Nutritional Value Calculator

By

Aqsa Kanwal

Haider Khan

Husnain Raza

Usman Ali Khan

Supervised by:

Dr. Hammad Afzal

Submitted to the Department of Computer Software Engineering,
Military College of Signals, National University of Sciences and Technology, Islamabad,
in partial fulfillment for the requirements of B.E Degree in Software Engineering.

June 2022

In the name of ALLAH, the Most benevolent, the Most Courteous

CERTIFICATE OF CORRECTNESS AND APPROVAL

This is to officially state that the thesis work contained in this report

"Nutritional Value Calculator"

is carried out by

Aqsa Kanwal, Haider Khan, Husnain Raza, and Usman Ali Khan

under my supervision and that in my judgement, it is fully ample, in scope and excellence, for the degree of Bachelor of Software Engineering in Military College of Signals, National University of Sciences and Technology (NUST), Islamabad.

Approved by

Supervisor

Date: _____

DECLARATION OF ORIGINALITY

We hereby declare that no portion of work presented in this thesis has been submitted in support of another award or qualification in either this institute or anywhere else.

ACKNOWLEDGEMENTS

Allah Subhan'Wa'Tala is the sole guidance in all domains.

Our parents, colleagues and most of all supervisors, Dr. Hammad Afzal and Lt Col Khawir Mahmood.

The group members, who through all adversities worked steadfastly.

Plagiarism Certificate (Turnitin Report)

This thesis has 7% similarity index. Turnitin report endorsed by Supervisor is attached.

_____

Aqsa Kanwal
00000280966

_____

Haider Khan
00000248078

_____

Husnain Raza
00000256180

_____

Usman Ali Khan
00000221958

_____

Signature of Supervisor

ABSTRACT

Over the past few years, obesity has become a global health concern and the situation is only getting worse with every passing day. As of 2022, there are 204,555 fast food businesses in USA, witnessing a 1.3% increase as compared to the statistics of 2021. This increase in the number of restaurants is directly proportional to the increase in fast-food consumption and the number of diet-related illnesses as well. Spreading awareness regarding the matter by running campaigns and educating the public has made a huge impact but still, the situation is not under control. As we are living in the age of technological advancements, we should shift our focus towards modern approaches; diet planner and calorie counter applications are the best choice we have right now. To implement this concept, we have come up with web and android applications of "Nutritional Value Calculator" which estimate the number of calories and nutrients in a meal using its image only. This task is easier to implement for western cuisine as compared to Pakistani cuisine mainly due to the abundance of datasets available online for western food [1]. As we intended to build this software for Pakistani users, we did some improvisation and created a custom dataset from scratch which contains images of some popular Pakistani meals.

Contents

1: Introduction

Overview

In addition to maintaining physical fitness through exercise, planning our diet is essential for living a healthy life. The following graph depicts the percentage of obesity among adults of different age groups and genders belonging to US from 2017 to 2018 [2]. The graph reveals alarming facts about deteriorating health of the general population and leaves a room for health professionals to imagine the underlying diseases these people are suffering from i.e., diabetes, kidney diseases, high cholesterol etc.

