



**RMK ENGINEERING COLLEGE**



**(An Autonomous Institution)**

**R.S.M. Nagar, Kavaraipettai-601 206, Gummidipoondi Taluk,  
Thiruvallur District.**

## **PROJECT**

**AI powered nutrition analyzer for fitness enthusiasts**

**DONE BY**

**TEAM ID: PNT2022TMID15800**

**P JAI SIVA RANJANI – 111719104118**

**ISHWARYA S – 111719104062**

**LAVANYA S – 111719104084**

**MANASA G C - 111719104090**

# 1.INTRODUCTION

## 1.1 Project Overview

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.

## 1.2 .Purpose

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.).

# 2 LITERATURE SURVEY

## 2.1 Existing problem

Nowadays people eat all junk food , they are not having balanced diet .

This affects their health life. Current lifestyle produces many health related problems like obesity , blood pressure . These need to be governed.

## 2.1 References

### 1. THE INFLUENCE OF AI-powered Nutrition Analyzer for Fitness Enthusiasts

Chair and Department of Humanities and Social Medicine, Medical University of Lublin, 20-093 Lublin, Poland

2 BioMolecular Resources Research Infrastructure Poland (BBMRI.pl), Poland  
3 Faculty of Medicine, Medical University of Lublin, 20-059 Lublin, Poland; [magda.suchodolska.1998@interia.eu](mailto:magda.suchodolska.1998@interia.eu)  
\* Correspondence: [jaroslaw.sak@umlub.pl](mailto:jaroslaw.sak@umlub.pl)

Artificial intelligence (AI), from time to time called machine intelligence is simulation of human intelligence in machines. It is the intellect exhibited by machines, in contrast to the natural knowledge demonstrated by humans. It is yet another online platform which uses deep learning and image recognition to analyse what the users eat and determine what is trending in terms of each popular dish that the user eats and consumption time. The machine learning facilitates provisions like recognition of past meals, make hierarchical predictions- that is detect high-level categories like beverages and soup as well as specific dishes and ingredients. It also integrates with their Food Knowledge Graph that contains a large set of commonly eaten foods, with nutrition facts, and hierarchical structure. The platform also further breaks down the nutrition information calories, macro and micro nutrients as well as ingredients.

### 2. Artificial Intelligence in Nutrients Science Research: A Review

Jarosław Sak 1,2,\* and Magdalena Suchodolska 3

Chang Liu, Yu Cao, Senior Member, IEEE, Yan Luo, Member, IEEE, Guanling Chen,  
Member, IEEE, Vinod Vokkarane, Senior Member, IEEE, Yunsheng Ma, Songqing Chen,  
Member, IEEE, Peng Hou

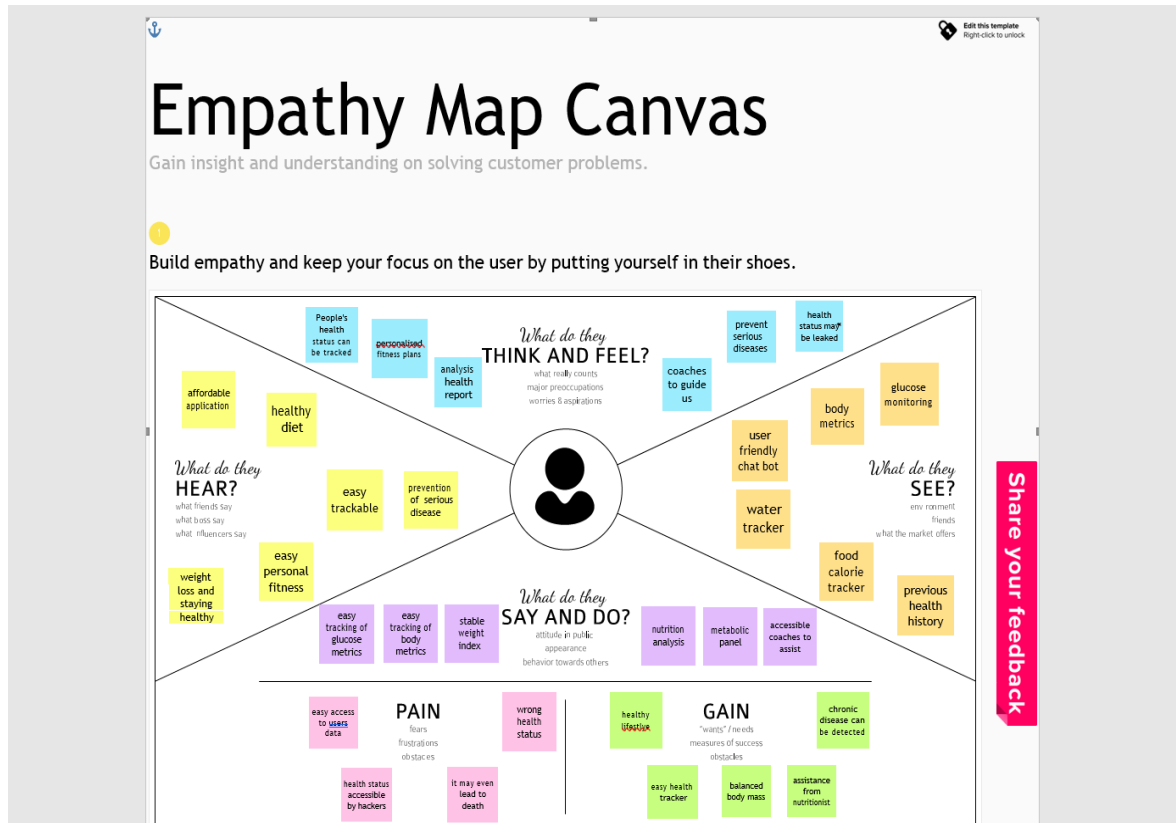
Literature has indicated that accurate dietary assessment is very important for assessing the effectiveness of weight loss interventions. However, most of the existing dietary assessment methods

## **2.3 PROBLEM STATEMENT DEFINITION**

As people across the globe are becoming more health conscious, eating more healthy food and avoiding junk food, a system that can measure calories and nutrition in every day meals can be very useful for maintaining one's health. Food calorie and nutrition measurement system is very beneficial for dieticians and patients to measure and manage their daily food intake. We also know that it's difficult to find an affordable nutritionist or a dietician across the street; therefore, we have proposed a system .

# 3. IDEATION & PROPOSE SOLUTION

## 3.1 EMPATHY MAP CANVAS



## 3.2 IDEATION AND BRAINSTORMING

### Brainstorm & Idea Prioritization Template:

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions. Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

Reference: <https://www.mural.co/templates/empathy-map-canvas>

### Step-1: Team Gathering, Collaboration and Select the Problem Statement

## Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

10 minutes to prepare  
1 hour to brainstorm  
2-8 people recommended

**1 Before you collaborate**

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

2 Then gathering

Before you start, get everyone in the session and send an email with the instructions (or print-out sheet).

3 On the go

Here's what the problem you'll be focusing on during the 10-minute gathering.

4 Let's get started

Let's get started with the brainstorming. We'll be focusing on the problem you've just defined.

Open session

**1 Define your problem statement**

Nowadays people are not concerned about what they are eating. They don't know what nutrition is, present in the food, what they are getting. This leads to serious health issues. It's important to be health conscious. So we have come up with a plan to track their health status.

**PROBLEM**

Just foods are found in the associated with obesity due to their high sugar content and the amount of fat present in the sugar, chemical additives, and sodium with the presence of a low amount of microelements and fiber. The complex diet structure which is a serious concern to be evident. It leads to various health issues like obesity, diabetes and high blood pressure.

**Key notes of understanding**

To set up smooth and productive session

- Don't talk.
- Encourage wild ideas.
- Defer judgment.
- Let everyone share.
- Go for quantity.
- One idea leads to another.

## Step-2: Brainstorm, Idea Listing and Grouping

**2 Brainstorm**

Write down any ideas that come to mind that address your problem statement.

15 minutes

**3 Group ideas**

Take turns sharing your ideas while clustering similar or related notes as you go. In the last 10 minutes, give each cluster a sentence-like label. If a cluster is bigger than 10 sticky notes, try and see if you can break it up into smaller sub-groups.

10 minutes

**Jai siva**

Health status would be monitored  
Images of food can be captured  
They can even select the images  
There will be a dashboard for sharing health tips

**Levansia**

Sugar level will be checked periodically  
It classifies the nutrients in the food  
Correct feedback would be given  
Apply the change in their dashboard

**DOROSIA**

Heart rate and pulse rate are checked frequently  
Motivational daily quotes in the dashboard  
Chatbot to interact with the user  
Previous health record

**Idvaxax**

Daily diet plans  
Indication of harmful food  
Tracking the progress  
Feedback of particular user

**User account**

Username and password  
Personal details  
Setting their goal  
Profile picture if necessary

**Get input from user**

user's height and weight  
medical report  
Snap of the food  
Name of the food

**Nutrition Assistant application**

**Suggesting diets**

Recipes according to diet  
Identifying harmful ingredients in the food  
Sharing health tips  
Classify nutritional content present in the food

**Monitoring progress**

Tracking their intake of food  
Calorie monitoring and BMI calculation  
Diet monitoring  
Monitoring the health condition

## Step-3: Idea Prioritization

**4 Prioritize**

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes

**5 After you collaborate**

You can export the mural as an image or pdf to share with members of your company who might find it helpful.

**Quick add-ons**

1 Share the mural  
Share a new link to the mural with stakeholders to keep them in the loop about the outcomes of the session.

2 Export the mural  
Export a copy of the mural as a PNG or PDF to attach to emails, include in slides, or save to your drive.

**Keep moving forward**

Strategy blueprint  
Define the components of a new idea or strategy.  
Open the template

Customer experience journey map  
Understand customer needs, motivations, and obstacles for an experience.  
Open the template

Strengths, weaknesses, opportunities & threats  
Identify strengths, weaknesses, opportunities, and threats (SWOT) to develop a plan.  
Open the template

Share template feedback

**Importance**

If each of these items is a goal, then the items that are most important are the ones that are most important to you.

**Feasibility**

Feasibility of their importance, which leads to more realistic outcomes (time, cost, effort, complexity, etc.)

