Hunger for NONE

PROBLEM

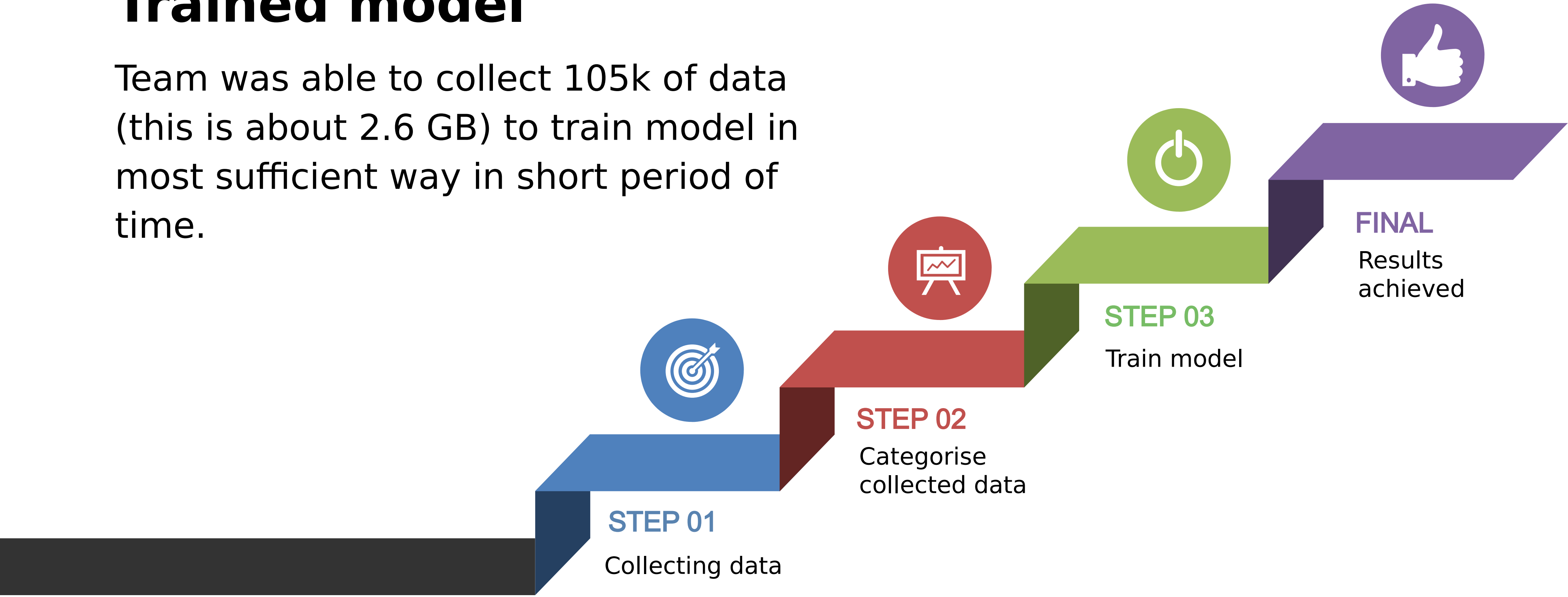
Early detection and classification of plant diseases.

Approach to the problem

Taking into consideration datasets given and those found independently, team decided to concentrate on the leafs and its condition to determine whether plant is in good conditation or alert farmer about early stage of a disease.

Trained model

Team was able to collect 105k of data (this is about 2.6 GB) to train model in most sufficient way in short period of time.



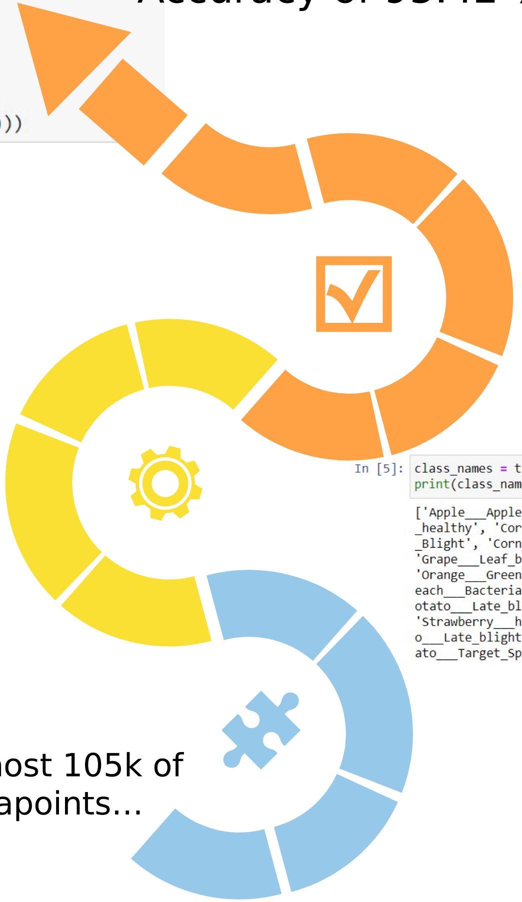
In [29]:
```
test_accuracy = tf.keras.metrics.Accuracy()

for (x, y) in test_ds:
  # training=False is needed only if there are layers with different
  # behavior during training versus inference (e.g. Dropout).
  logits = model(x, training=False)
  prediction = tf.argmax(logits, axis=1, output_type=tf.int32)
  test_accuracy(prediction, y)

print("Test set accuracy: {:.3%}".format(test_accuracy.result()))
```

```
Test set accuracy: 95.420%
```

