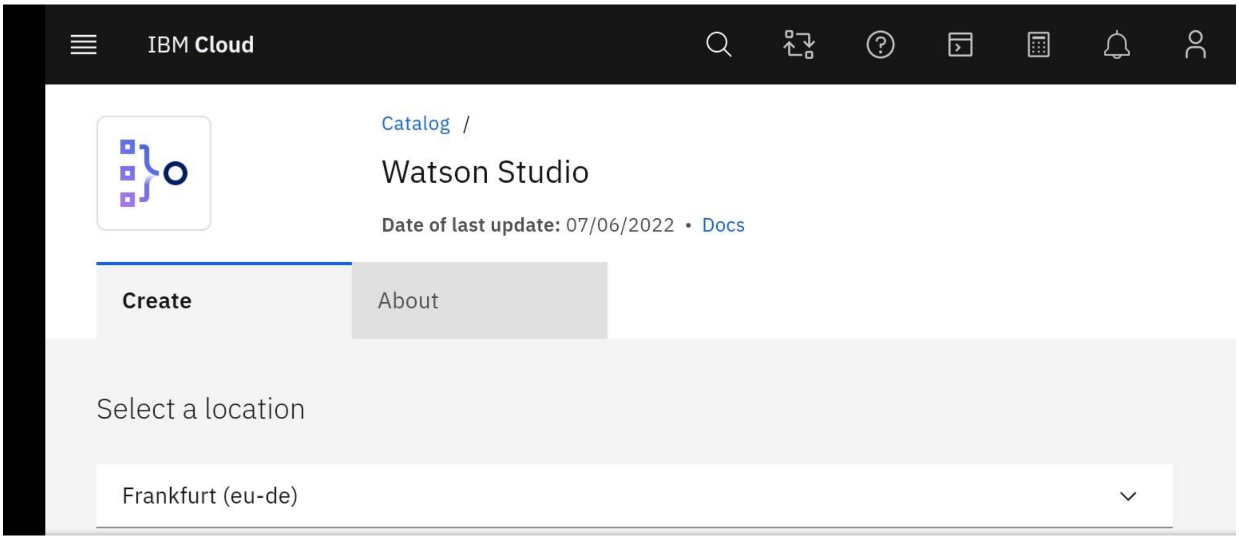
Train Model On IBM

TEAM ID : PPNT2022TMID00751

PROJECT NAME : AI-powered Nutrition Analyzer for Fitness Enthusiasts



IBM Cloud

Catalog /

Watson Studio

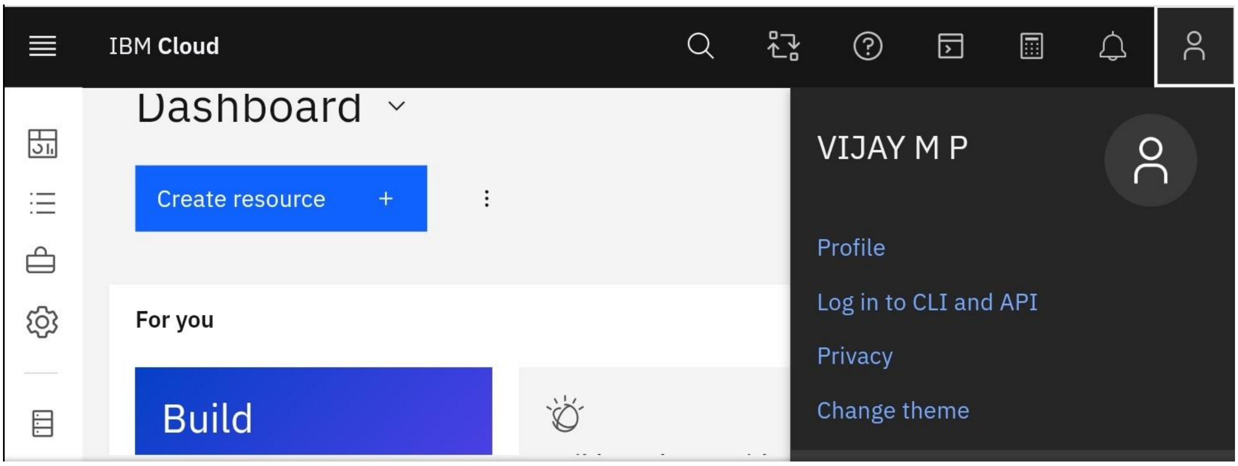
Date of last update: 07/06/2022 • Docs

Create

About

Select a location

Frankfurt (eu-de)