Classify nutritional content present in the food

Calorie monitoring and BMI calculation

Diet monitoring

Monitoring the health condition

Recipes according to diet

user's height and weight

Personal details

medical report

Setting their goal

Snap or name of the food

### 3.3 PROPOSED SOLUTION

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	<ul style="list-style-type: none"><li>• 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.</li><li>• Here the user can capture the images of different fruits and then the image will be sent the trained model.</li><li>• The model analyses the image and detect the nutrition based on the fruits like (Sugar, Fibre, Protein, Calories, etc.).</li></ul>
2.	Idea / Solution description	<ul style="list-style-type: none"><li>• The user logs in to the system using the login credentials.</li><li>• The user interacts with the UI (User Interface) and give the image as input.</li><li>• Then the input image is passed to the flask application.</li><li>• Next is to develop an AI model to capture the image and recognize the fruit and find it's nutritional values.</li><li>• And finally with the help of the model which we build we will classify the result and showcase it on the UI.</li></ul>

3.	Novelty / Uniqueness	<ul style="list-style-type: none"> <li>• This model can capture images and detect the nutritional values of the fruits and showcase it to the user.</li> <li>• The model collects the user input and prepares diet charts for the user.</li> <li>• Personalized fitness and diet plans for the user.</li> </ul>
4.	Social Impact / Customer Satisfaction	<ul style="list-style-type: none"> <li>• Helps the fitness enthusiasts to find the nutritional value of the fruits they're consuming which will help them to maintain a balanced diet by taking adequate amount of nutrients needed every day.</li> <li>• To consult with the nutritionist and personal trainers for their diet plans and training schedules.</li> <li>• Generalised diet and fitness plans to all the users based on their goals.</li> </ul>
5.	Business Model (Revenue Model)	<ul style="list-style-type: none"> <li>• Specific diet and fitness plans according to their body type and their goals for the premium subscription members.</li> <li>• Personal trainers to help the fitness enthusiasts with the workouts.</li> <li>• Personal nutritionist for the fitness enthusiast to help with diet plans based on their fitness goals.</li> </ul>
6.	Scalability of the Solution	<ul style="list-style-type: none"> <li>• The solution is developed in the such a way that we can update the system without disturbing the current model.</li> <li>• All the future enhancements can be added to the system without changing the model.</li> </ul>



## 3.4 PROBLEM SOLUTION FIT

Problem-Solution Fit canvas		Purpose / Vision	To enhance people's lifestyle	Version:
Define CS, fit into CL	<b>1. CUSTOMER SEGMENT(S)</b> <span>CS</span> People who are in need to enhance their lifestyle and keep their lifestyle balanced. Those who want to follow the healthy diet .	<b>6. CUSTOMER LIMITATIONS</b> <span>CL</span> <small>EG. BUDGET, DEVICES</small> Sometimes the images may not be processed well. In that case it may produce wrong analysis of the food.	<b>5. AVAILABLE SOLUTIONS</b> <span>AS</span> <small>PLUSES &amp; MINUSES</small> The packed food may contain the nutrition content of the food, but it may not be apt according to the customers diet.	Explore AS, differentiate
	<b>2. PROBLEMS / PAINS + ITS FREQUENCY</b> <span>PR</span> Nowadays people are not tracking their personal nutrition diet because of these we are witnessing many health related issues like obesity , diabetics , blood pressure .	<b>9. PROBLEM ROOT / CAUSE</b> <span>RC</span> People diet is filled with excess of sucrose ,fatty acids , as they have started to eat junk food users need to control their daily calorie intake to lead a healthy lifestyle.	<b>7. BEHAVIOR + ITS INTENSITY</b> <span>BE</span> The behavioral changes in users reflect in their day- to-day life such as they will maintain a proper diet and follow the daily routine in eating and intake of healthy food. So, that it helps them to improve their health	
Identify strong TR & EM	<b>3. TRIGGERS TO ACT</b> <span>TR</span> People who have already followed our healthy diet and maintained a healthy body. This inspires other people.	<b>10. YOUR SOLUTION</b> <span>SL</span> Our solution comes with tracking their nutritional diet. The food tracker also maintains a healthy diet for people using AI-based tracker. This helps people to maintain a healthy lifestyle.	<b>8. CHANNELS of BEHAVIOR</b> <span>CH</span> ONLINE The application provides a user-friendly environment that enables users to interact through chatbot to clarify their queries and a dashboard is displayed to know the activities	Extract online & offline CH of BE
	<b>4. EMOTIONS</b> <span>EM</span> <small>BEFORE / AFTER</small> When people starts to get a healthy body with correct body mass then they would be happy to go for continuing it.		OFFLINE Connecting all the users through offline meeting and giving some complimentary gifts. Conducting offline session by nutrition expert.	

## 4 REQUIREMENT ANALYSIS

### 4.1 FUNCTIONAL REQUIREMENT

#### **FUNCTIONAL REQUIREMENT:**

FOLLOWING ARE THE FUNCTIONAL REQUIREMENTS OF THE PROPOSED SOLUTION.

FUNCTION NUMBER	FUNCTIONAL REQUIREMENT	SUB REQUIREMENT
FN.NO-1	USER REGISTRATION/LOGIN	REGISTER AFTER OPENING THE APPLICATION
FN.NO-2	USER DETAIL	PROVIDE DETAIL ABOUT HEALTH STATUS AFTER REGISTRATION BY SELECTING AND SPECIFY ABOUT IT
FN.NO-3	CAPTURING IMAGE	CAPTURE THE IMAGE AND CHECK THE PARAMETERS OF THE CAPTURED IMAGE.
FN.NO-4	IMAGE PROCESSING	UPLOAD THE IMAGE FOR PROCESSING
FN.NO-5	IMAGE IDENTIFICATION	IDENTIFY THE FOOD OR DRINK PROVIDED IN THE IMAGE.
FN.NO-6	IMAGE DESCRIPTION	PROVIDE THE NUTRITIONAL CONTENTS OF THE FOOD OR DRINK IDENTIFIED.

## **4.2 NON-FUNCTIONAL REQUIREMENT:**

FOLLOWING ARE THE NON-FUNCTIONAL REQUIREMENT OF THE PROPOSED SOLUTION

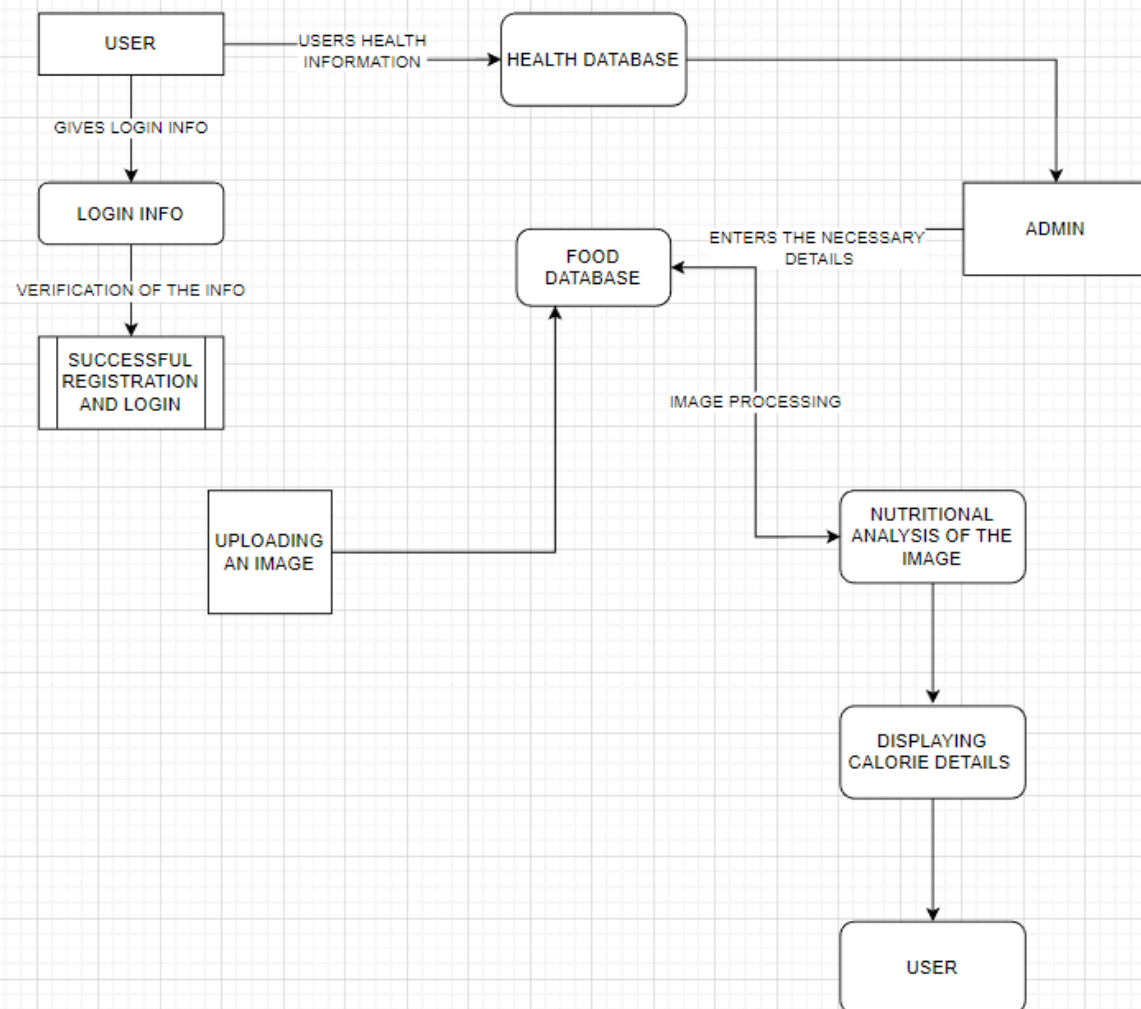
NON - FUNCTION NUMBER	NON-FUNCTIONAL REQUIREMENT	SUB REQUIREMENT
NFN.NO-1	USABILITY	DATASET OF ALL FOOD/DRINK TO IDENTIFY THE NUTRITIONAL CONTENT OF THE ITEM.
NFN.NO-2	SECURITY	THE INFORMATION AND THE HEALTH STATUS OF USER AND NUTRITIONAL DETAILS ABOUT FOOD ARE SECURED HIGHLY.
NFN.NO-3	RELIABILITY	THE IMAGE QUALITY IS IMPORTANT TO PROVIDE THE NUTRITIONAL DETAILS ABOUT FOOD
NFN.NO-4	PERFORMANCE	PERFORMANCE IS BASED ON THE FOOD THAT IS SCANNED.
NFN.NO-5	AVAILABILITY	IT IS AVAILABLE FOR ALL THE USER TO DETECT THE NUTRITIONAL DETAILS ABOUT FOOD / DRINK.
NFN.NO-6	SCALABILITY	INCREASE THE PREDICTION OF NUTRITIONAL DETAILS IN THE FOOD.

