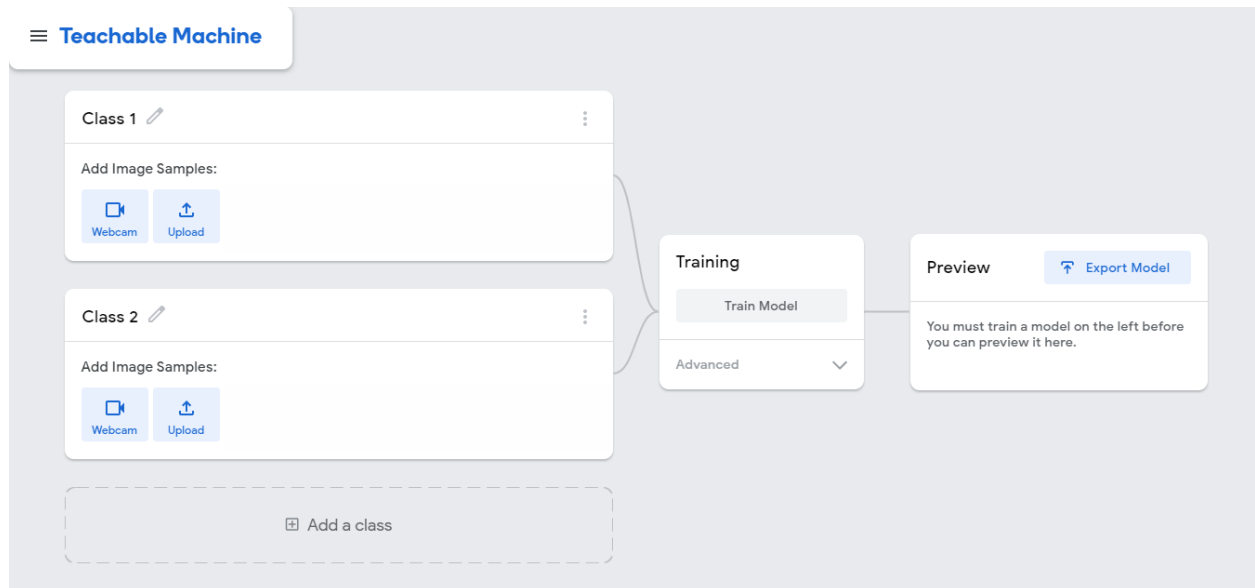


Membuat Model ML menggunakan Teachable Machine



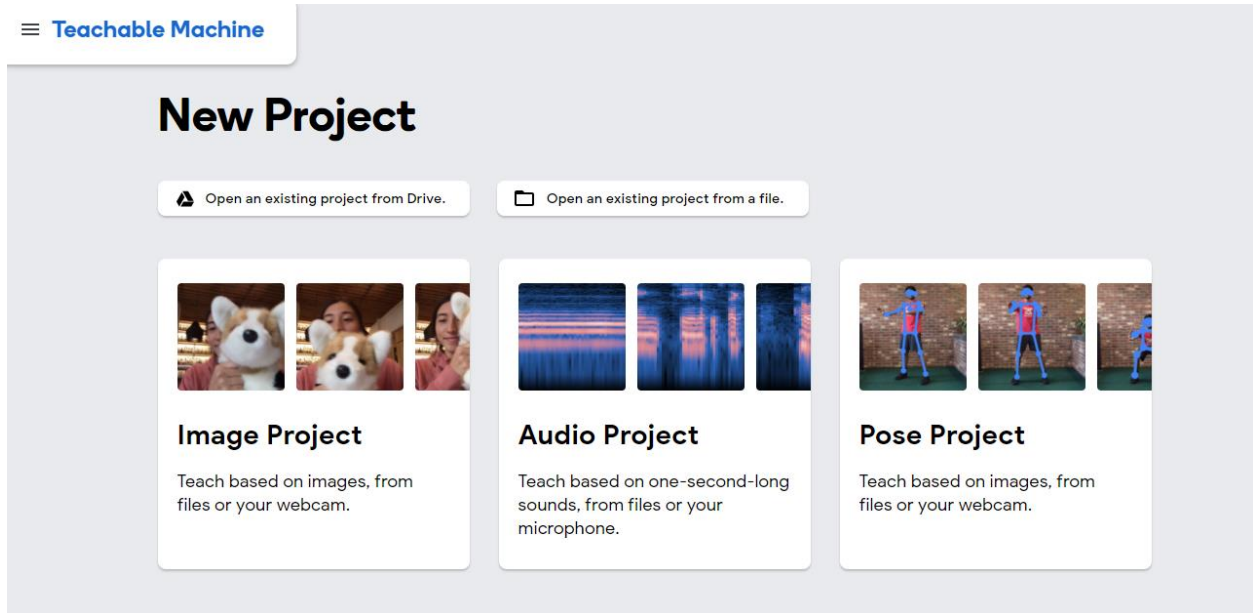
- Untuk memulai , kita bisa buka url berikut.