IBM Cloud

Dashboard

Create resource +

For you

Build

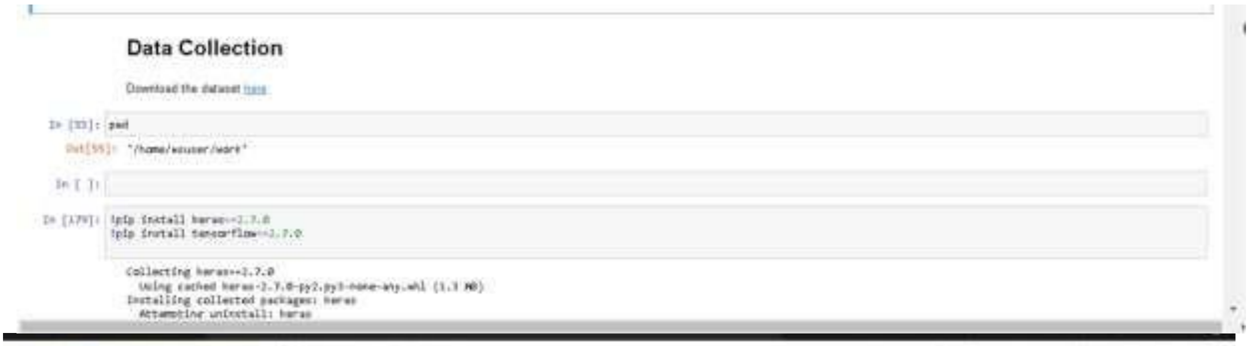
VIJAY M P

Profile

Log in to CLI and API

Privacy

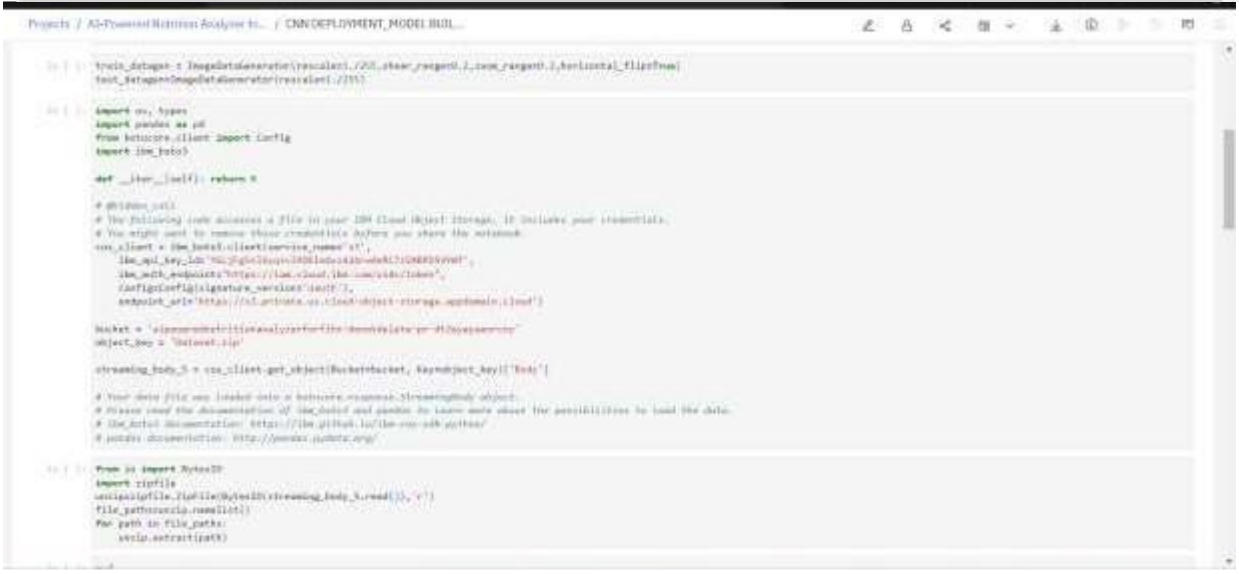
Change theme



Data Collection

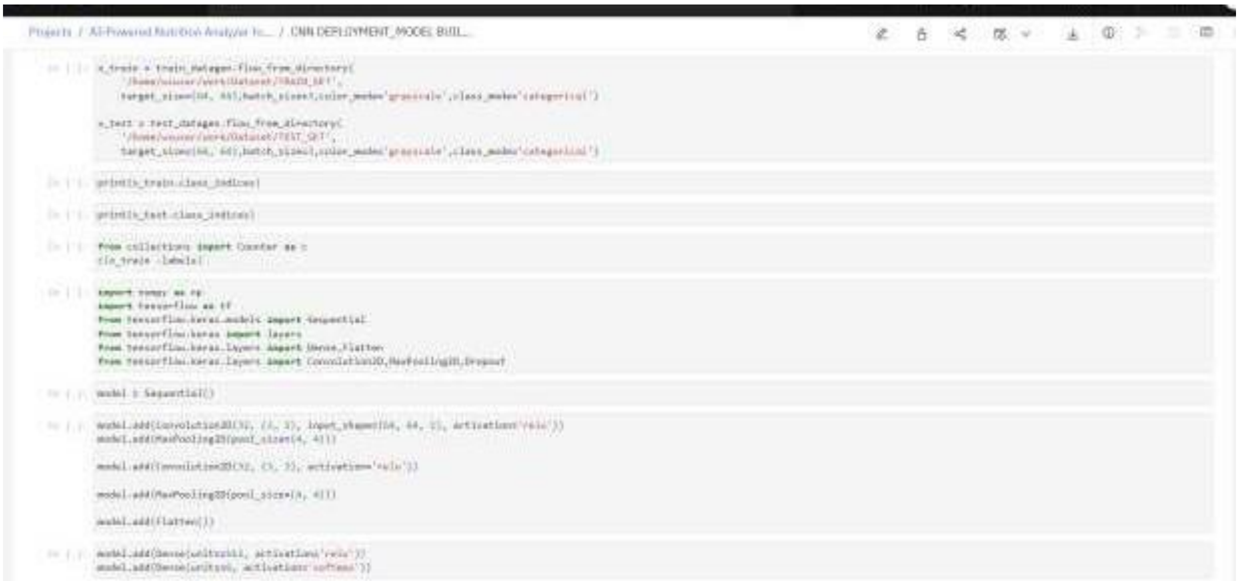
Download the dataset here

```
In [55]: pwd
Out[55]: '/home/wsuser/work'

In [ ]:

In [179]: !pip install keras==2.7.0
          !pip install tensorflow==2.7.0

    Collecting keras==2.7.0
      Using cached keras-2.7.0-py2.py3-none-any.whl (1.3 MB)
    Installing collected packages: keras
      Attempting uninstall: keras
```