## **5. PROJECT DESIGN**

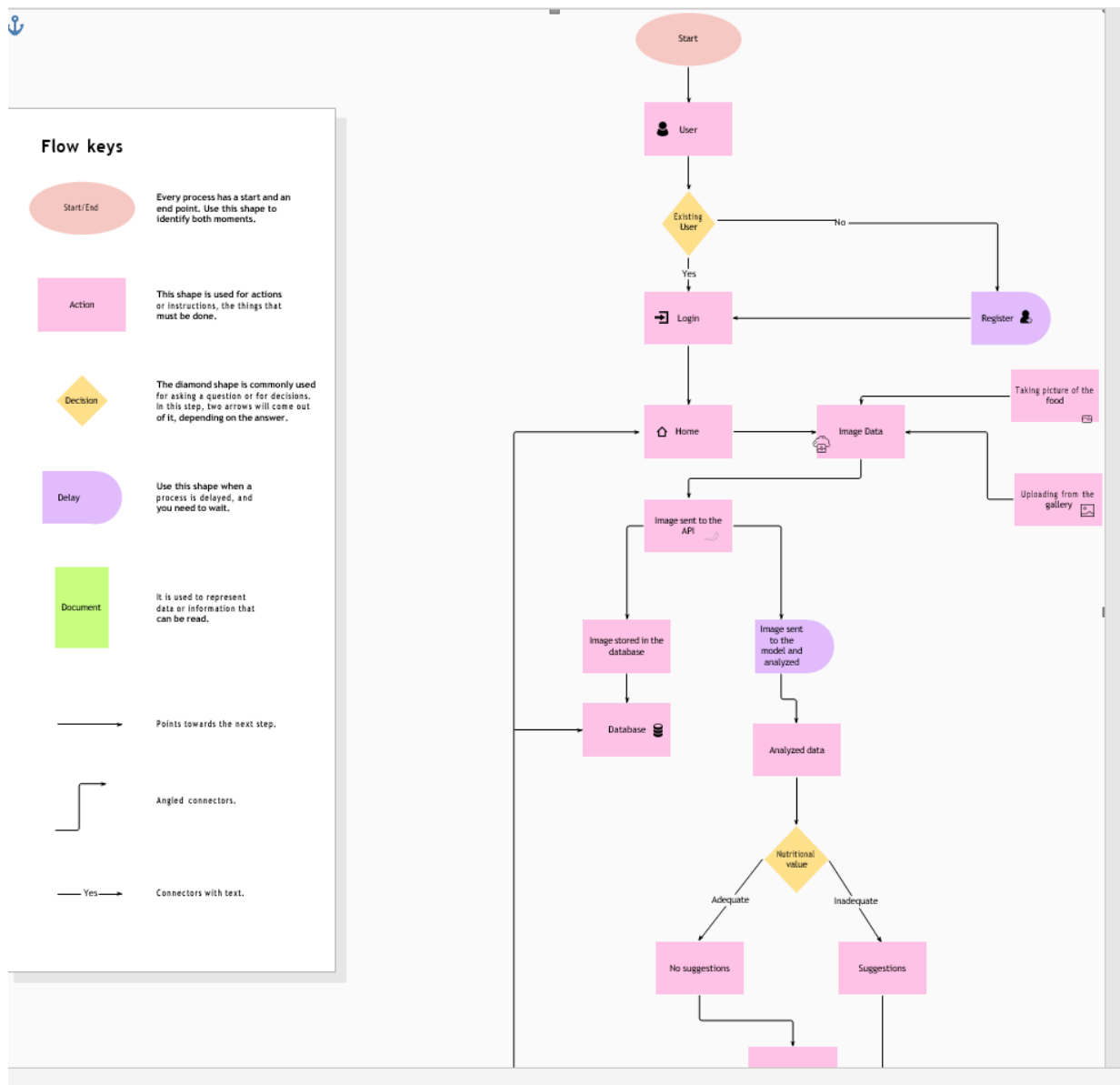
### **5.1 DATA FLOW DIAGRAMS**

#### **Data Flow Diagrams:**

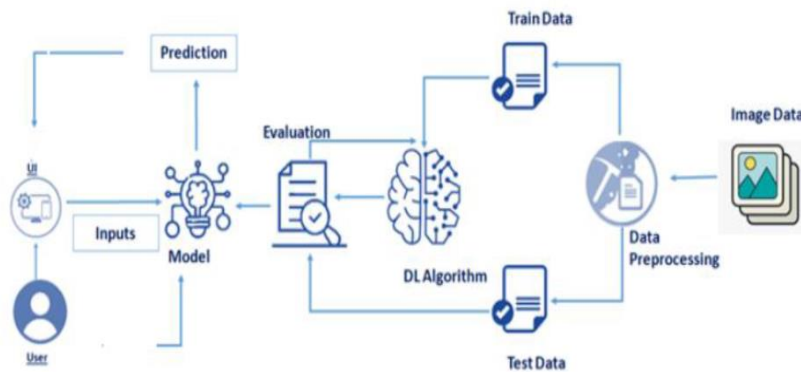
A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



## 5.2 SOLUTION AND TECHNICAL ARCHITECTURE



The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2



## 5.3 USER STORIES

**They were so excited to use this application because it balanced a healthy lifestyle. They were able to track their health status . And maintain a healthy balanced body.**

## 6. PROJECT PLANNING AND SCHEDULING

### 6.1 SPRINT PLANNING AND ESTIMATION


TITLE	DESCRIPTION	DATE
Literature Survey & Information Gathering	Literature survey on the selected project & gathering information by referring the, technical papers, research publications etc.	25 SEPTEMBER 2022
Prepare Empathy Map	Prepare Empathy Map Canvas to capture the user Pains & Gains, Prepare list of problem statements	26 SEPTEMBER 2022
Ideation	List the by organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility & importance.	30 SEPTEMBER 2022
Proposed Solution	Prepare the proposed solution document, which includes the novelty, feasibility of idea, business model, social impact, scalability of solution, etc.	3 OCTOBER 2022

<b>Problem Solution Fit</b>	Prepare problem - solution fit document.	4 SEPTEMBER 2022
<b>Solution Architecture</b>	Prepare solution architecture document.	4 OCTOBER 2022
<b>Customer Journey</b>	Prepare the customer journey maps to understand the user interactions & experiences with the application (entry to exit).	7 OCTOBER 2022
<b>Functional Requirement</b>	Prepare the functional requirement document.	13 OCTOBER 2022
<b>Data Flow Diagrams</b>	Draw the data flow diagrams and submit for review.	14 OCTOBER 2022
<b>Technology Architecture</b>	Prepare the technology architecture diagram.	WORK IN PROGRESS
<b>Prepare Milestone &amp; Activity List</b>	Prepare the milestones & activity list of the project.	31 OCTOBER 2022
<b>Project Development - Delivery of Sprint-1, 2, 3 &amp; 4</b>	Develop & submit the developed code by testing it.	WORK IN PROGRESS



## 6.2 SPRINT DELIVERY SCHEDULE

<b>Sprint</b>	<b>Functional Requirement (Epic)</b>	<b>User Story Number</b>	<b>User Story / Task</b>	<b>Story Points</b>	<b>Priority</b>	<b>Team Members</b>
Sprint-1	Data Collection	USN-1	Download the food nutrition dataset	2	High	Jai Siva Ranjani
Sprint-1	Data Preprocessing	USN-2	Importing the Dataset into Workspace	1	Medium	Jai Siva Ranjani
Sprint-1		USN-3	Handling Missing data	3	Low	Jai Siva Ranjani
Sprint-1		USN-4	Feature Scaling	3	Medium	Jai Siva Ranjani
Sprint-1		USN-5	Data Visualization	3	Low	Jai Siva Ranjani
Sprint-1		USN-6	Splitting Data into Train and set	4	High	Jai Siva Ranjani
Sprint-1		USN-7	Creating A Dataset with Sliding Windows	4	Medium	Jai Siva Ranjani
Sprint-2	Model Building	USN-8	Importing The Model	1	HIGH	Ishwarya

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
			Building Libraries			
Sprint-2 		USN-9	Initializing The Model	1	Medium	Ishwarya
Sprint-2		USN-10	Adding CNN Layers	2	High	Ishwarya
Sprint-2		USN-11	Adding Dense Layers	3	Low	Ishwarya
Sprint-2		USN-12	Configure The Learning Process	4	Medium	Ishwarya
Sprint-2		USN-13	Train the model	2	Medium	Ishwarya
Sprint-2		USN-14	Save the model	2	Medium	Ishwarya
Sprint-2		USN-15	Test the model	3	High	Ishwarya
Sprint-3	Application Building	USN-16	Create an HTML file	4	Medium	Lavanya
Sprint-3		USN-17	Build Python code	4	High	Manasa
Sprint-3		USN-18	Creating our flask application & loading our model using local model method	4	Medium	Manasa

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-3		USN-19	Run the application	4	High	Manasa , Lavanya
Sprint-4	Train the model on IBM	USN-20	Register for IBM Cloud	4	Medium	Jai SivaRanjani, Ishwarya, Lavanya, Manasa
Sprint-4		USN-21	Train the ML Model on IBM	4	High	Manasa
Sprint-4		USN-22	Integrate Flask with scoring End Point	8	High	Manasa

