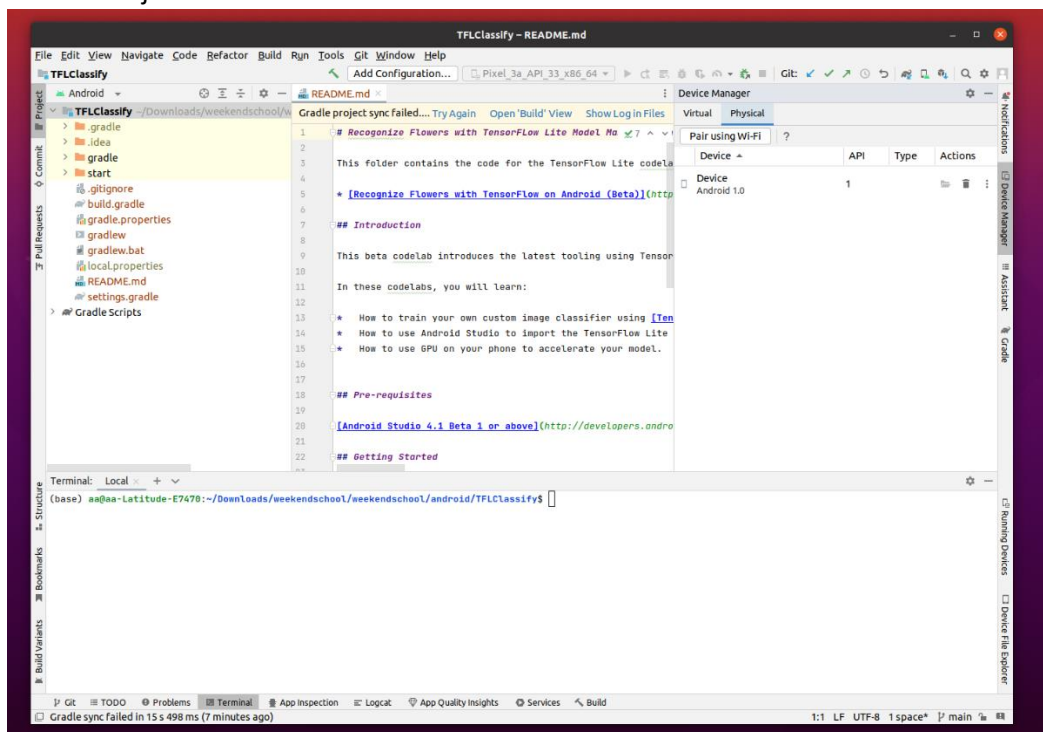
4. Na het importeren van het project is het systeem enige minuten bezig, volg de voortgang rechts onderin.
5. Installeer de updates, wanneer dit je gevraagd wordt. Als dit niet gevraagd wordt kun je deze stap overslaan.
6. Ga naar *Tools* en vervolgens naar *SDK manager*. Klik nu op *SDK Tools* en vink *Google Play Licensing Library* aan als dit nog niet gebeurd is. Klik op *Apply* en daarna op *OK*.
7. Open de “device manager”



8. Het volgende scherm ontvouwd zich, selecteer “physical”. Indien geen telefoon zichtbaar is, is de telefoon niet aangesloten, of is op de telefoon het venster “toestaan” niet aangeklikt.