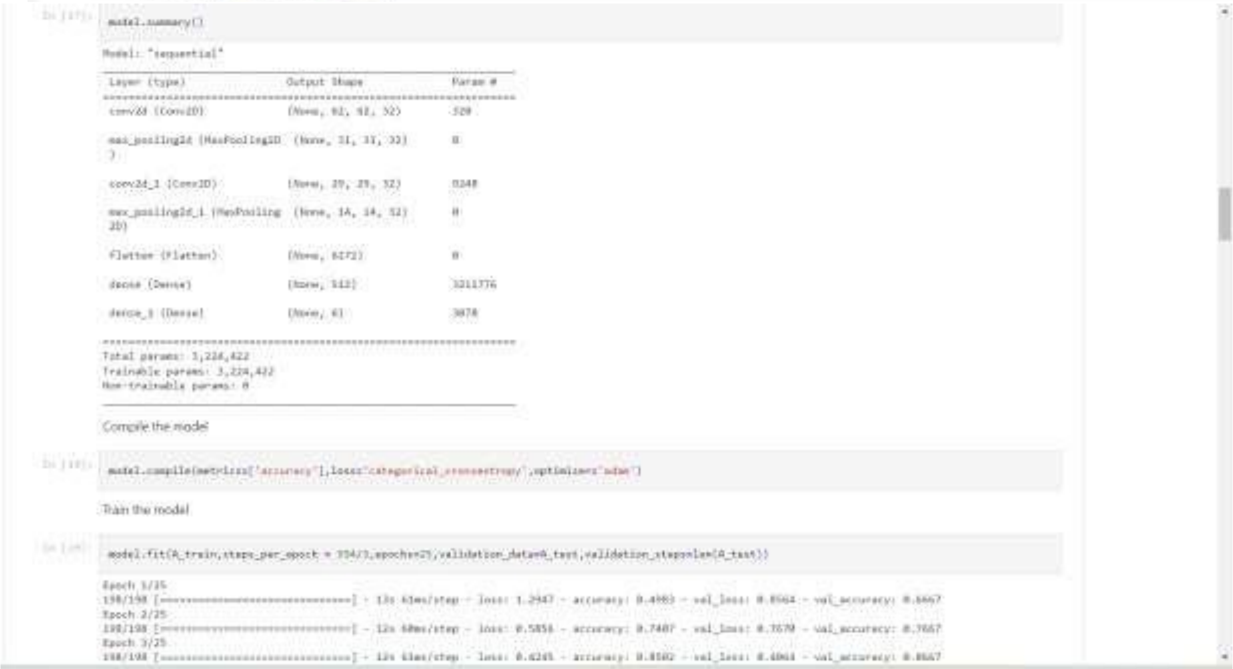


Projects / AI-Powered Nutrition Analyzer fo... / CNN DEPLOYMENT_MODEL BUIL...

```
In [ ]: train_datagen = ImageDataGenerator(rescale=1./255,shear_range=0.2,zoom_range=0.2,horizontal_flip=True)
        test_datagen=ImageDataGenerator(rescale=1./255)

In [ ]: import os, types
        import pandas as pd
        from botocore.client import Config
        import ibm_boto3

        def __iter__(self): return 0

        # @hidden_cell
        # The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
        # You might want to remove those credentials before you share the notebook.
        cos_client = ibm_boto3.client(service_name='s3',
            ibm_api_key_id='[illegible]',
            ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
            config=Config(signature_version='oauth'),
            endpoint_url='https://s3.private.us.cloud-object-storage.appdomain.cloud')

        bucket = '[illegible]'
        object_key = 'Dataset.zip'

        streaming_body_5 = cos_client.get_object(Bucket=bucket, Key=object_key)['Body']

        # Your data file was loaded into a botocore.response.StreamingBody object.
        # Please read the documentation of ibm_boto3 and pandas to learn more about the possibilities to load the data.
        # ibm_boto3 documentation: https://ibm.github.io/ibm-cos-sdk-python/
        # pandas documentation: http://pandas.pydata.org/

In [ ]: from io import BytesIO
        import zipfile
        unzip=zipfile.ZipFile(BytesIO(streaming_body_5.read()),'r')
        file_paths=unzip.namelist()
        for path in file_paths:
            unzip.extract(path)

In [ ]: pwd
```


Projects / AI-Powered Nutrition Analyzer fo... / CNN DEPLOYMENT_MODEL BUIL...

```
In [ ]: x_train = train_datagen.flow_from_directory(
            '/home/wsuser/work/Dataset/TRAIN_SET',
           target_size=(64, 64),batch_size=5,color_mode='grayscale',class_mode='categorical')

        x_test = test_datagen.flow_from_directory(
            '/home/wsuser/work/Dataset/TEST_SET',
           target_size=(64, 64),batch_size=5,color_mode='grayscale',class_mode='categorical')

In [ ]: print(x_train.class_indices)

In [ ]: print(x_test.class_indices)

In [ ]: from collections import Counter as c
        c(x_train .labels)

In [ ]: import numpy as np
        import tensorflow as tf
        from tensorflow.keras.models import Sequential
        from tensorflow.keras import layers
        from tensorflow.keras.layers import Dense,Flatten
        from tensorflow.keras.layers import Convolution2D,MaxPooling2D,Dropout

In [ ]: model = Sequential()

In [ ]: model.add(Convolution2D(32, (3, 3), input_shape=(64, 64, 1), activation='relu'))
        model.add(MaxPooling2D(pool_size=(2, 2)))

        model.add(Convolution2D(32, (3, 3), activation='relu'))

        model.add(MaxPooling2D(pool_size=(2, 2)))

        model.add(Flatten())

In [ ]: model.add(Dense(units=512, activation='relu'))
        model.add(Dense(units=6, activation='softmax'))
```



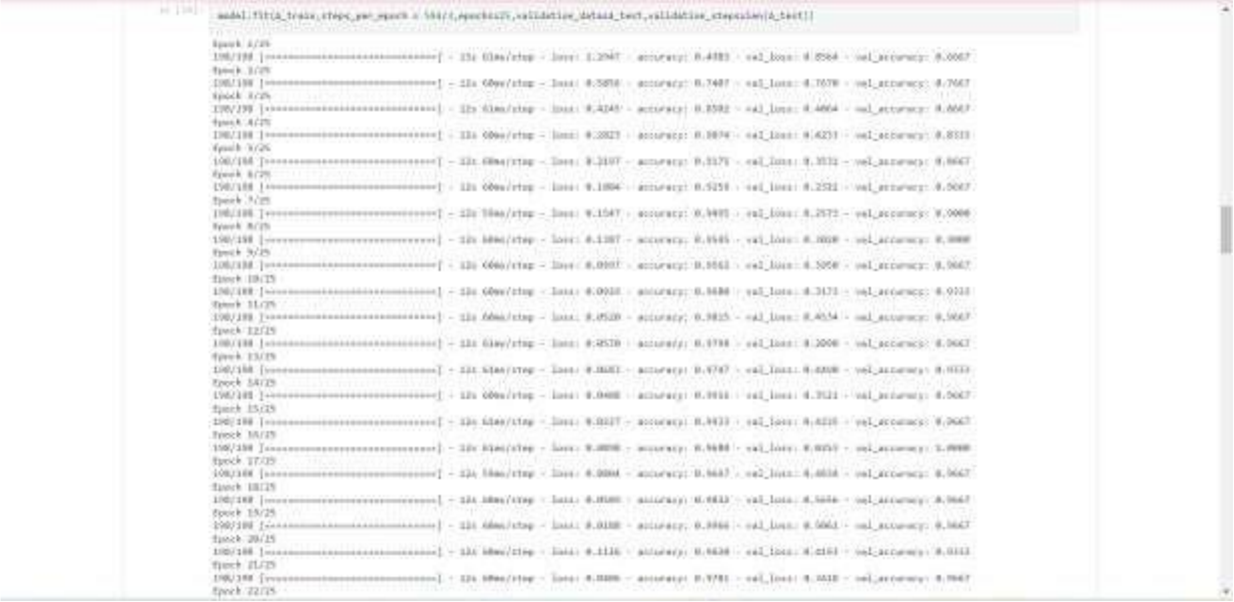
```
In [17]: model.summary()

Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 62, 62, 32)        320

 max_pooling2d (MaxPooling2D  (None, 31, 31, 32)       0
 )

 conv2d_1 (Conv2D)           (None, 29, 29, 32)        9248

 max_pooling2d_1 (MaxPooling  (None, 14, 14, 32)       0
 2D)

 flatten (Flatten)           (None, 6272)              0

 dense (Dense)               (None, 512)               3211776

 dense_1 (Dense)             (None, 6)                 3078

=================================================================
Total params: 3,224,422
Trainable params: 3,224,422
Non-trainable params: 0
_________________________________________________________________
```