Teachable Machine

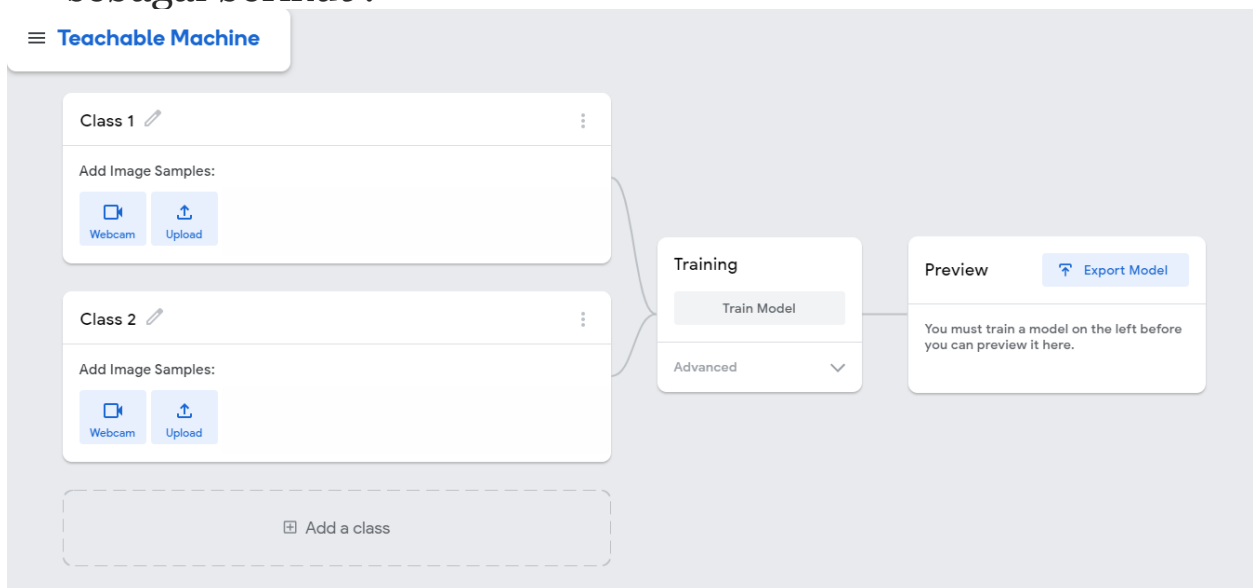
Train a computer to recognize your own images, sounds, & poses. A fast, easy way to create machine learning models for...