#### Project Tracker, Velocity & Burndown Chart:

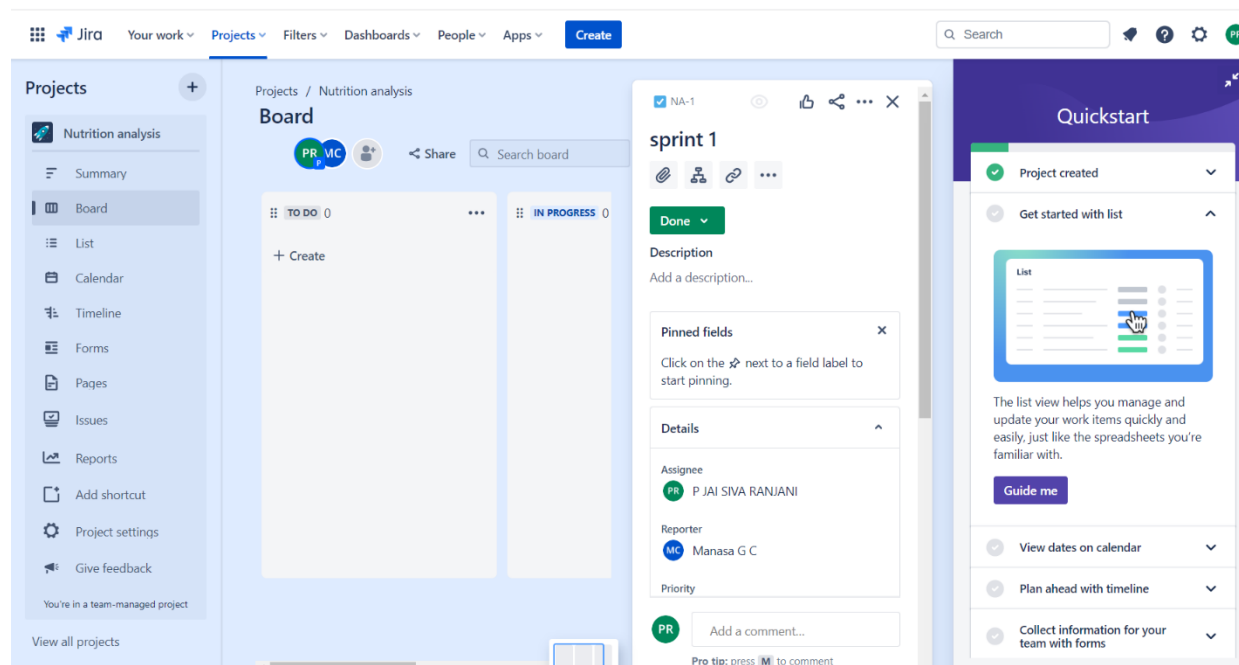
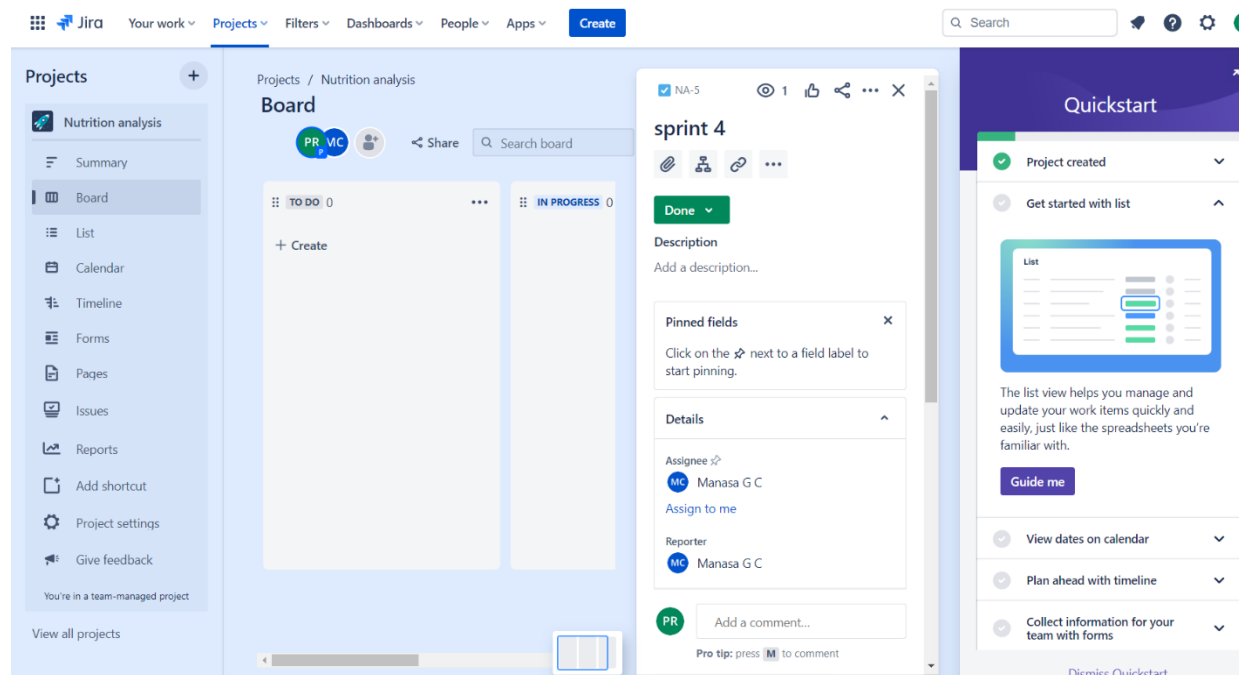
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

#### Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

## 6.3 REPORTS FROM JIRA

Jira Software is part of a family of products designed to **help teams of all types manage work**. Originally, Jira was designed as a bug and issue tracker. But today, Jira has evolved into a powerful work management tool for all kinds of use cases, from requirements and test case management to agile software development.













## 7. CODING AND SOLUTIONING

### **DATA COLLECTION**

Data Collection Collect images of different food items organized into subdirectories based on their respective names as shown in the project structure. Create folders of types of food items that need to be recognized. In this project, we have collected images of 5 types of food items apples, 'banana', 'orange', , they are saved in the respective subdirectories with their respective names.

DATASET LINK:

<https://drive.google.com/drive/folders/1yNVuLA2hxIstOcDV58enyD74Y9drEs6Y?usp=sharing>

- ▼  MyDrive
  - ▶  106106182
  - ▶  Colab Notebooks
  - ▼  Dataset
    - ▶  TEST\_SET
    - ▶  TRAIN\_SET
    -  IBM\_review.pptx
    -  Nutrition.h5
    -  nutrition(1).h5
    -  photo-1589820296156

My Drive > Dataset > TEST\_SET > APPLES ▾

Files

Name ↑



 n07740461\_20.jpg



 n07740461\_40.jpg



 n07740461\_41.jpg



 n07740461\_51.jpg



 n07740461\_81.jpg



 n07740461\_91.jpg



# Image Preprocessing

we will be improve the image data that suppresses unwilling distortions or enhances some image features important for further processing, although performing some geometric transformations of images like rotation, scaling, translation, etc.

- 1) **Import The ImageDataGenerator Library**
- 2) **Configure ImageDataGenerator Class**
- 3) **Apply Image DataGenerator Functionality ToTrainset And Testset**

The Keras deep learning neural network library provides the capability to fit models using image data augmentation via the ImageDataGenerator class.

Let us import the ImageDataGenerator class from Keras

There are five main types of data augmentation techniques for image data; specifically:

Image shifts via the `width_shift_range` and `height_shift_range` arguments.

The image flips via the `horizontal_flip` and `vertical_flip` arguments.

Image rotations via the `rotation_range` argument

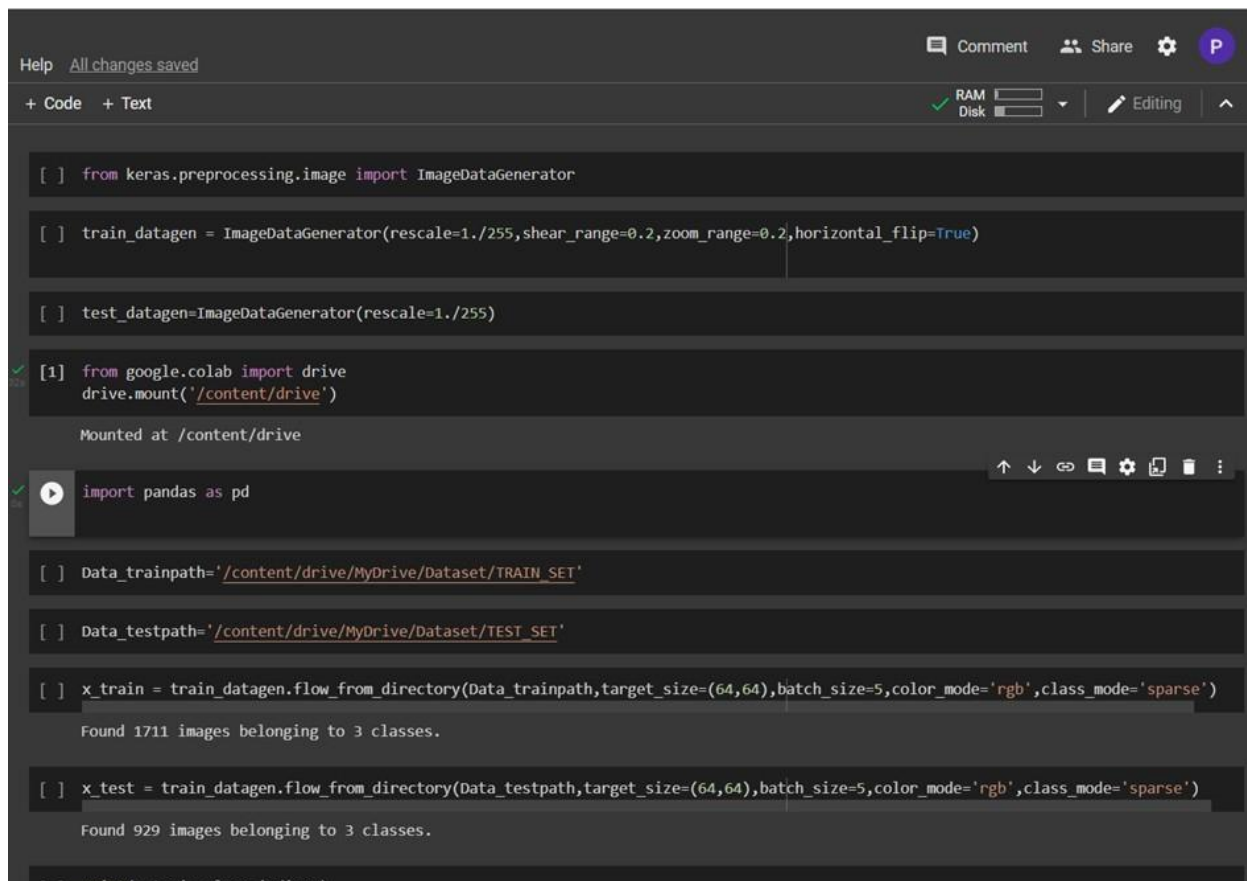
Image brightness via the `brightness_range` argument.

