

Implementing Web Application Integrate Nutrition API

Date	18 November 2022
Team ID	PNT2022TMID50239
Project Name	Nutrition Assistant Application

Clarifai:

The screenshot displays the Clarifai Community web application interface. The top navigation bar includes 'My Apps' and 'Community' links, along with buttons for 'Switch to Legacy Portal', 'Create an App', and a user profile icon labeled 'NS'. The left sidebar shows a menu with 'Account', 'Billing', 'Security', and 'Usage' options. The main content area is titled 'Account Settings' and contains two sections: 'Contact Information' and 'Profile'. The 'Contact Information' section includes fields for First Name (Nithisha), Last Name (S), Company (Dr. Sivanthi Aditanar College Of Engineering), Country (India), Job Title (Student), and Job Role (Other). An 'Update Contact Information' button is located below these fields. The 'Profile' section shows the User ID (clarifaiforme) and a 'Change User id' button. The bottom of the image shows a Windows taskbar with various application icons and system status information.

Clarifai Community

clarifai.com/settings

My Apps Community

Switch to Legacy Portal Create an App

NS

NS Nithisha S

Account

Billing

Security

Usage

Account Settings

Contact Information

First Name * Nithisha

Last Name * S

Company * Dr. Sivanthi Aditanar College Of Engineering

Country * India

Job Title * Student

Job Role * Other

Update Contact Information

Profile

User ID clarifaiforme

Change User id

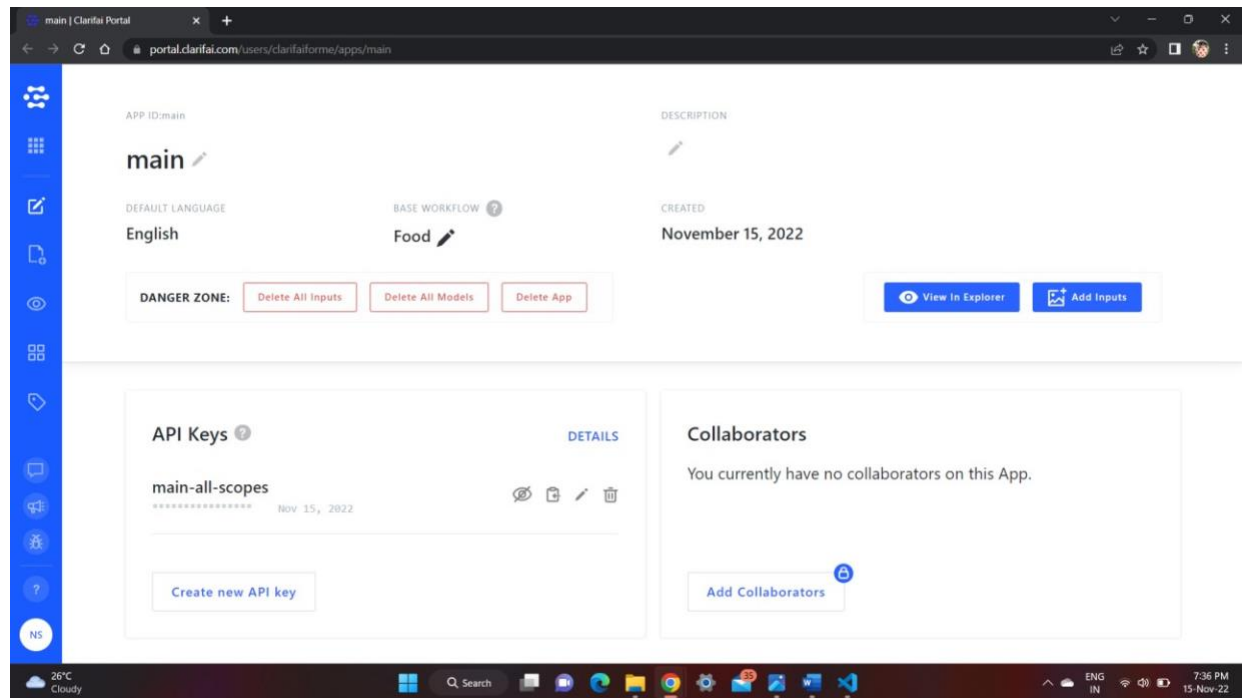
26°C Cloudy

Search

ENG IN

8:23 PM 15-Nov-22

Clarifai API:



Python code to integrate with the Nutrition API:

```

from clarifai_grpc.channel.clarifai_channel import ClarifaiChannel from
clarifai_grpc.grpc.api import resources_pb2, service_pb2, service_pb2_grpc
from clarifai_grpc.grpc.api.status import status_pb2, status_code_pb2 import
os
channel =
ClarifaiChannel.get_grpc_channel() stub =
service_pb2_grpc.V2Stub(channel)

metadata = (('authorization', 'Key ' + ' '),)
userDataObject =
resources_pb2.UserAppIDSet(user_id='clarifaiforme', app_id='main')

def food_identifier(string):
with open(string, "rb") as f:
    file_bytes = f.read()
    post_model_outputs_response =
stub.PostModelOutputs(
service_pb2.PostModelOutputsRequest(
user_app_id=userDataObject,
model_id="food-item-recognition",

```

```

        version_id="1d5fd481e0cf4826aa72ec3ff049e044
inputs=[
    resources_pb2.Input(
        data=resources_pb2.Data(
            image=resources_pb2.Image(
                base64=file_bytes
            )
        )
    )
],
    ),
metadata=metadata
    )
    if post_model_outputs_response.status.code != status_code_pb2.SUCCESS:
        print("There was an error with your request!")
print("\tCode:
{}".format(post_model_outputs_response.outputs[0].status.code))
print("\tDescription:
{}".format(post_model_outputs_response.outputs[0].status.description))
print("\tDetails:
{}".format(post_model_outputs_response.outputs[0].status.details))
raise Exception("Post model outputs failed, status: " +
post_model_outputs_response.status.description)
    output = post_model_outputs_response.outputs[0]
    print("Predicted concepts:")
    for
concept in output.data.concepts:
        print("\t%s %.2f" % (concept.name, concept.value))

    return output.data.concepts[0].name

```