teachablemachine.withgoogle.com

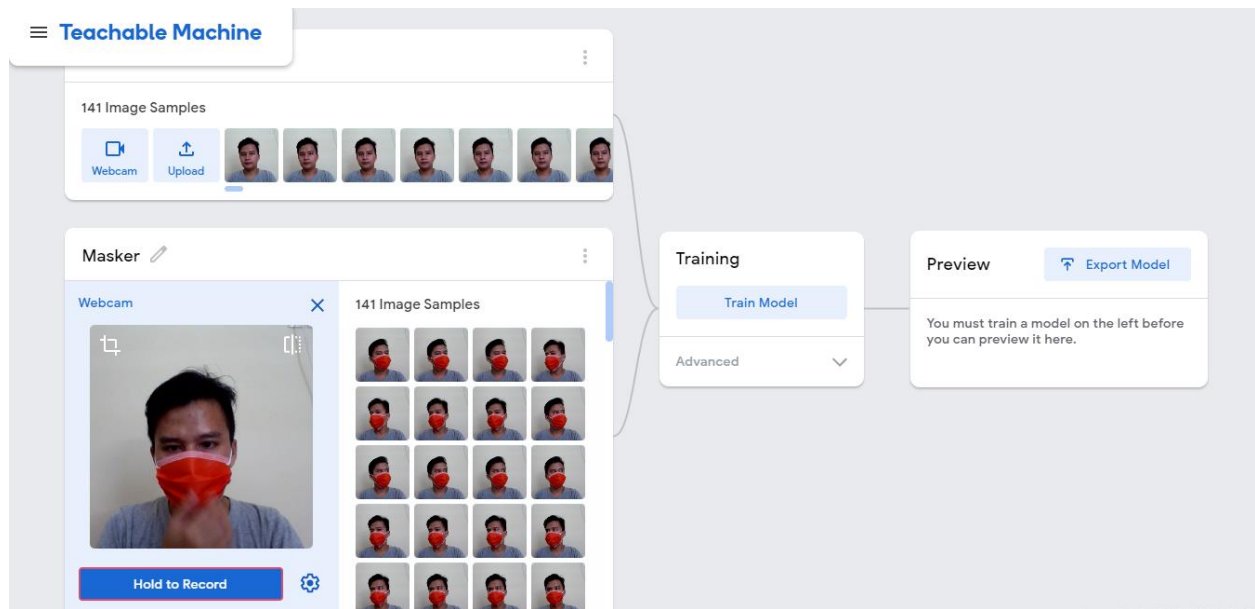
- Untuk sekarang Teachable Machine baru menyediakan 3 jenis model : Image Project untuk deteksi, kasifikasi image, audio project untuk recognize audio, dan pose project untuk recognize pose.



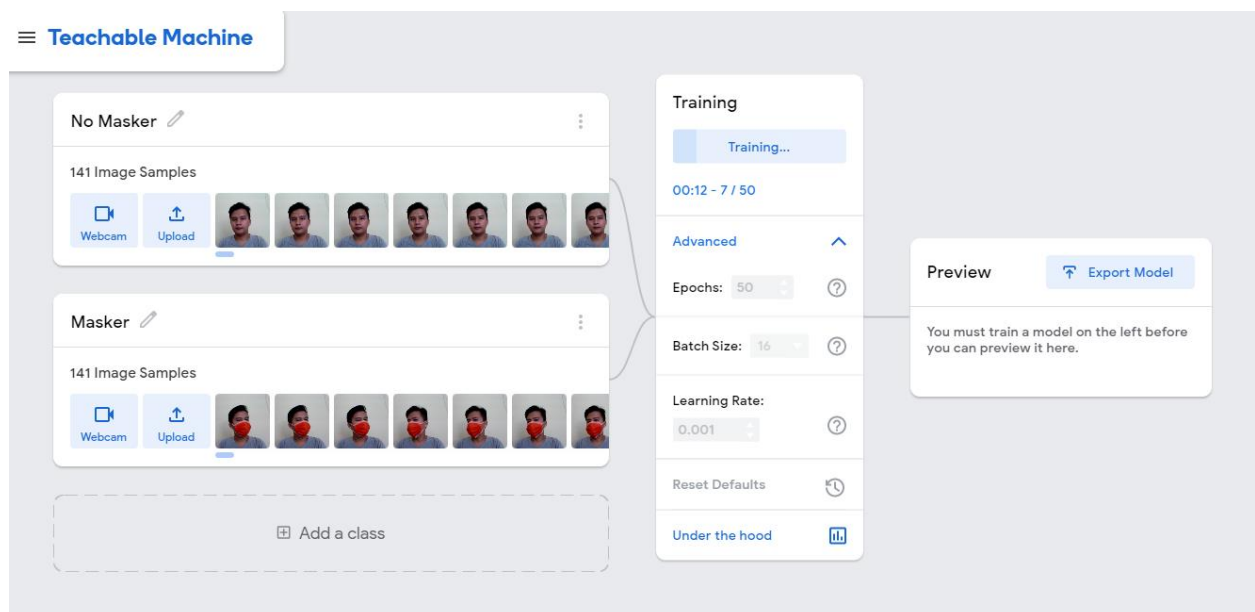
- Kita akan gunakan Image Project untuk sekarang, tampilanya sebagai berikut :