Image zoom via the `zoom_range` argument.

Let us apply ImageDataGenerator functionality to Trainset and Testset by using the following code

For Training set using `flow_from_directory` function.





```
Help All changes saved
+ Code + Text
RAM
Disk
Editing

[ ] from keras.preprocessing.image import ImageDataGenerator

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

[ ] test_datagen=ImageDataGenerator(rescale=1./255)

[1] from google.colab import drive
drive.mount('/content/drive')
Mounted at /content/drive

import pandas as pd

[ ] Data_trainpath='/content/drive/MyDrive/Dataset/TRAIN_SET'

[ ] Data_testpath='/content/drive/MyDrive/Dataset/TEST_SET'

[ ] x_train = train_datagen.flow_from_directory(Data_trainpath,target_size=(64,64),batch_size=5,color_mode='rgb',class_mode='sparse')
Found 1711 images belonging to 3 classes.

[ ] x_test = train_datagen.flow_from_directory(Data_testpath,target_size=(64,64),batch_size=5,color_mode='rgb',class_mode='sparse')
Found 929 images belonging to 3 classes.

[ ] print(x_train.class_indices)
```

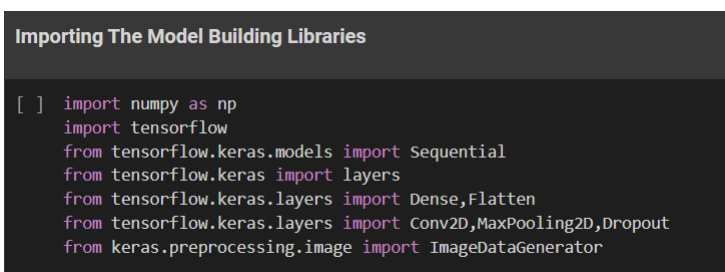
## MODEL BUILDING

Here we are going to build our Convolutional Neural Network which contains an input layer along with the convolution, max- pooling, and finally an output layer.

This is an important step because the model is the main thing needed for prediction.

### □ Importing The Model Building Libraries

We have imported all the necessary libraries needed for model building.



```
Importing The Model Building Libraries

[ ] import numpy as np
import tensorflow
from tensorflow.keras.models import Sequential
from tensorflow.keras import layers
from tensorflow.keras.layers import Dense, Flatten
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Dropout
from keras.preprocessing.image import ImageDataGenerator
```

## □ Initializing The Model

Keras has 2 ways to define a neural network:

- Sequential
- Function API

The Sequential class is used to define linear initializations of network layers which then, collectively, constitute a model. We will use the Sequential constructor to create our model, which will then have layers added to it using the add() method.

### Initializing The Model

```
[ ] model=Sequential()
```

## □ Adding CNN Layers

- As the input image contains three channels, we are specifying the input shape as (64,64,3).
- We are adding a two-convolution layer with an activation function as “relu” and with a small filter size (3,3) and the number of filters (32) followed by a max-pooling layer.
- Max pool layer is used to downsample the input (Max pooling is a pooling operation that selects the maximum element from the region of the feature map covered by the filter)
- Flatten layer flattens the input. Does not affect the batch size.

## Adding CNN Layers

### First Convolution Layer and pooling

```
[ ] model.add(Conv2D(32, (3, 3), input_shape=(64, 64, 3), activation='relu'))  
    model.add(MaxPooling2D(pool_size=(2, 2)))
```

### Second Convolution Layer and pooling

```
▶ model.add(Conv2D(32, (3, 3), activation='relu'))  
    model.add(MaxPooling2D(pool_size=(2, 2)))
```

### Flatten layer

```
[ ] model.add(Flatten())
```

## □ Adding Dense Layers

- A dense layer is a deeply connected neural network layer. It is the most common and frequently used layer.
- The number of neurons in the Dense layer is the same as the number of classes in the training set. The neurons in the last Dense layer, use softmax activation to convert their outputs into respective probabilities.
- Understanding the model is a very important phase to properly use it for training and prediction purposes. Keras provides a simple method, and a summary to get the full information about the model and its layers.

### Adding Dense Layers

```
[ ] model.add(Dense(units=128, activation='relu'))  
    model.add(Dense(units=5, activation='softmax'))
```

```
▶ model.summary()
```

Model: "sequential\_2"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 62, 62, 32)	896
max_pooling2d (MaxPooling2D)	(None, 31, 31, 32)	0
conv2d_1 (Conv2D)	(None, 29, 29, 32)	9248
max_pooling2d_1 (MaxPooling2D)	(None, 14, 14, 32)	0
flatten (Flatten)	(None, 6272)	0
dense (Dense)	(None, 128)	802944
dense_1 (Dense)	(None, 5)	645
Total params: 813,733		
Trainable params: 813,733		
Non-trainable params: 0		

### □ Configure The Learning Process

- The compilation is the final step in creating a model. Once the compilation is done, we can move on to the training phase. The loss function is used to find errors or deviations in the learning process. Keras requires a loss function during the model compilation process.
- Optimization is an important process that optimizes the input weights by comparing the prediction and the loss function. Here we are using adam optimizer.
- Metrics are used to evaluate the performance of your model. It is similar to the loss function, but not used in the training process.

### Configure The Learning Process

```
[ ] model.compile(optimizer='adam', loss='sparse_categorical_crossentropy', metrics=['accuracy'])
```

## □ Train The Model

Now, let us train our model with our image dataset. The model is trained for 15 epochs and after every epoch, the current model state is saved if the model has the least loss encountered till that time. We can see that the training loss decreases in almost every epoch till 15 epochs and probably there is further scope to improve the model.

`fit_generator` functions used to train a deep-learning neural network Arguments:

- `steps_per_epoch`: it specifies the total number of steps taken from the generator as soon as one epoch is finished and the next epoch has started. We can calculate the value of `steps_per_epoch` as the total number of samples in your dataset divided by the batch size.
- `Epochs`: an integer and number of epochs we want to train our model for.
- `validation_data` can be either:
  - an inputs and targets list
  - a generator
  - inputs, targets, and `sample_weights` list which can be used to evaluate the loss and metrics for any model after any epoch has ended.
- `validation_steps`: only if the `validation_data` is a generator then only this argument can be used. It specifies the total number of steps taken from the generator before it is stopped at every epoch and its value is calculated as the total number of validation data points in your dataset divided by the validation batch size.
-

## Save The Model

The model is saved with .h5 extension as follows

An H5 file is a data file saved in the Hierarchical Data Format (HDF). It contains multidimensional arrays of scientific data.

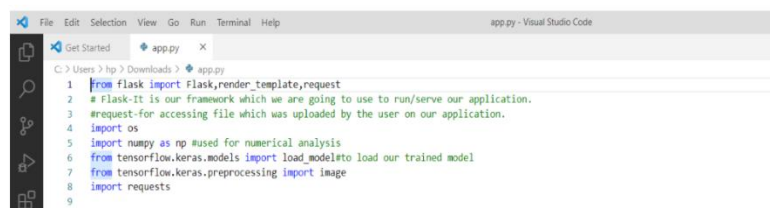
```
Save The Model

[21] model.save('nutrition.h5')
```

## APPLICATION BUILDING

### Build Python Code:

Importing the flask module into the project is mandatory. An object of the Flask class is our WSGI application. Flask constructor takes the name of the current module (\_\_name\_\_) as an argument. Pickle library to load the model file.



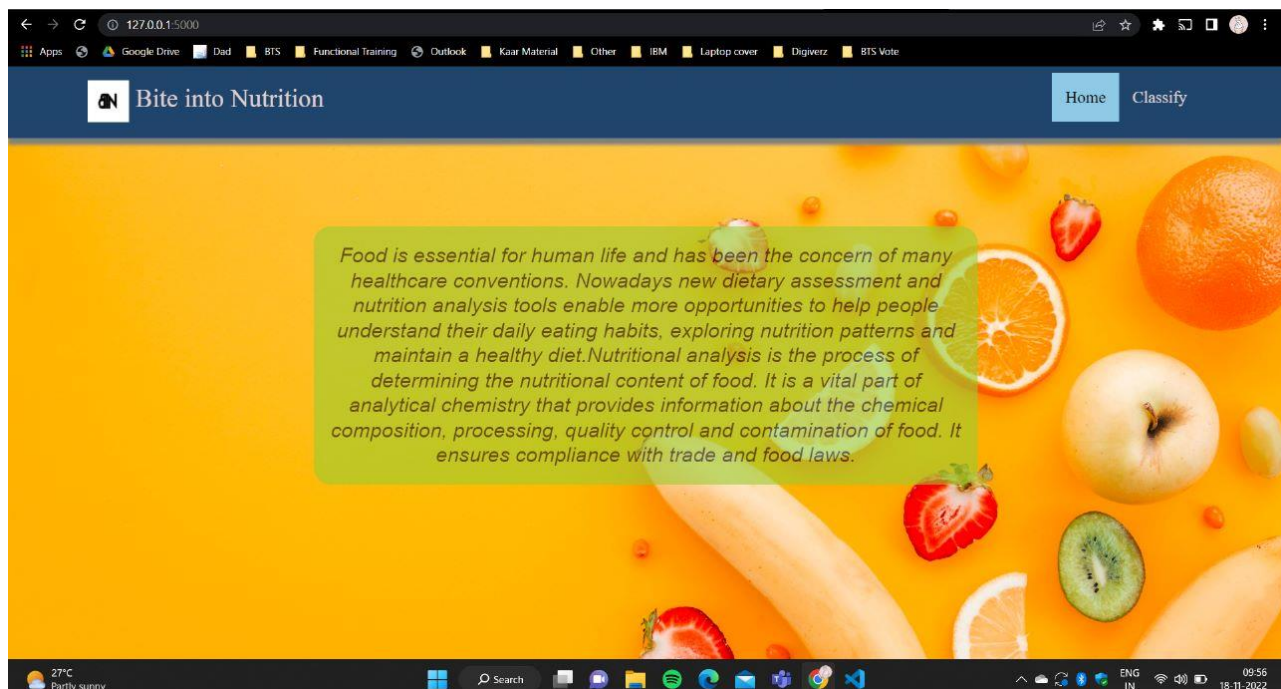
```
File Edit Selection View Go Run Terminal Help app.py - Visual Studio Code
Get Started app.py X
C:\Users\hp> hp> Downloads> app.py
1 from flask import Flask,render_template,request
2 # Flask-It is our framework which we are going to use to run/serve our application.
3 #request-for accessing file which was uploaded by the user on our application.
4 import os
5 import numpy as np #used for numerical analysis
6 from tensorflow.keras.models import load_model to load our trained model
7 from tensorflow.keras.preprocessing import image
8 import requests
9
```