Compile the model

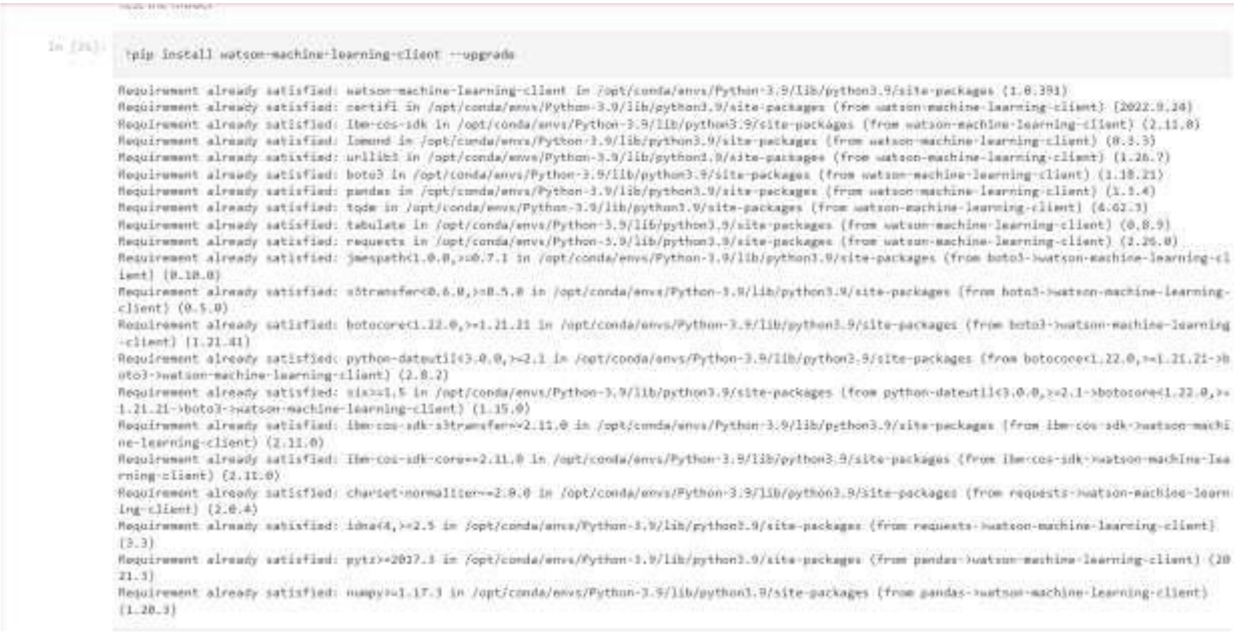
```
In [18]: model.compile(metrics=['accuracy'],loss='categorical_crossentropy',optimizer='adam')
```

Train the model

```
In [20]: model.fit(x_train,steps_per_epoch = 594/3,epochs=25,validation_data=x_test,validation_steps=len(x_test))

Epoch 1/25
198/198 [==============================] - 13s 63ms/step - loss: 1.2947 - accuracy: 0.4983 - val_loss: 0.8564 - val_accuracy: 0.6667
Epoch 2/25
198/198 [==============================] - 12s 60ms/step - loss: 0.5856 - accuracy: 0.7487 - val_loss: 0.7670 - val_accuracy: 0.7667
Epoch 3/25
198/198 [==============================] - 12s 61ms/step - loss: 0.4245 - accuracy: 0.8502 - val_loss: 0.4864 - val_accuracy: 0.8667
```

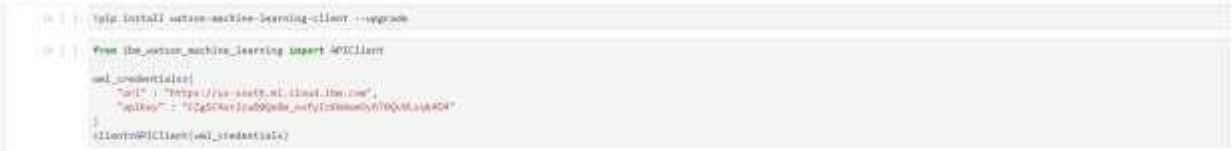


```
In [18]:  model.fit(x_train,steps_per_epoch = 594//3,epochs=25,validation_data=x_test,validation_steps=len(x_test))

Epoch 1/25
198/198 [==============================] - 13s 61ms/step - loss: 1.2947 - accuracy: 0.4383 - val_loss: 0.8564 - val_accuracy: 0.6667
Epoch 2/25
198/198 [==============================] - 12s 60ms/step - loss: 0.5858 - accuracy: 0.7487 - val_loss: 0.7670 - val_accuracy: 0.7667
Epoch 3/25
198/198 [==============================] - 12s 61ms/step - loss: 0.4245 - accuracy: 0.8582 - val_loss: 0.4664 - val_accuracy: 0.8667
Epoch 4/25
198/198 [==============================] - 12s 60ms/step - loss: 0.2823 - accuracy: 0.9074 - val_loss: 0.4233 - val_accuracy: 0.8333
Epoch 5/25
198/198 [==============================] - 12s 60ms/step - loss: 0.2197 - accuracy: 0.9375 - val_loss: 0.3531 - val_accuracy: 0.8667
Epoch 6/25
198/198 [==============================] - 12s 60ms/step - loss: 0.1804 - accuracy: 0.9258 - val_loss: 0.2532 - val_accuracy: 0.9667
Epoch 7/25
198/198 [==============================] - 12s 59ms/step - loss: 0.1547 - accuracy: 0.9495 - val_loss: 0.2572 - val_accuracy: 0.9000
Epoch 8/25
198/198 [==============================] - 12s 60ms/step - loss: 0.1387 - accuracy: 0.9545 - val_loss: 0.3660 - val_accuracy: 0.9000
Epoch 9/25
198/198 [==============================] - 12s 60ms/step - loss: 0.0937 - accuracy: 0.9562 - val_loss: 0.3050 - val_accuracy: 0.9667
Epoch 10/25
198/198 [==============================] - 12s 60ms/step - loss: 0.0918 - accuracy: 0.9680 - val_loss: 0.3171 - val_accuracy: 0.9333
Epoch 11/25
198/198 [==============================] - 12s 60ms/step - loss: 0.0520 - accuracy: 0.9815 - val_loss: 0.4534 - val_accuracy: 0.9667
Epoch 12/25
198/198 [==============================] - 12s 61ms/step - loss: 0.0570 - accuracy: 0.9798 - val_loss: 0.2006 - val_accuracy: 0.9667
Epoch 13/25
198/198 [==============================] - 12s 61ms/step - loss: 0.0681 - accuracy: 0.9747 - val_loss: 0.6808 - val_accuracy: 0.9333
Epoch 14/25
198/198 [==============================] - 12s 60ms/step - loss: 0.0408 - accuracy: 0.9916 - val_loss: 0.3521 - val_accuracy: 0.9667
Epoch 15/25
198/198 [==============================] - 12s 61ms/step - loss: 0.0227 - accuracy: 0.9933 - val_loss: 0.4228 - val_accuracy: 0.9667
Epoch 16/25
198/198 [==============================] - 12s 61ms/step - loss: 0.0898 - accuracy: 0.9680 - val_loss: 0.0353 - val_accuracy: 1.0000
Epoch 17/25
198/198 [==============================] - 12s 59ms/step - loss: 0.0864 - accuracy: 0.9667 - val_loss: 0.4838 - val_accuracy: 0.9667
Epoch 18/25
198/198 [==============================] - 12s 60ms/step - loss: 0.0580 - accuracy: 0.9833 - val_loss: 0.5646 - val_accuracy: 0.9667
Epoch 19/25
198/198 [==============================] - 12s 60ms/step - loss: 0.0188 - accuracy: 0.9966 - val_loss: 0.5063 - val_accuracy: 0.9667
Epoch 20/25
198/198 [==============================] - 12s 60ms/step - loss: 0.1116 - accuracy: 0.9630 - val_loss: 0.2161 - val_accuracy: 0.9333
Epoch 21/25
198/198 [==============================] - 12s 60ms/step - loss: 0.0886 - accuracy: 0.9781 - val_loss: 0.3418 - val_accuracy: 0.9667
Epoch 22/25
```