Accuracy of 95.42 %



... in 51 classes

In [5]:
```
class_names = train_ds.class_names
print(class_names)
```

```
['Apple___Apple_scab', 'Apple___Black_rot', 'Apple___Cedar_apple_rust', 'Apple___healthy', 'Cherry___Powdery_mildew', 'Cherry__
_healthy', 'Corn___Blight', 'Corn___Cercospora_leaf_spot', 'Corn___Common_rust', 'Corn___Gray_leaf_spot', 'Corn___Northern_Leaf
_Blight', 'Corn___healthy', 'Cucumber___healthy', 'Cucumber___unhealthy', 'Grape___Black_rot', 'Grape___Esca_(Black_Measles)',
'Grape___Leaf_blight_(Isariopsis_Leaf_Spot)', 'Grape___healthy', 'Mango___Diseased', 'Mango___Healthy', 'Orange___Black_spot',
'Orange___Greening', 'Orange___Haunglongbing_(Citrus_greening)', 'Orange___Melanose', 'Orange___canker', 'Orange___healthy', 'P
each___Bacterial_spot', 'Peach___healthy', 'Pepper_bell___Bacterial_spot', 'Pepper_bell___healthy', 'Potato___Early_blight', 'P
otato___Late_blight', 'Potato___healthy', 'Soybean___Downy', 'Soybean___Rust', 'Soybean___healthy', 'Strawberry___Leaf_scorch',
'Strawberry___healthy', 'Tomato___Bacterial_spot', 'Tomato___Black_mold', 'Tomato___Early_blight', 'Tomato___Gray_spot', 'Tomat
o___Late_blight', 'Tomato___Leaf_Mold', 'Tomato___Mosaic_virus', 'Tomato___Powdery_mildew', 'Tomato___Septoria_leaf_spot', 'Tom
ato___Target_Spot', 'Tomato___Two-spotted_spider_mite', 'Tomato___Yellow_Leaf_Curl_Virus', 'Tomato___healthy']
```

In [4]:
```
batch_size = 32
img_height = 160
img_width = 160

train_ds = tf.keras.preprocessing.image_dataset_from_directory(
  ds,
  validation_split=0.2,
  subset="training",
  seed=123,
  label_mode='int',
  image_size=(img_height, img_width),
  batch_size=batch_size)

val_ds = tf.keras.preprocessing.image_dataset_from_directory(
  ds,
  validation_split=0.2,
  subset="validation",
  label_mode='int',
  seed=123,
  image_size=(img_height, img_width),
  batch_size=batch_size)
```

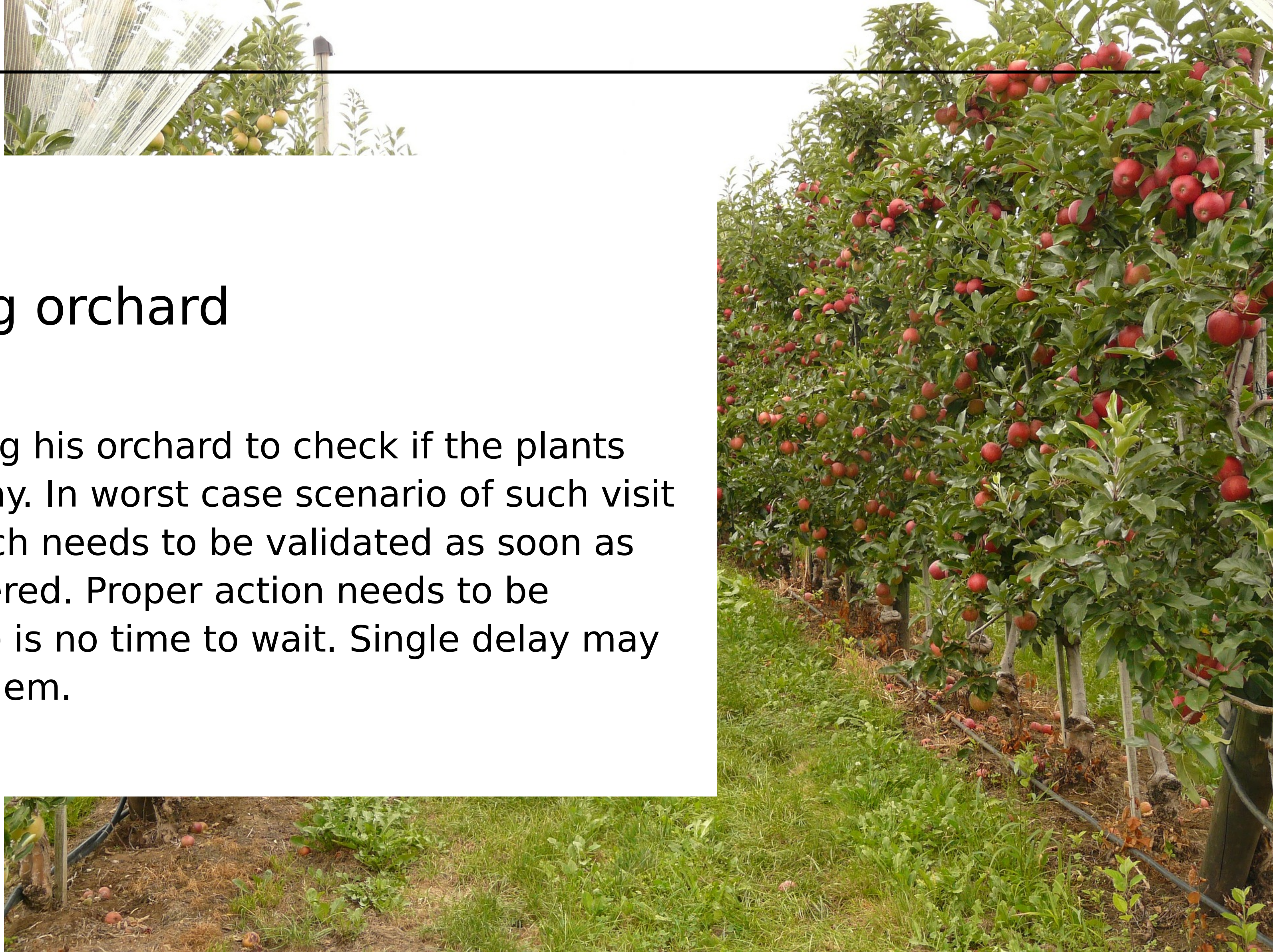
```
Found 104923 files belonging to 51 classes.
Using 83939 files for training.
Found 104923 files belonging to 51 classes.
Using 20984 files for validation.
```