# Creating Our Flask Application And Loading Our Model By Using Load\_model Method

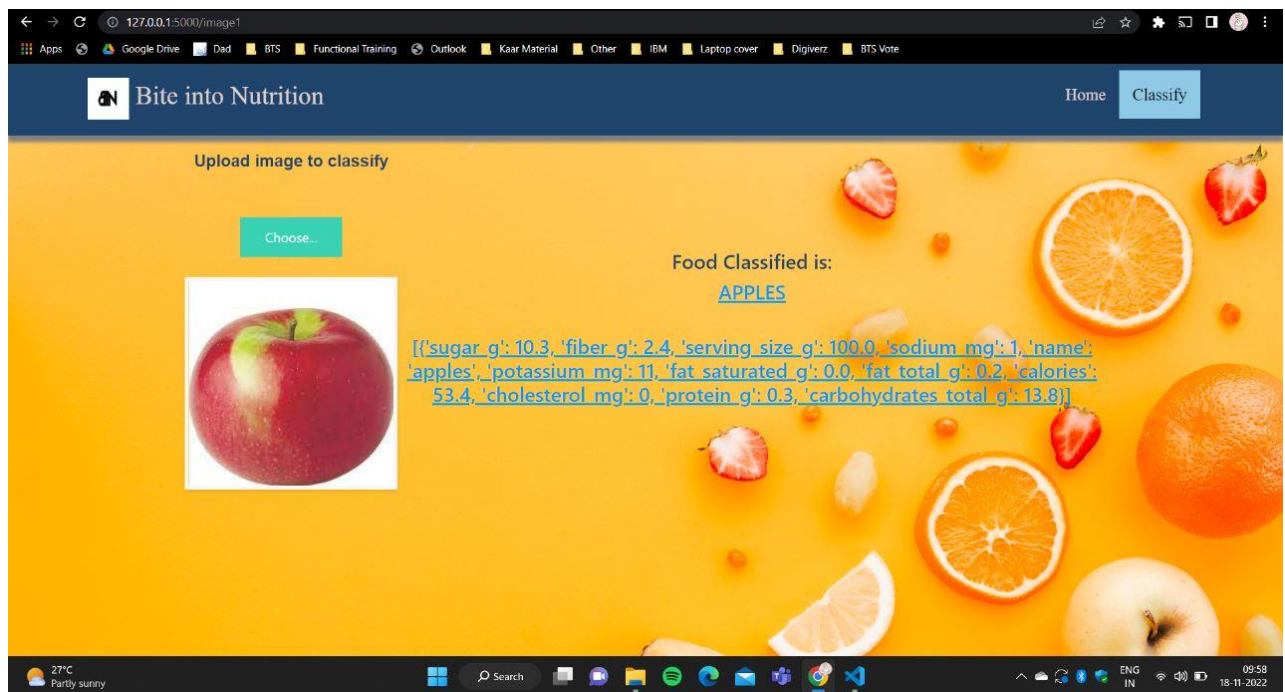
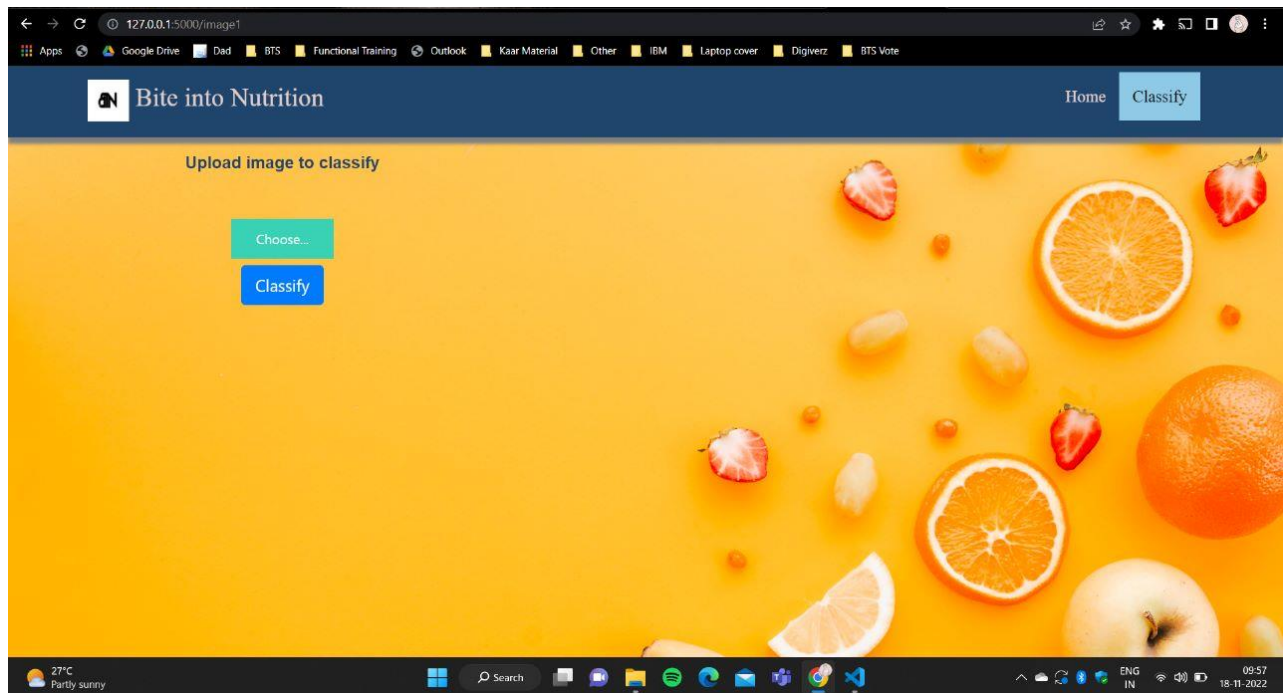
Creating our flask application and loading our model by using the load\_model method

```
9
10
11 app = Flask(__name__, template_folder='templates') # initializing a flask app
12 # loading the model
13 model=load_model('nutrition.h5')
14 print("loaded model from disk")
15
16 @app.route('/') # route to display the home page
17 def home():
18     return render_template("home.html") # rendering the home page
19
20 @app.route('/image', methods=['GET', 'POST']) # routes to the index html
21 def image():
22     return render_template("image.html")
23
24 @app.route('/predict', methods=['GET', 'POST']) # route to show the predictions in a web ui
25 def launch():
26     if request.method == 'POST':
27         f=request.files['file'] #requesting the file
28         basepath=os.path.dirname(__file__) #storing the file directory
29         filepath=os.path.join(basepath, "uploads", f.filename) #storing the file in uploads folder
30         f.save(filepath) #saving the file
31
32
33         img=image.load_img(filepath, target_size=(64,64)) #load and reshaping the image
34         x=image.img_to_array(img) #converting image to an array
35         x=np.expand_dims(x, axis=0) #changing the dimensions of the image
```

## CREATING HTML PAGES AND ROUTING









## Routing To The Html Page

Here, the declared constructor is used to route to the HTML page created earlier.

In the above example, the ‘/’ URL is bound with the home.html function. Hence, when the home page of the webserver is opened in the browser, the HTML page is rendered. Whenever you enter the values from the HTML page the values can be retrieved using the POST Method.

Here, “home.html” is rendered when the home button is clicked on the UI

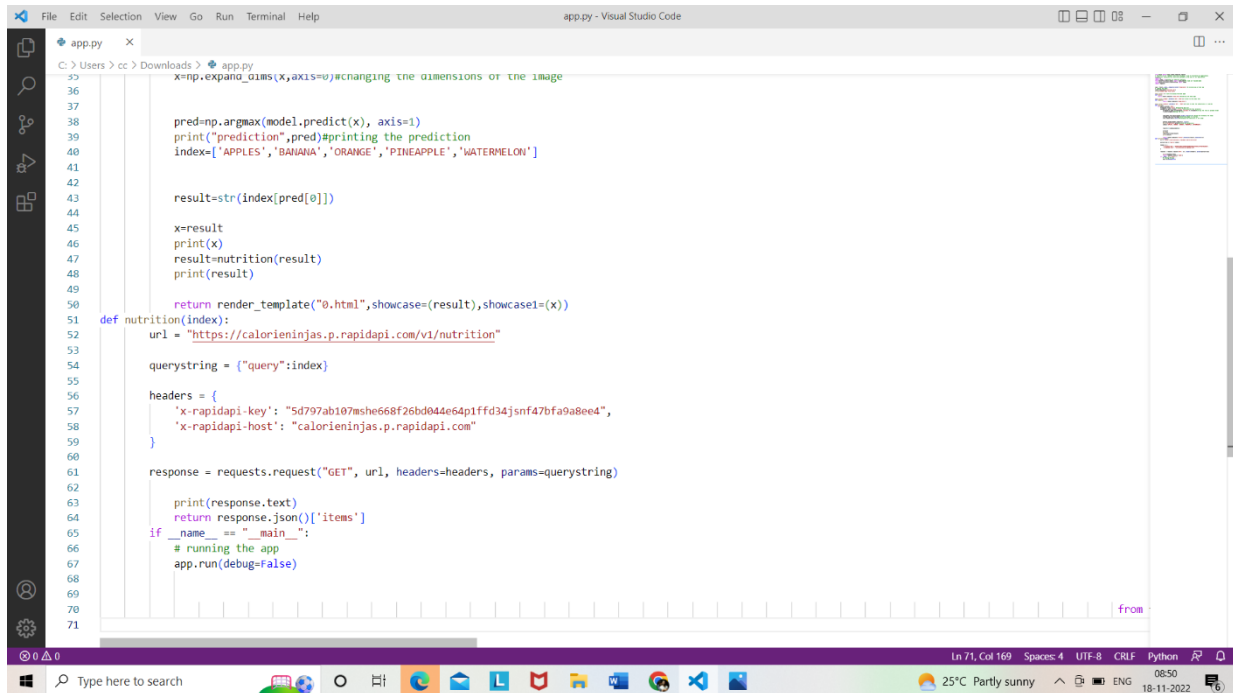
```
15
16 @app.route('/')# route to display the home page
17 def home():
18     return render_template('home.html')#rendering the home page
19
20 @app.route('/image1',methods=['GET','POST'])# routes to the index html
21 def image1():
22     return render_template("image.html")
23
```