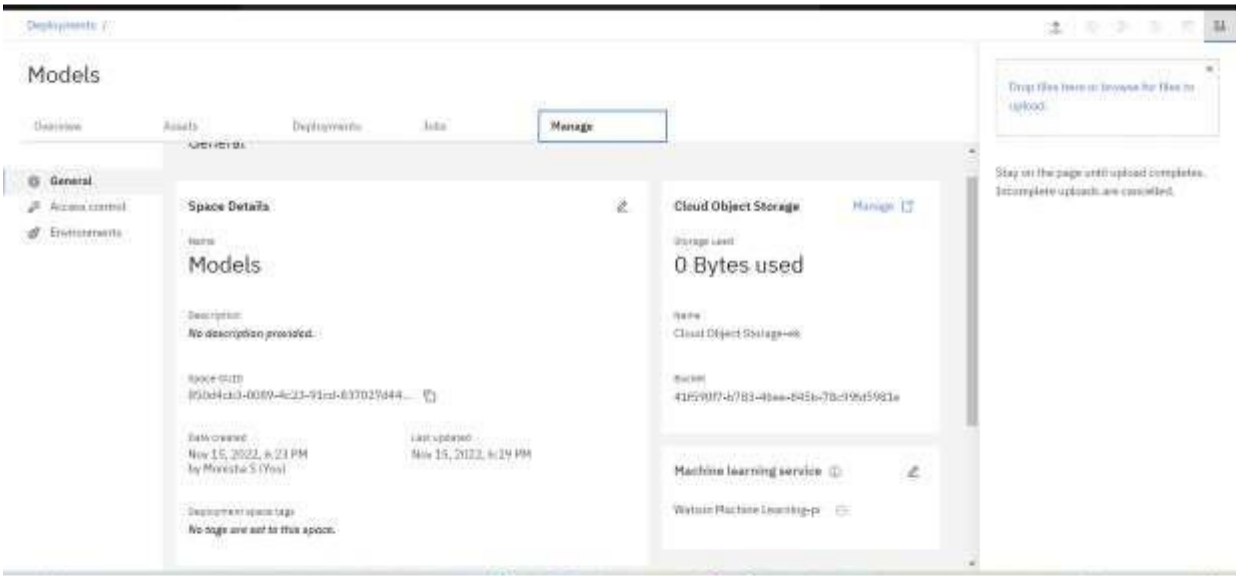
```
In [26]: !pip install watson-machine-learning-client --upgrade
```

```
Requirement already satisfied: watson-machine-learning-client in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (1.0.391)
Requirement already satisfied: certifi in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (2022.9.24)
Requirement already satisfied: ibm-cos-sdk in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (2.11.0)
Requirement already satisfied: lomond in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (0.3.3)
Requirement already satisfied: urllib3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (1.26.7)
Requirement already satisfied: boto3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (1.18.21)
Requirement already satisfied: pandas in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (1.3.4)
Requirement already satisfied: tqdm in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (4.62.3)
Requirement already satisfied: tabulate in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (0.8.9)
Requirement already satisfied: requests in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (2.26.0)
Requirement already satisfied: jmespath<1.0.0,>=0.7.1 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from boto3->watson-machine-learning-cl
ient) (0.10.0)
Requirement already satisfied: s3transfer<0.6.0,>=0.5.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from boto3->watson-machine-learning-
client) (0.5.0)
Requirement already satisfied: botocore<1.22.0,>=1.21.21 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from boto3->watson-machine-learning
-client) (1.21.41)
Requirement already satisfied: python-dateutil<3.0.0,>=2.1 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from botocore<1.22.0,>=1.21.21->b
oto3->watson-machine-learning-client) (2.8.2)
Requirement already satisfied: six>=1.5 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from python-dateutil<3.0.0,>=2.1->botocore<1.22.0,>=
1.21.21->boto3->watson-machine-learning-client) (1.15.0)
Requirement already satisfied: ibm-cos-sdk-s3transfer==2.11.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk->watson-machi
ne-learning-client) (2.11.0)
Requirement already satisfied: ibm-cos-sdk-core==2.11.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk->watson-machine-lea
rning-client) (2.11.0)
Requirement already satisfied: charset-normalizer~=2.0.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from requests->watson-machine-learn
ing-client) (2.0.4)
Requirement already satisfied: idna<4,>=2.5 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from requests->watson-machine-learning-client)
(3.3)
Requirement already satisfied: pytz>=2017.3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from pandas->watson-machine-learning-client) (20
21.3)
Requirement already satisfied: numpy>=1.17.3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from pandas->watson-machine-learning-client)
(1.20.3)
```



```
In [ ]: !pip install watson-machine-learning-client --upgrade
```

```
In [ ]: from ibm_watson_machine_learning import APIClient

        wml_credentials={
            "url" : "https://us-south.ml.cloud.ibm.com",
            "apikey" : "ZZgSC4ur1cu00Qe8e_nvfyIz0mbwm0yNT0QU0LujkAO4"
        }
        client=APIClient(wml_credentials)
```



Models

Space Details

Name
Models

Description
No description provided.

Space GUID
850d4cb3-0089-4c23-91cd-837027d44...

Date created
Nov 15, 2022, 6:23 PM
by Monisha S (You)

Last updated
Nov 15, 2022, 6:29 PM

Deployment space tags
No tags are set to this space.

Cloud Object Storage

Storage used
0 Bytes used

Name
Cloud Object Storage-ek

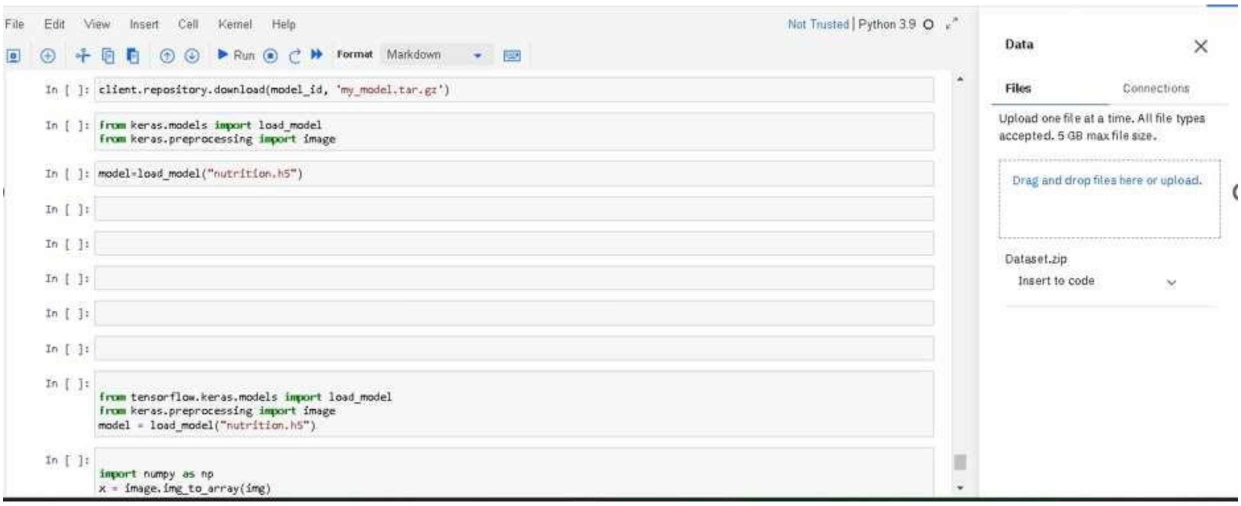
Bucket
41f590f7-b783-4bee-845b-78c99d5981e