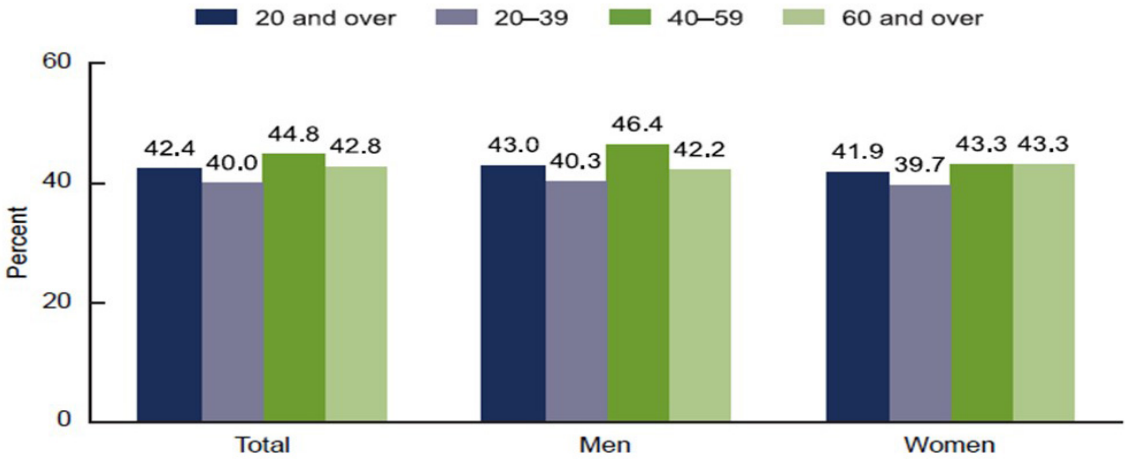


Figure 1: Percentage of obesity among adults of different age groups and gender

According to WHO's survey in 2020, approximately 149 million children under the age of five were estimated to be stunted, 45 million were estimated to be too thin for height, and 38.9 million were obese. [3]

Keeping in mind the figures discussed above, we believe that it's the right time to give people control over their own health by providing them assistance in calculating the nutritional value of

every meal they eat. Our goal is to reduce the complexity involved in this process and provide a convenient solution which requires minimum effort and skill.

Problem Statement

Pakistan is also vulnerable to the epidemic of obesity but unlike the western world, Pakistanis don't have access to the best medical facilities and professional guidance related to the subject matter. Around 40% of children in Pakistan suffer from obesity according to a recent analysis conducted by Pakistan Health Commission along with several hospital-based surveys [4].

The effect of malnutrition is also becoming apparent in Pakistan, with almost one in three children underweight alongside a high prevalence of overweight (9.5%) in the same age group. [5]. In that same survey, it was reported that in Pakistan, 14.4% children are undernourished. In NNS 2011, 28% children were reported to be overweight or obese, rising to 37.8% in 2018. [5]

The problem lies in the fact that we are unable to distinguish the heathy food items from the unhealthy ones because there is no simple formula to conclude the results. The nutritional value of packaged food items is very simple to calculate because the percentage of each nutrient and the number of calories mentioned on the packaging. For traditional meals like Biryani, Pulao, and Nihari, we need to do some complex mathematical calculation, find out the size of the meal, and list down every constituent ingredient to be able to come up with a rough estimation of the total calories. Obviously, this method can't be carried out before every meal and for an ordinary person with limited knowledge, doing this kind of calculation will be close to impossible.

Proposed Solution

Nutritional Value Calculator is deployed as web and mobile applications through which the users will be able to calculate the nutritional value of any meal/platter within a few seconds. First the

users will take picture of a meal simply with their phone's camera and upload it to the application. The application will process the picture, recognize every food item in it, predict each item's quantity and then calculate the overall nutritional value (proteins, carbohydrates, fats, etc.) along with the total number of calories.

Users will also be given the option to plan their diet by entering the total amount of calories and nutrients they would like to consume in a day and on this basis, every meal's data processed by the application will be stored in the database and the users will be able to track their daily calories intake.

A chatbot will also be introduced as a meal recommender by suggesting a healthy/balanced/fast-food meal according to the user's requirements. This functionality will be achieved through rapid scanning of the datasets and returning all the results that satisfy the user's requirements. The chatbot will also notify the user when he is exceeding/not fulfilling his daily nutrients intake set in the application.

Working Principle

The project works on the concept of supervised machine learning and is divided into three major phases of development: Model training, Web application, and Android application

Model Training

We have used a Python library known as "Detecto" which is built on top of PyTorch to train our model. The biggest advantage of using Detecto is that it only uses a few lines of code and offers the functionality to input an image to predict the labels associated with it.

Dataset training and processing:

The images and labels were divided into two categories: Train and Test. Both folders were uploaded to Google drive which was then mounted on Google Colab.