When “image is uploaded “on the UI, the launch function is executed

```
23
24 @app.route('/predict',methods=['GET', 'POST'])# route to show the predictions in a web UI
25 def launch():
26     if request.method=='POST':
27         f=request.files['file'] #requesting the file
28         basepath=os.path.dirname('_file_')#storing the file directory
29         filepath=os.path.join(basepath,"uploads",f.filename)#storing the file in uploads folder
```

It will take the image request and we will be storing that image in our local system then we will convert the image into our required size and finally, we will be predicting the results with the help of our model which we trained and depending upon the class identified we will showcase the class name and its properties by rendering the respective html pages.





```
35 x=np.expand_dims(x,axis=0)#changing the dimensions of the image
36
37
38 pred=np.argmax(model.predict(x), axis=1)
39 print("prediction",pred)#printing the prediction
40 index=['APPLES','BANANA','ORANGE','PINEAPPLE','WATERMELON']
41
42
43 result=str(index[pred[0]])
44
45 x=result
46 print(x)
47 result=nutrition(result)
48 print(result)
49
50 return render_template("0.html",showcase=(result),showcase1=(x))
51
52 def nutrition(index):
53     url = "https://calorieninjas.p.rapidapi.com/v1/nutrition"
54
55     querystring = {"query":index}
56
57     headers = {
58         'x-rapidapi-key': "5d797ab107mshe668f26bd044e64p1ffd34jsnf47bfa9a8ee4",
59         'x-rapidapi-host': "calorieninjas.p.rapidapi.com"
60     }
61
62     response = requests.request("GET", url, headers=headers, params=querystring)
63
64     print(response.text)
65     return response.json()['items']
66
67 if __name__ == "__main__":
68     # running the app
69     app.run(debug=False)
70
71
```

Finally, Run the application

This is used to run the application in a localhost. The local host runs on port number 5000.(We can give different port numbers)



```
65 if __name__ == "__main__":
66     # running the app
67     app.run(debug=False)
68
69
70
71
```

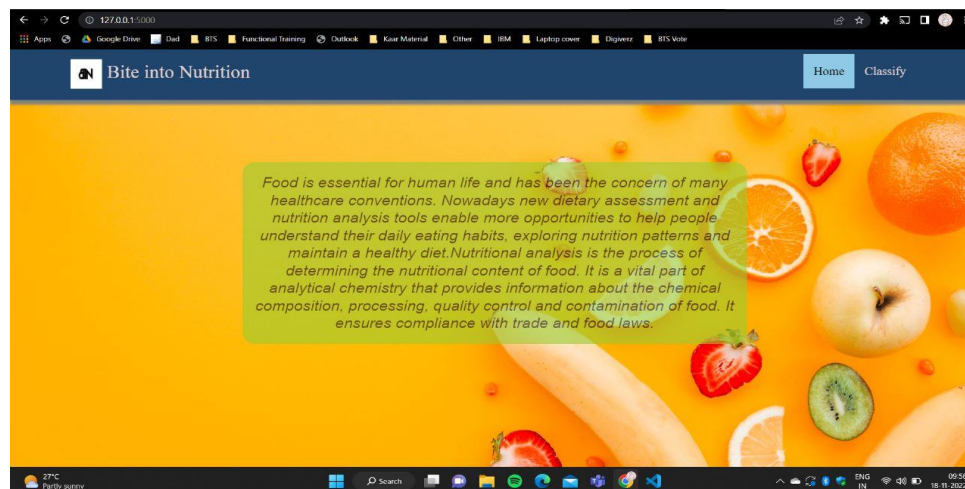
## RUN THE APPLICATION

- In Visual Studio open the integrated terminal.
- Now type the “python app.py” command.
- It will show the local host where your app is running on

<http://127.0.0.1:5000/>

- Ctr+Click the URL. It does navigate to where you can view your web page.
- Enter the values, click on the predict button and see the result/prediction on the web page.
- Then it will run on localhost:5000

```
PS D:\IBM Proj\Nutrition Analyzer> python app.py
2022-11-18 09:30:57.921283: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'cudart64_110.dll'; dlderror: cudart64_110.dll not found
2022-11-18 09:30:57.922769: I tensorflow/stream_executor/cuda/cudart_stub.cc:29] Ignore above cudart dlerror if you do not have a GPU set up on your machine.
2022-11-18 09:31:01.134670: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'nvcuda.dll'; dlderror: nvcuda.dll not found
2022-11-18 09:31:01.135439: W tensorflow/stream_executor/cuda/cuda_driver.cc:326] failed call to cuInit: UNKNOWN ERROR (303)
2022-11-18 09:31:01.142146: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:169] retrieving CUDA diagnostic information for host: Ishwarya
2022-11-18 09:31:01.143014: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:176] hostname: Ishwarya
2022-11-18 09:31:01.144128: I tensorflow/core/platform/cpu_feature_guard.cc:142] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical operations: AVX AVX2
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
Loaded model from disk
* Serving Flask app 'app'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
127.0.0.1 - - [18/Nov/2022 09:31:06] "GET / HTTP/1.1" 200 -
```

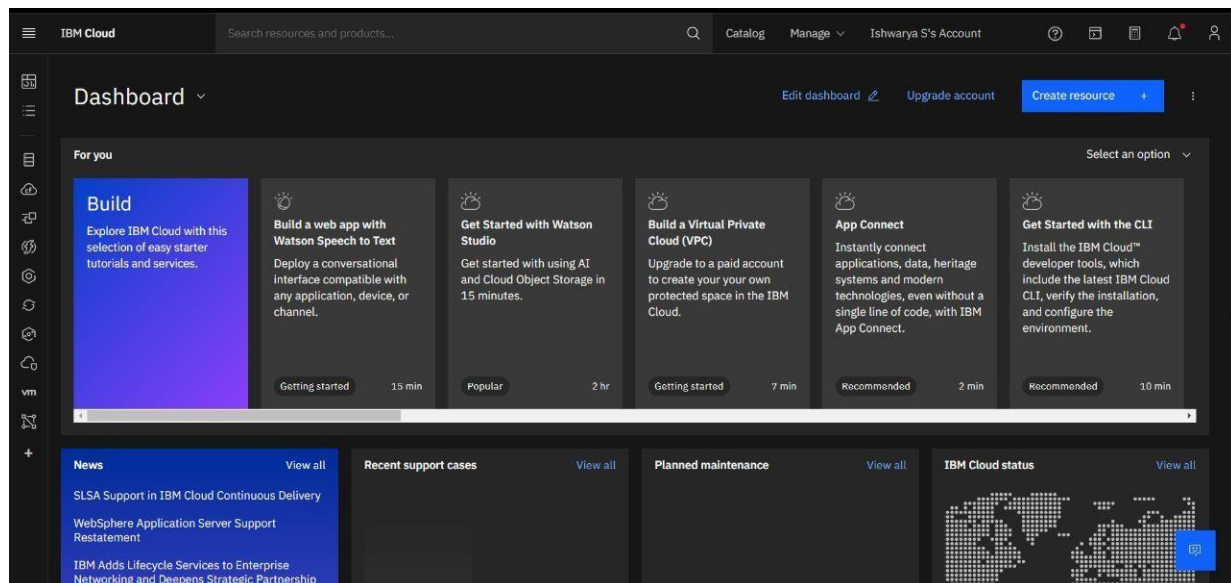


# Register For IBM Cloud

IBM Account :

Register link : <https://www.ibm.com/academic/home>

Log in : <https://cloud.ibm.com/>



## Train Model On IBM

Login to ibm cloud and create ibm Watson studio to train the model on ibm. Copy the file, paste it and run the file in notebook.

IBM Watson Studio interface showing a Jupyter Notebook titled "Image Preprocessing".

### Image Preprocessing

**Import The ImageDataGenerator Library**

```
In [ ]: from keras.preprocessing.image import ImageDataGenerator
```

**Configure ImageDataGenerator Class**

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

In [ ]: test_datagen = ImageDataGenerator(rescale=1./255)
```

**Apply Image DataGenerator Functionality To Trainset And Testset**

```
In [ ]: x_train = train_datagen.flow_from_directory(Data_trainpath, target_size=(64,64), batch_size=5, color_mode='rgb', class_mode='sparse')

In [ ]: x_test = train_datagen.flow_from_directory(Data_testpath, target_size=(64,64), batch_size=5, color_mode='rgb', class_mode='sparse')

In [ ]: print(x_train.class_indices)

In [ ]: print(x_test.class_indices)

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

The interface includes a top navigation bar with tabs for "Inbox (279)", "IBM", "IBM-Project-24498-1659943754", "Service Details - IBM Cloud", and "Model\_Building - IBM Watson S...". The main toolbar shows "File", "Edit", "View", "Insert", "Cell", "Kernel", and "Help". The bottom status bar displays "Type here to search", system icons, and the date "18-11-2022".