Machine learning service

Watson Machine Learning-p

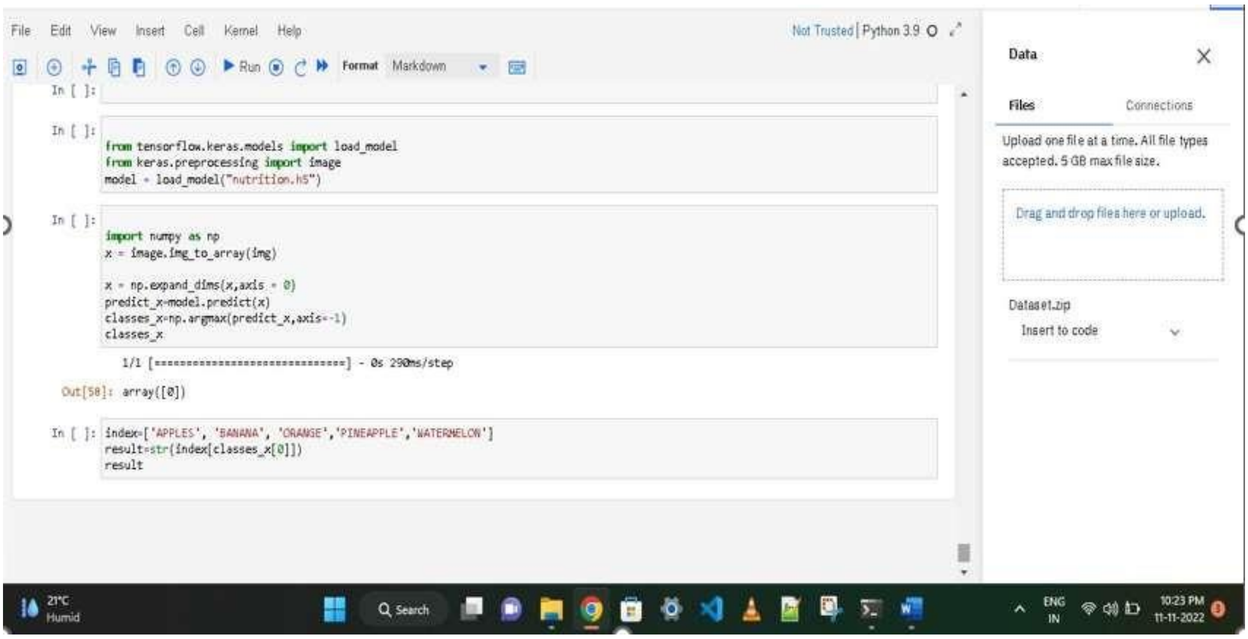
Drop files here or browse for files to upload.

Stay on the page until upload completes. Incomplete uploads are cancelled.



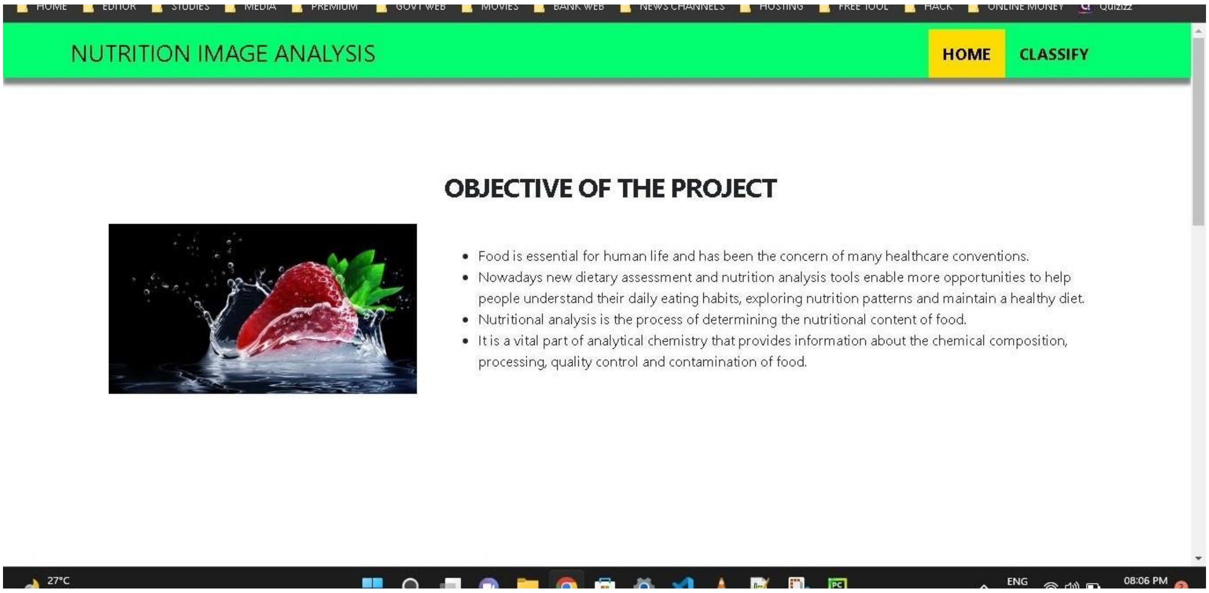
Upload one file at a time. All file types accepted. 5 GB max file size.

```
In [ ]: client.repository.download(model_id, 'my_model.tar.gz')

In [ ]: from keras.models import load_model
        from keras.preprocessing import image

In [ ]: model=load_model("nutrition.h5")

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:
        from tensorflow.keras.models import load_model
        from keras.preprocessing import image
        model = load_model("nutrition.h5")

In [ ]:
        import numpy as np
        x = image.img_to_array(img)
```



Upload one file at a time. All file types accepted. 5 GB max file size.