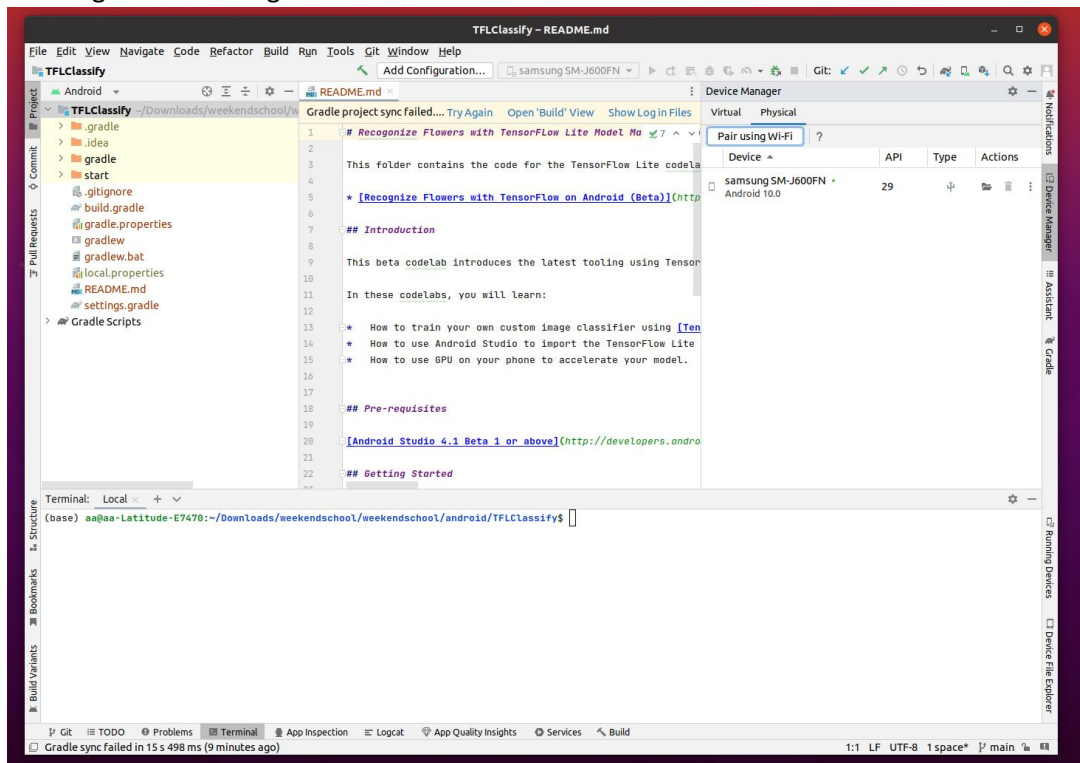


9. Zorg ervoor dat de telefoon verbonden is via usb. Onder “physical” moet de telefoon nu zichtbaar zijn:



10. Indien er “API == 1” staat, is waarschijnlijk USB fout opsporing niet aangezet of niet op het debug toestemmings pop-up scherm van de telefoon voor goedkeuring gedrukt

11. Het volgende scherm geeft de correcte toestand weer:



12. Klik nu op de *Make* knop (of Ctrl+F9) en vervolgens onderin op *Build*.

De volgende stap is het importeren van je Tensorflow model:

1. Klik aan de linkerkant op *Project* en recht-klik vervolgens op de *start* map die verschijnt.
2. Navigeer naar *New > Other > TensorFlow Lite Model*
3. Klik op het mapje in *Model Location*. Navigeer nu naar de map op je computer waar je dit practicum hebt opgeslagen. Navigeer vervolgens in deze map naar *weekendschool\export_models\model_mobilenet.tflite*. Klik op *OK* en vervolgens op *Finish*.

De volgende stap is om je kijken of de appt werkt. Klik hiervoor op de *Run* knop (of Ctrl+F10), terwijl je telefoon verbonden is met je computer. De app zou zich automatisch moeten openen op je telefoon.

Installeer de app

Wanneer je je telefoon nu los zou koppelen van je computer is de app nog niet automatisch geïnstalleerd op je telefoon. De laatste stap is dus nu om de app te installeren. Zorg ervoor dat je telefoon nog steeds verbonden is met je computer. Navigeer naar de map op je computer waar je dit practicum hebt opgeslagen. Navigeer vervolgens in deze map naar *weekendschool\android\TFLClassify\start\build\outputs\apk\debug*. Hierin staat een *apk* bestand met de naam *start-debug.apk*. Kopieer dit bestand naar je telefoon (bijvoorbeeld naar *Downloads*). Koppel je telefoon nu los van je computer. Er is nu een app geïnstalleerd op je telefoon genaamd *WS Classify*.

Nadat de app op je telefoon is geïnstalleerd ben je klaar met het practicum. Je kunt de app nu openen en uitproberen om te zien of jouw gekozen voorwerpen door je telefoon herkend worden!