IBM Watson Studio interface showing a Jupyter Notebook titled "Model Building". The notebook is running on a Python 3.9 kernel. The code in the notebook includes:

```
In [ ]: x_train = train_datagen.flow_from_directory(Data_trainpath, target_size=(64,64), batch_size=5, color_mode='rgb', class_mode='sparse')
In [ ]: x_test = train_datagen.flow_from_directory(Data_testpath, target_size=(64,64), batch_size=5, color_mode='rgb', class_mode='sparse')
In [ ]: print(x_train.class_indices)
In [ ]: print(x_test.class_indices)
In [ ]: from collections import Counter as c
In [ ]: c(x_train.labels)
```

### Model Building

#### Importing The Model Building Libraries

```
In [7]: import numpy as np
import tensorflow
from tensorflow.keras.models import Sequential
from tensorflow.keras import layers
from tensorflow.keras.layers import Dense, Flatten
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Dropout
from keras.preprocessing.image import ImageDataGenerator
```

#### Initializing The Model

Continuation of the Jupyter Notebook "Model Building". The code in the notebook includes:

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

### Train The Model

```
In [ ]: model.fit_generator(
    generator=x_train, steps_per_epoch = len(x_train),
    epochs=15, validation_data=x_test, validation_steps = len(x_test))
```

### Save The Model

```
In [ ]: model.save('nutrition.h5')
```

### Test The Model

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

In [ ]: from tensorflow.keras.utils import img_to_array

In [ ]: img = tensorflow.keras.utils.load_img("/content/drive/MyDrive/Nutrition Image Analysis using CNN and Rapid API/Nutrition Analysis Using Image Classification/Flask/Sample_Images")
x = img_to_array(img)
x = np.expand_dims(x, axis = 0)
pred = np.argmax(final_model.predict(x), axis=1)
pred
```



## 8. TESTING

### 8.1 TEST CASES

A **test case** is a singular set of actions or instructions for a tester to perform that validates a specific aspect of a product or application functionality. **If the test fails, the result might be a software defect that the organization can triage.**

**1.Create test cases that are as simple as possible.**

**2.Create Test Case with End User in Mind**

**3.Do not repeat test cases**

#### ☐ Test The Model

- ☐ Evaluation is a process during the development of the model to check whether the model is the best fit for the given problem and corresponding data.
- ☐ Load the saved model using `load_model`
- ☐ Taking an image as input and checking the results
- ☐ By using the model we are predicting the output for the given input image
- ☐ The predicted class index name will be printed here.

```
Test The Model

[22] from tensorflow.keras.models import load_model
     from keras.preprocessing import image
     final_model = load_model("nutrition.h5")

[23] from tensorflow.keras.utils import img_to_array


img = tensorflow.keras.utils.load_img("/content/drive/MyDrive/Nutrition Image Analysis using CNN and Rapid API/Nutrition Analysis Using Image Classification/Flask/Sample Images/Test_Images4.jpg", grayscale=False)
x = img_to_array(img)
x = np.expand_dims(x, axis=0)
pred = np.argmax(final_model.predict(x), axis=-1)
pred

1/1 [=====] - 0s 103ms/step
array([2])

[25] index=['APPLES', 'BANANA', 'ORANGE']
     result=index[pred[0]]
     result

'ORANGE'
```

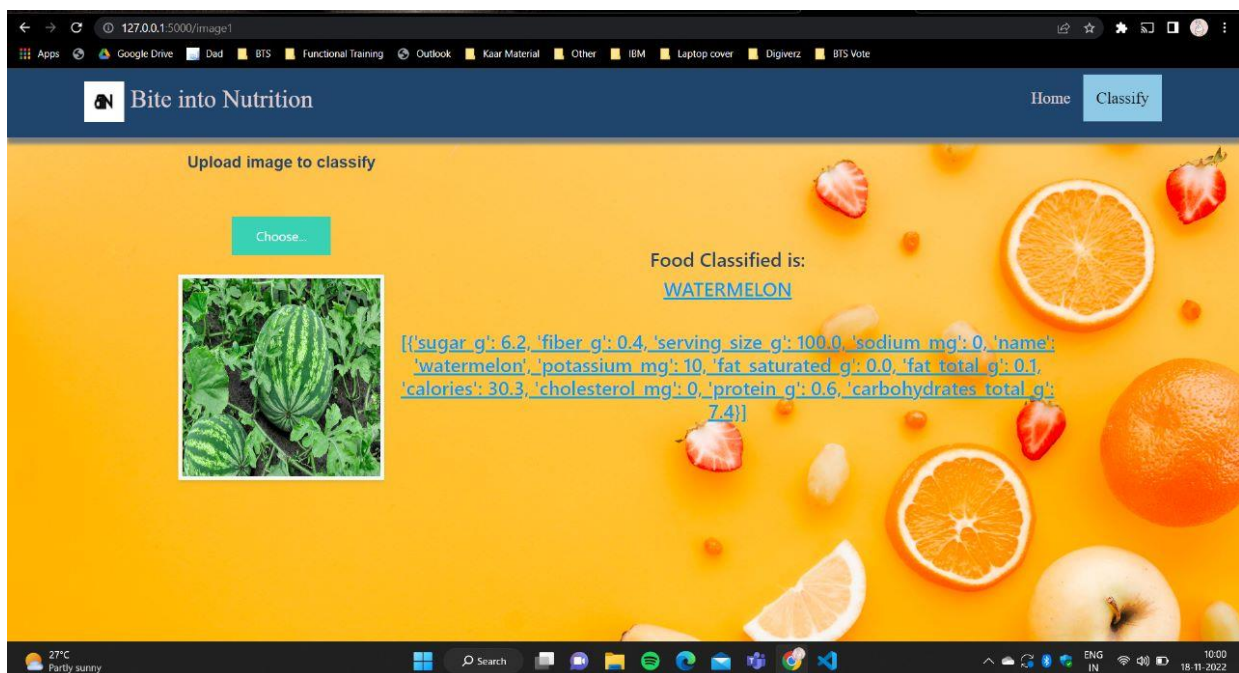
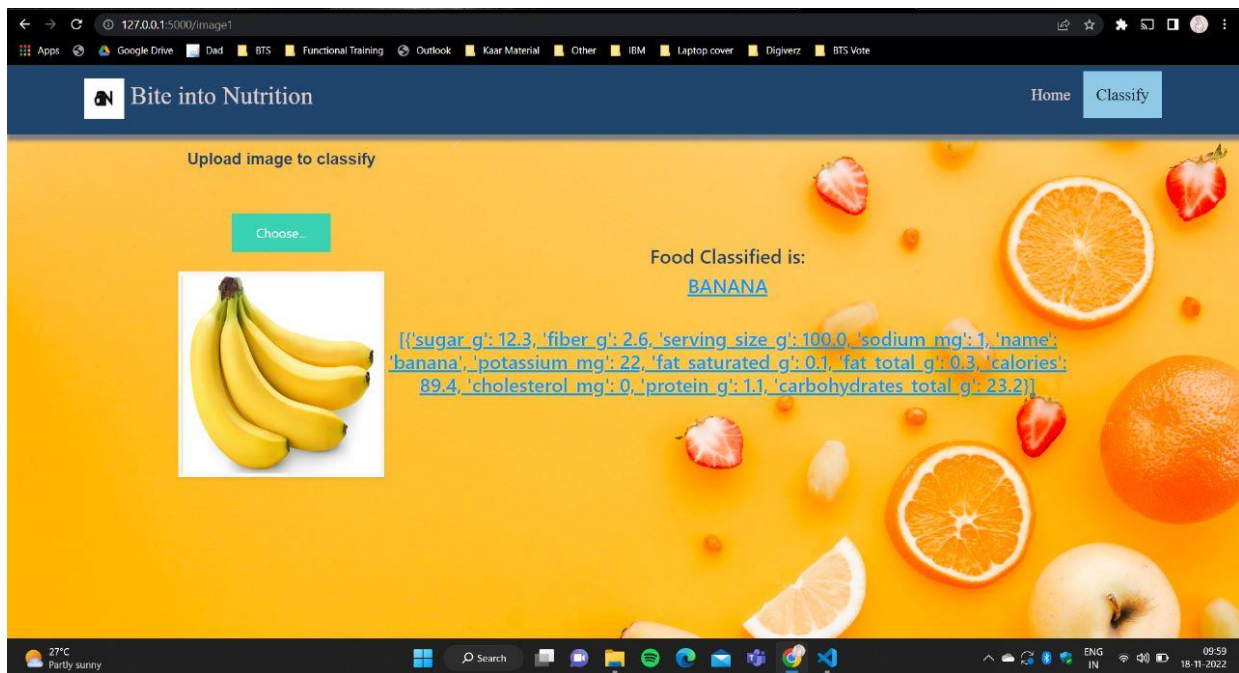
img



### 8.2 USER ACCEPTANCE TESTING



User Acceptance Testing (UAT), which is performed on most UIT projects, sometimes called beta testing or end-user testing, is a **phase of software development in which the software is tested in the "real world" by the intended audience or business representative.**



## **9. RESULTS**

### **9.1 PERFORMANCE METRICS**

**Performance measurement is the process of collecting, analyzing and/or reporting information regarding the performance of an individual, group, organization, system or component . Definitions of performance measurement tend to be predicated upon an assumption about why the performance is being measured .**

**Permonfance metric of this nutrition analysis using artificial intelligence is good . They perform in a good level.**

## **10. ADVANTAGES AND DISADVANTAGES**

### **ADVANTAGES:**

**It maintains a healthy lifestyle for a human . It calculates the BMI (Body mass index) . It regulates our body blood pressure level. It controls the weight of the human body.**

**Because of todays lifestyle people tend to eat more junk which causes serious health related disorders such as diabetes, blood pressure obesity. The can be prevented if we a track on our nutrition intake level.**

### **DISADVANTAGES:**

**Our health report is completely exposed to others . There may be security level problem in this application. Because hackers can easily track our health status if they break all the security barriers in the application. They may even use to kill a person if needed.**

**Security of data is a concern in this type of application.**

## 11. CONCLUSION

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.

Our project plays a vital role in maintain the humans healthy lifestyle by analysing their nutrition intake . And keeps a track on their health status in a very serious manner. It helps in avoiding serious health diseases. So that a persons life can be saved.

## **12. FUTURE SCOPE**

In future or upcoming days many people may use this app for their personal assistance of nutrition intake and to maintain a healthy lifestyle. This application has a major scope in future. We may include even pulse rate , heart rhythm , kidney functionality in the near future to our application

## **13. APPENDIX**

**Source code is available in the below mentioned github link :**

**<https://github.com/IBM-EPBL/IBM-Project-24498-1659943754/tree/main/Final%20Deliverables>**