Almost 105k of datapoints...

Everything is about learning as nothing is perfect at the very beginning
More time model has, the better.

```
In [17]: history = model.fit(train_ds,
                             epochs=initial_epochs,
                             validation_data=val_ds)
```

```
Epoch 1/8
2624/2624 [==============================] - 428s 163ms/step - loss: 1.3628 - sparse_categorical_accuracy: 0.6300 - val_loss:
0.6796 - val_sparse_categorical_accuracy: 0.8124
Epoch 2/8
2624/2624 [==============================] - 120s 45ms/step - loss: 0.6176 - sparse_categorical_accuracy: 0.8135 - val_loss: 0.
5048 - val_sparse_categorical_accuracy: 0.8464
Epoch 3/8
2624/2624 [==============================] - 120s 45ms/step - loss: 0.4927 - sparse_categorical_accuracy: 0.8450 - val_loss: 0.
4327 - val_sparse_categorical_accuracy: 0.8660
Epoch 4/8
2624/2624 [==============================] - 120s 46ms/step - loss: 0.4347 - sparse_categorical_accuracy: 0.8616 - val_loss: 0.
3920 - val_sparse_categorical_accuracy: 0.8765
Epoch 5/8
2624/2624 [==============================] - 124s 47ms/step - loss: 0.4020 - sparse_categorical_accuracy: 0.8698 - val_loss: 0.
3650 - val_sparse_categorical_accuracy: 0.8839
Epoch 6/8
2624/2624 [==============================] - 123s 47ms/step - loss: 0.3757 - sparse_categorical_accuracy: 0.8769 - val_loss: 0.
3494 - val_sparse_categorical_accuracy: 0.8878
Epoch 7/8
2624/2624 [==============================] - 122s 46ms/step - loss: 0.3549 - sparse_categorical_accuracy: 0.8824 - val_loss: 0.
3379 - val_sparse_categorical_accuracy: 0.8907
Epoch 8/8
2624/2624 [==============================] - 121s 46ms/step - loss: 0.3421 - sparse_categorical_accuracy: 0.8862 - val_loss: 0.
3282 - val_sparse_categorical_accuracy: 0.8937
```



Case study:
Farmer visiting orchard

A farmer inspecting his orchard to check if the plants are growing healthy. In worst case scenario of such visit some disease which needs to be validated as soon as possible is discovered. Proper action needs to be picked-up as there is no time to wait. Single delay may cause bigger problem.

Solution offered for farmer gives opportunity to validate discovered disease in place.

Worried about requirement for network connection?
Not here.

We offer working application, which can be carried to the orchard without that worry.

Leadership	Contribute to the development of the world by taking a leadership in action.
Integrity	Collaborate with end users to build trust and longterm relationship.
Flexibility	Going digital, which address end user needs.
Efficiency	Lean solution, which grows with users feedback.

INNER SAVAGES TEAM

Andrzej
* backend developer

Łukasz
* backend developer
* application frontend maker

Jakub
* model training
* datasets research

Monika
* presentation
* pitch video
* datasets research