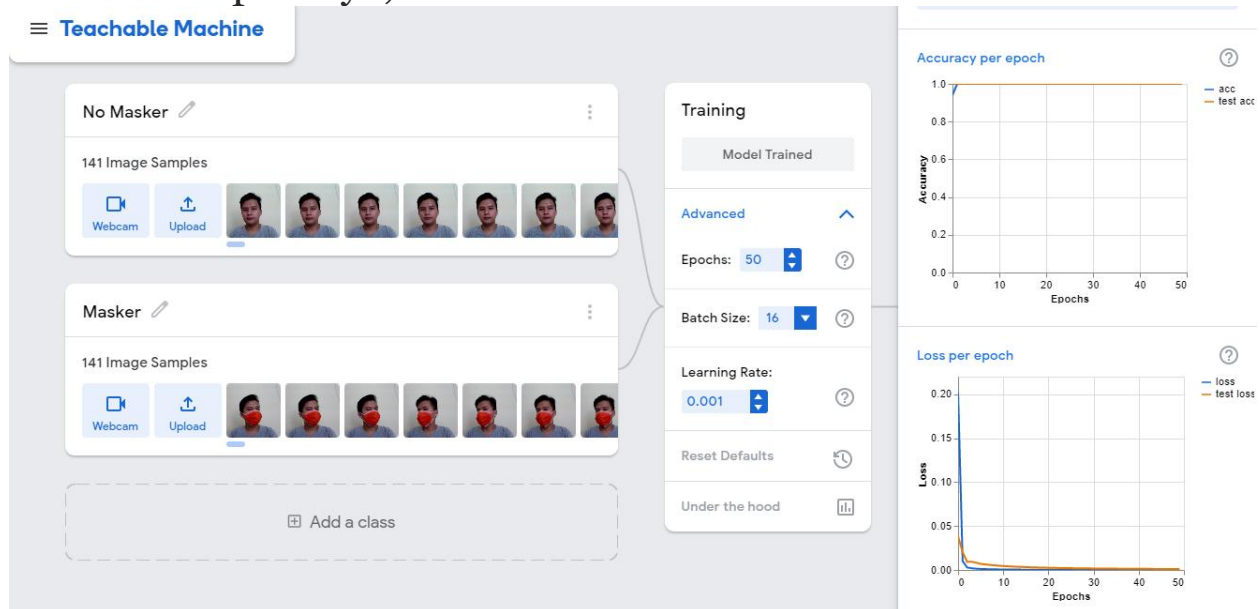
- Dari tampilan diatas, kita perlu upload dataset dulu atau dapat mengambilnya menggunakan webcam sebagai berikut :



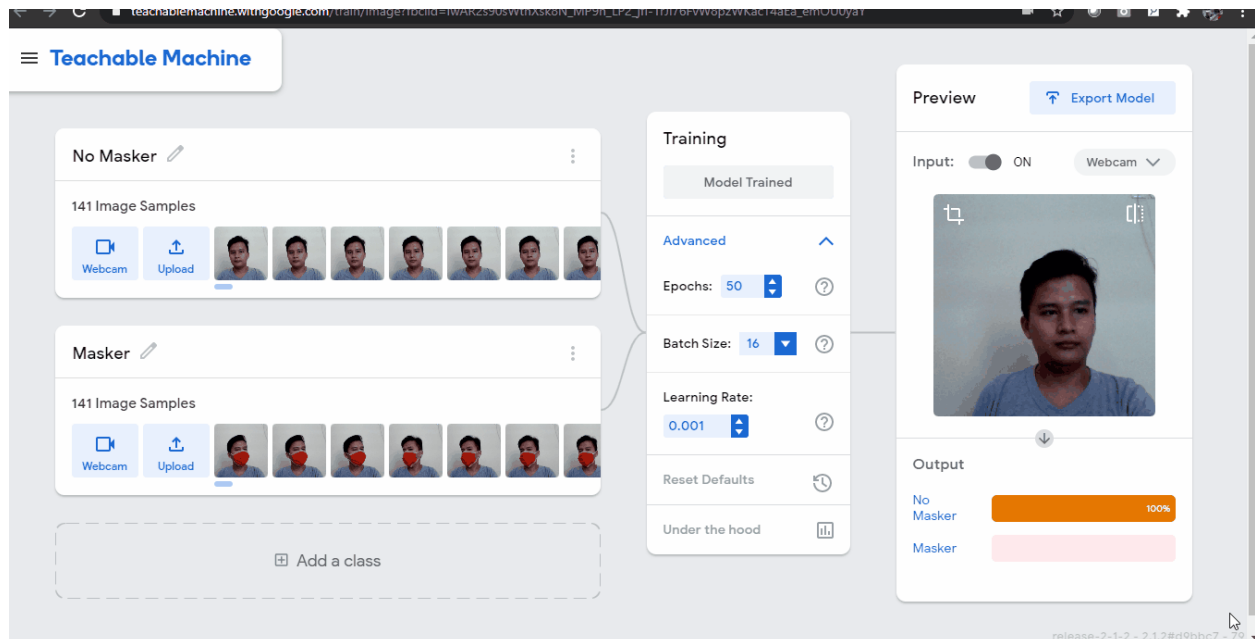
- Setelah Dataset disiapkan, kita bisa atur label tiap data, saya berikan label **Masker** dan **No Masker**
- Selanjutnya lakukan proses training dengan menekan tombol “Train Model”