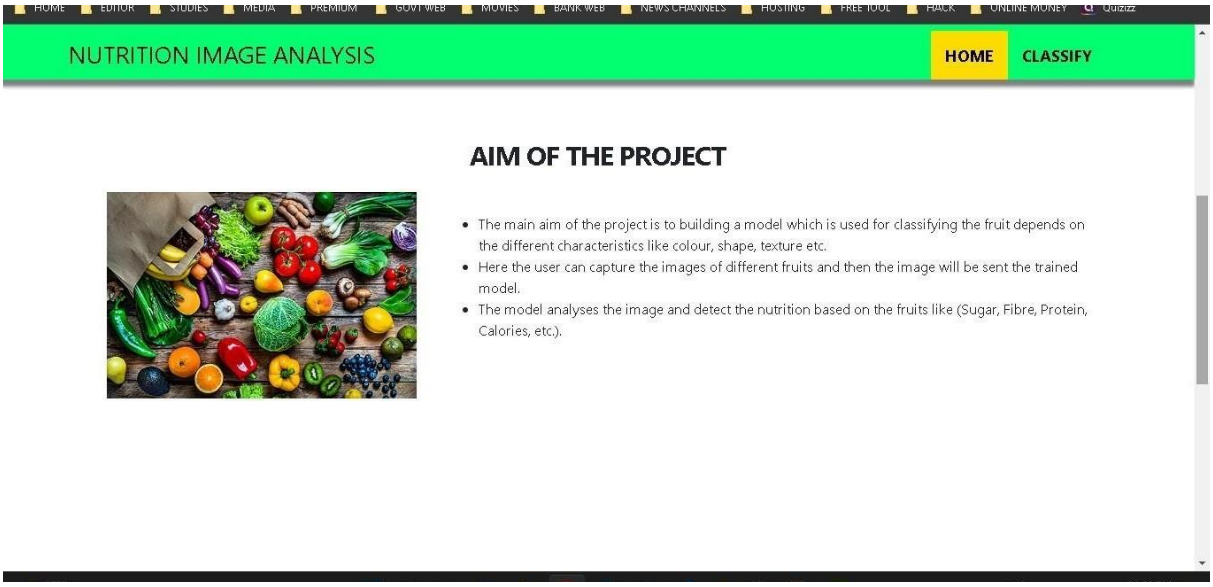
```
In [ ]:

In [ ]:
        from tensorflow.keras.models import load_model
        from keras.preprocessing import image
        model = load_model("nutrition.h5")

In [ ]:
        import numpy as np
        x = image.img_to_array(img)

        x = np.expand_dims(x,axis = 0)
        predict_x=model.predict(x)
        classes_x=np.argmax(predict_x,axis=-1)
        classes_x

           1/1 [==============================] - 0s 290ms/step

Out[58]: array([0])

In [ ]: index=['APPLES', 'BANANA', 'ORANGE','PINEAPPLE','WATERMELON']
        result=str(index[classes_x[0]])
        result
```



NUTRITION IMAGE ANALYSIS

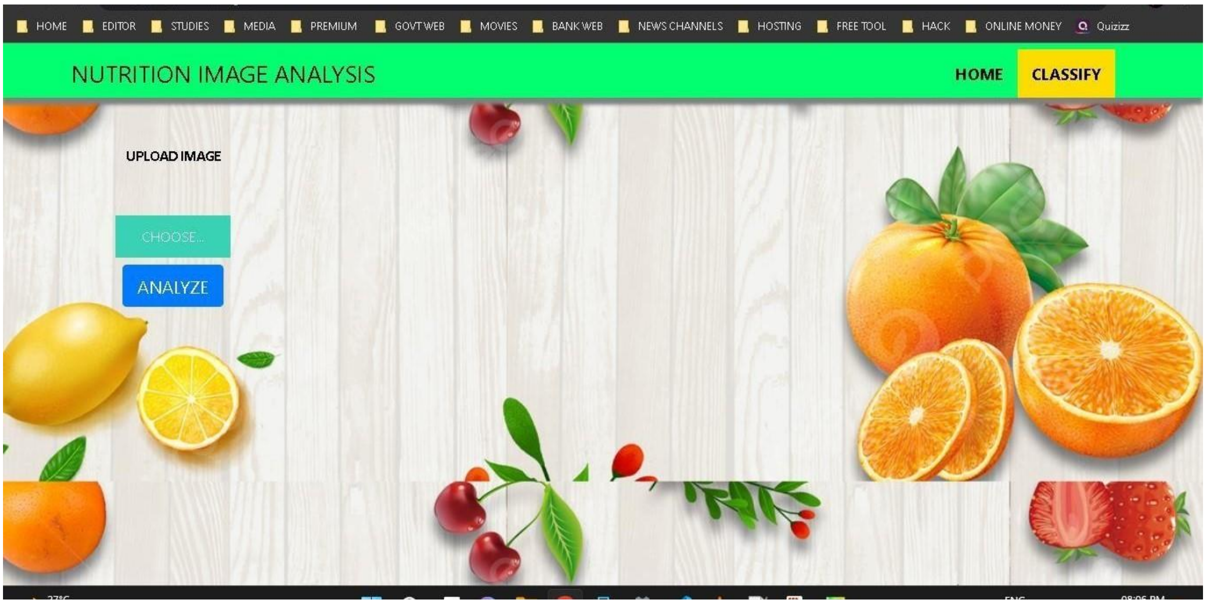
OBJECTIVE OF THE PROJECT

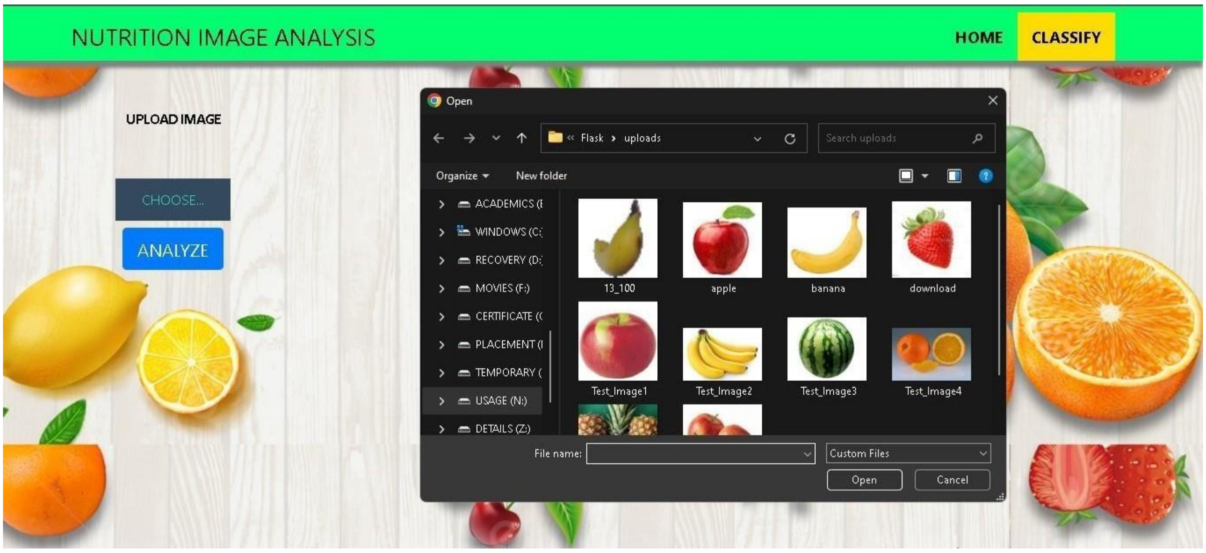
- Food is essential for human life and has been the concern of many healthcare conventions.
- Nowadays new dietary assessment and nutrition analysis tools enable more opportunities to help people understand their daily eating habits, exploring nutrition patterns and maintain a healthy diet.
- Nutritional analysis is the process of determining the nutritional content of food.
- It is a vital part of analytical chemistry that provides information about the chemical composition, processing, quality control and contamination of food.