Google Colab was then given access to the drive and the paths to Train and Test folders were entered to start the training process.

```
[ ]  from google.colab import drive
     drive.mount('/content/drive')
```

```
Train_dataset=core.Dataset('/content/drive/MyDrive/Detecto-Updated Dataset/Detecto-UpdatedDataset/Train/',transform=custom_transforms)#L1
Test_dataset = core.Dataset('/content/drive/MyDrive/Detecto-Updated Dataset/Detecto-UpdatedDataset/Test/')#L2
loader=core.DataLoader(Train_dataset, batch_size=2, shuffle=True)#L3
```

As training a machine learning model is a resource-intensive process, GPU acceleration is enabled in Google Colab to speed up the process.

GUI presentation:

The visual demonstration of the project is done through the aid of GUI (graphical user interface) which is made using ReactJS, HTML, CSS, Bootstrap (in case of Web-App).

Objectives

General Objectives:

“To build an innovative state of the art software app assisted by machine learning algorithms and RESTful API to provide an easy and smart way to get to know your food.”

Academic Objectives:

- Development of a smart and intelligent Food Nutritional Value Calculator
- To implement Machine Learning techniques and simulate the results

- To increase productivity and coordination by working in a team
- To use Python-written algorithms in our web application using MERN stack.
- To design a project that contributes to the welfare of society
- To enhance the ability of researching the right material to learn from and to work on.

Scope

This project finds its scope wherever there is someone worried about the intake of his/her food. Be it a gym-doer, a dietician or a common person who wants to take care of his/her health. It is an innovative and easy-to-use software powered by machine learning algorithms and image processing techniques.

Deliverables

Trained Model

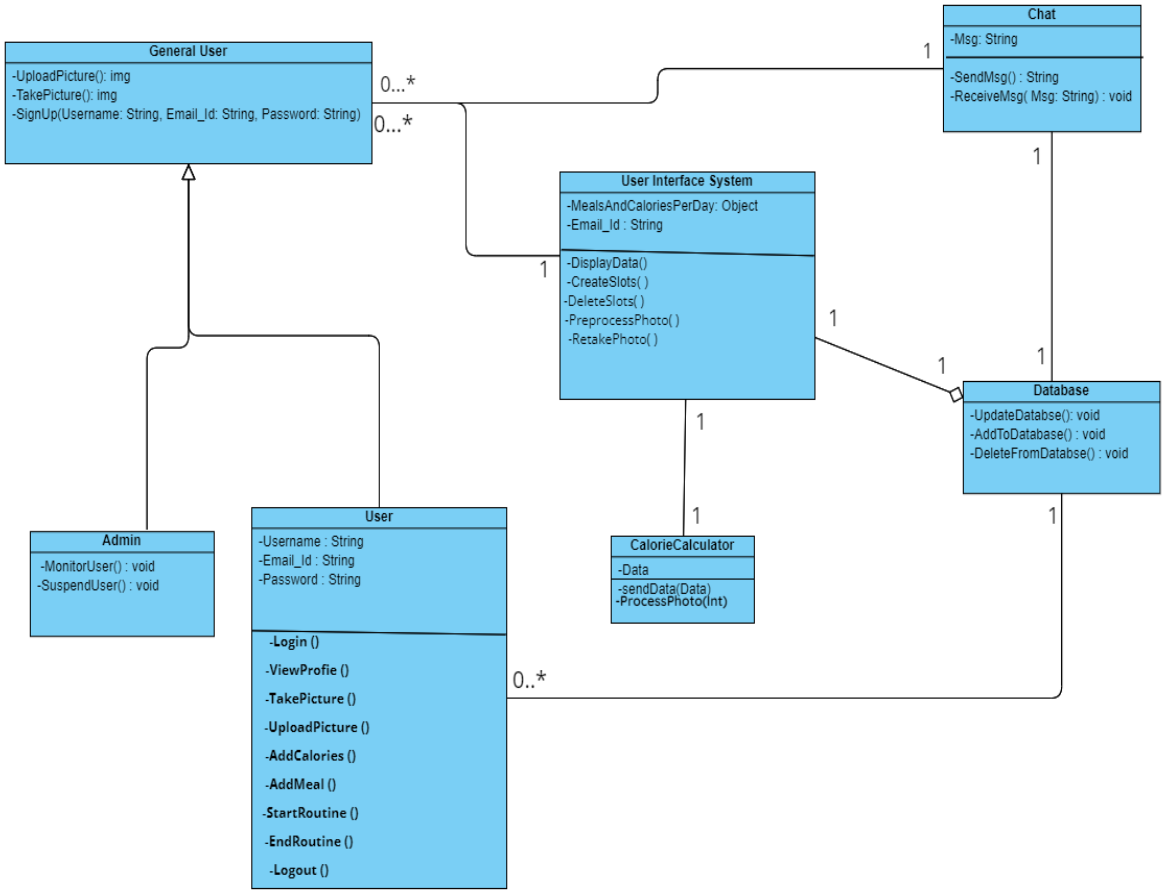
The foundation for this project is the trained model which is incorporated in the web and android applications for predicting the nutritional value of different meals.