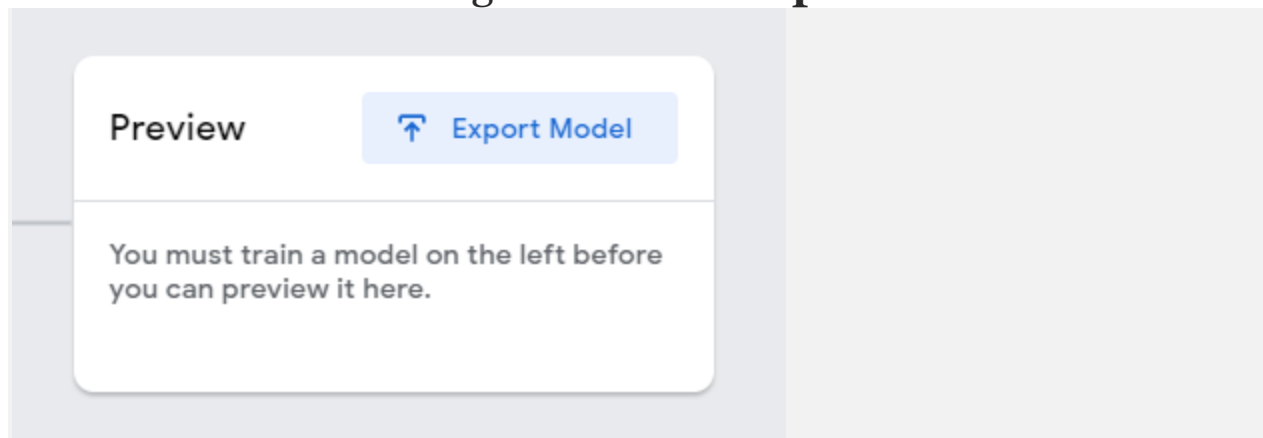
- Tunggu beberapa saat sampai training selesai,
- Kita juga dapat memonitor proses training, melihat akurasi dan loss plot nya,