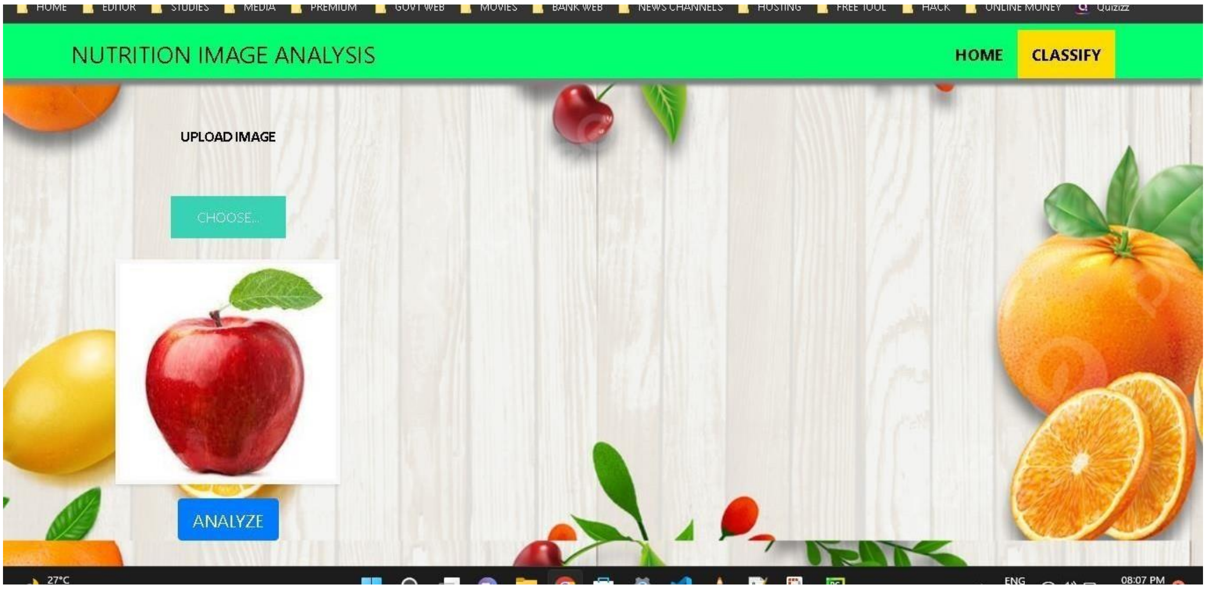


AIM OF THE PROJECT

- The main aim of the project is to building a model which is used for classifying the fruit depends on the different characteristics like colour, shape, texture etc.
- Here the user can capture the images of different fruits and then the image will be sent the trained model.
- The model analyses the image and detect the nutrition based on the fruits like (Sugar, Fibre, Protein, Calories, etc.).




NUTRITION IMAGE ANALYSIS
HOME
CLASSIFY
UPLOAD IMAGE
CHOOSE...
ANALYZE
Open
Flask › uploads
Search uploads
Organize
New folder
ACADEMICS (I
WINDOWS (C:
RECOVERY (D:
MOVIES (F:)
CERTIFICATE (
PLACEMENT (I
TEMPORARY (
USAGE (N:)
DETAILS (Z:)
13_100
apple
banana
download
Test_Image1
Test_Image2
Test_Image3
Test_Image4
File name:
Custom Files
Open
Cancel


HOME
EDITOR
STUDIES
MEDIA
PREMIUM
GOVT WEB
MOVIES
BANK WEB
NEWS CHANNELS
HOSTING
FREE TOOL
HACK
ONLINE MONEY
Quizizz
NUTRITION IMAGE ANALYSIS
HOME
CLASSIFY
UPLOAD IMAGE
CHOOSE...
ANALYZE
27°C
ENG
08:07 PM

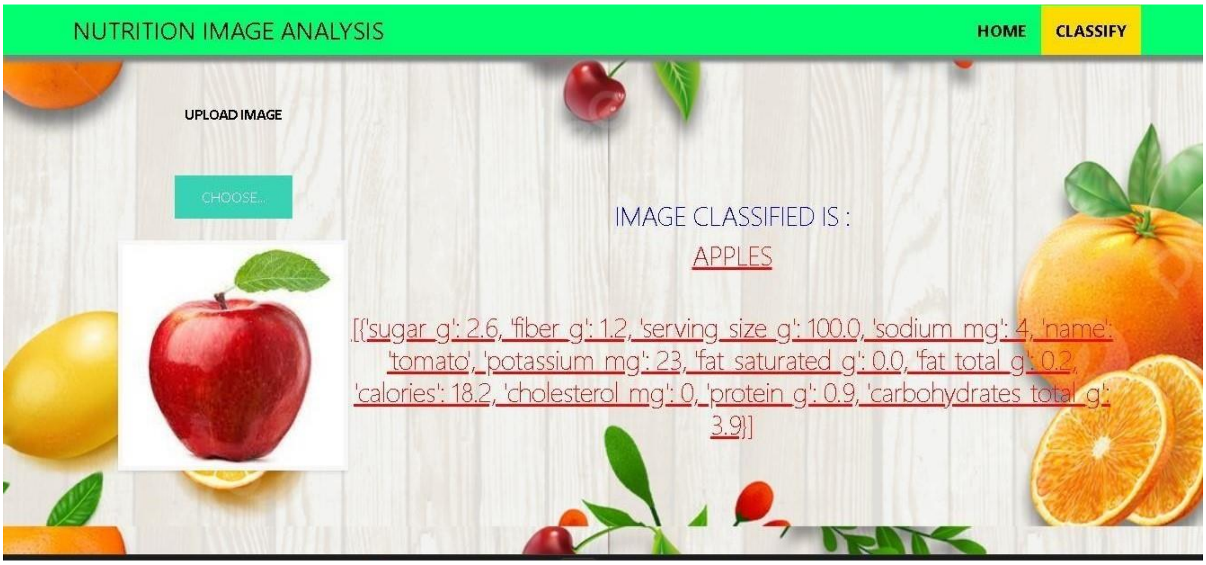


IMAGE CLASSIFIED IS :

APPLES

```
[{'sugar_g': 2.6, 'fiber_g': 1.2, 'serving_size_g': 100.0, 'sodium_mg': 4, 'name':
'tomato', 'potassium_mg': 23, 'fat_saturated_g': 0.0, 'fat_total_g': 0.2,
'calories': 18.2, 'cholesterol_mg': 0, 'protein_g': 0.9, 'carbohydrates_total_g':
3.9}]
```