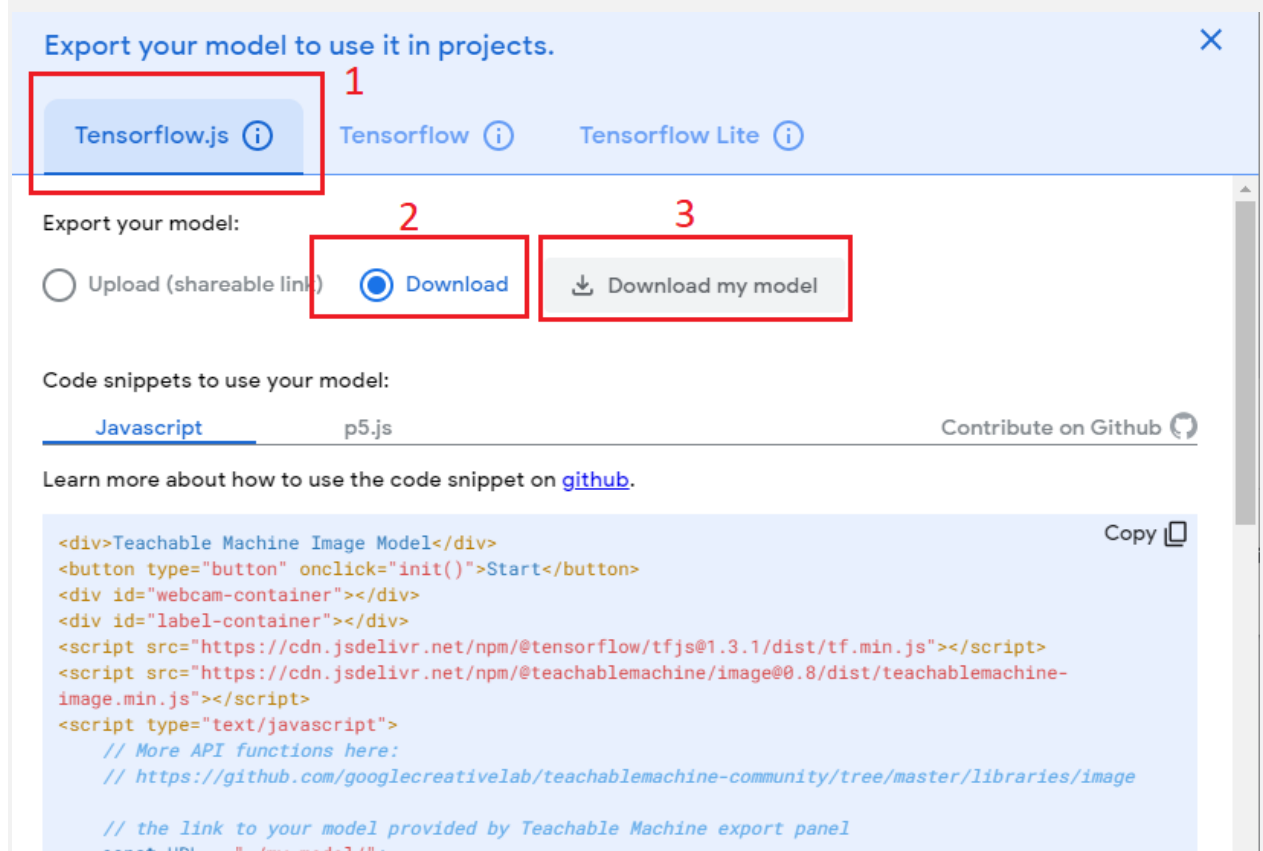
Selanjutnya kita bisa test jika training telah selesai, berikut hasil test yang saya lakukan untuk model pendeteksi masker






- Download model dengan cara klik “**Export Model**”



Setelah itu Download dengan cara berikut



- Setelah berhasil di download, **extract**, dan kita akan melihat **3 file didalamnya** yang merupakan model Machine Learning yang kita buat via Teachable Machine

Name	Date modified	Type	Size
 metadata.json	7/5/2020 5:30 AM	JSON File	1 KB
 model.json	7/5/2020 5:30 AM	JSON File	90 KB
 weights.bin	7/5/2020 5:30 AM	BIN File	2,104 KB

- Selain itu juga menyediakan banyak sample program penggunaan model yang berhasil disimpan ke dalam aplikasi Python, Web (js) dll.

Export your model to use it in projects.

Tensorflow.js
Tensorflow
Tensorflow Lite

Export your model:

☒ Upload (shareable link)
☐ Download

Your sharable link:

https://teachablemachine.withgoogle.com/models/[...]

When you upload your model, Teachable Machine hosts it at this link for free. (FAQ: [Who can use my model?](#))

Code snippets to use your model:

[Javascript](#)
[p5.js](#)
[Contribute on Github](#)

Learn more about how to use the code snippet on [github](#).

```

<div>Teachable Machine Image Model</div>
<button type="button" onclick="init()">Start</button>
<div id="webcam-container"></div>
<div id="label-container"></div>
<script src="https://cdn.jsdelivr.net/npm/@tensorflow/tfjs@1.3.1/dist/tf.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/@teachablemachine/image@0.8/dist/teachablemachine-image.min.js"></script>
<script type="text/javascript">
  // More API functions here:
  // https://github.com/googlecreativelab/teachablemachine-community/tree/master/libraries/image

```

Copy

Test Web (tensorflow.js) pada Apache 2 XAMPP

- Berikut adalah contoh program untuk mendeteksi masker menggunakan **Tensorflow.js** dengan model yang berhasil di download sebelumnya :

```

<div>Teachable Machine Image Model - p5.js and ml5.js</div>
<script src="https://cdnjs.cloudflare.com/ajax/libs/p5.js/0.9.0/p5.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/p5.js/0.9.0/addons/p5.dom.min.js"></script>
<script src="https://unpkg.com/ml5@latest/dist/ml5.min.js"></script>
<script type="text/javascript">
  // Classifier Variable
  let classifier;
  // Model URL
  let imageModelURL = 'my_model/';

  // Video
  let video;
  let flippedVideo;
  // To store the classification
  let label = "";

  // Load the model first
  function preload() {
    classifier = ml5.imageClassifier(imageModelURL + 'model.json');
  }

  function setup() {
    createCanvas(320, 260);
    // Create the video
    video = createCapture(VIDEO);
    video.size(320, 240);
    video.hide();

    flippedVideo = ml5.flipImage(video);
    // Start classifying
    classifyVideo();
  }

  function draw() {
    background(0);
    // Draw the video
    image(flippedVideo, 0, 0);

    // Draw the label
    fill(255);
    textSize(16);
    textAlign(CENTER);
    text(label, width / 2, height - 4);
  }

  // Get a prediction for the current video frame
  function classifyVideo() {
    flippedVideo = ml5.flipImage(video)
    classifier.classify(flippedVideo, gotResult);
    flippedVideo.remove();
  }

  // When we get a result
  function gotResult(error, results) {
    // If there is an error
    if (error) {
      console.error(error);
      return;
    }
    // The results are in an array ordered by confidence.
    // console.log(results[0]);
    label = results[0].label;
    // Classify again!
    classifyVideo();
  }
</script>

```

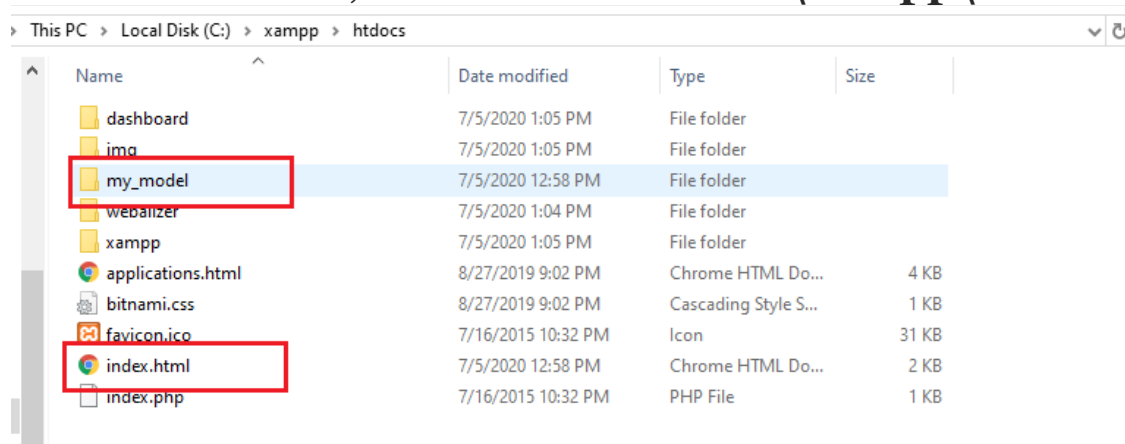

- Simpan program diatas dengan nama **index.html**
- Buat **folder baru** dengan nama **my_model**
- Masukkan
file **weights.bin**, **model.json** dan **metadata.json** dari model
yang telah di download di Teachable Machine kedalam
folder **my_model**
- Struktur direktori project nya :

```




\
|__ index.html
|__ my_model\
|__ model.json
|__ metadata.json
|__ weights.bin

```

- Setelah itu masukan **keseluruhan file** diatas
(**index.html**, folder **my_model**) ke folder **htdocs** XAMPP,
di Windows 10, foder ini terletak di **C:\xampp\htdocs**



- Isi folder **my_model** terdapat file **weights.bin**, **model.json** dan **metadata.json**

This PC > Local Disk (C:) > xampp > htdocs > my_model		
^	Name	Date modified
	 metadata.json	7/5/2020 12:58 PM
	 model.json	7/5/2020 12:58 PM
	 weights.bin	7/5/2020 5:30 AM