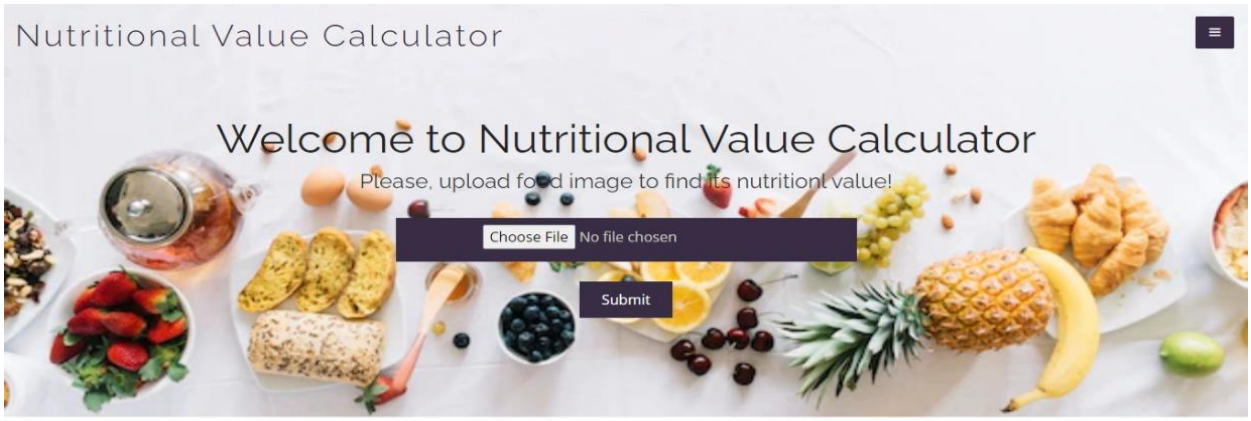
Web Application

It provides an easy-to-use interface for the users which is developed using ReactJS, HTML, CSS, and Bootstrap.

- The landing page of the website will display login form along with the option to sign up.
- There will be two buttons named “Upload a Picture” and “Take a Picture” present underneath the login form.

- There will be an arrow pointing downwards to the bar with a message written on it “Need Help?”, stick to the bottom-right of the screen.
- On clicking the registration button, user will be taken to another page to fill the registration form.
- After registering or logging into the account, the user will be taken to a page where the ‘User Dashboard” will be displayed.
- User Dashboard will show the type and amount of nutrients the user had each day of the last week.
- Underneath the dashboard will be a button “Start Your Routine”. For a new user, only a button will be displayed “Start Your Routine”.
- After clicking the button, the fields “Number of Calories”, “No. of Meals in a day” and “Upload a Picture” button will be displayed.



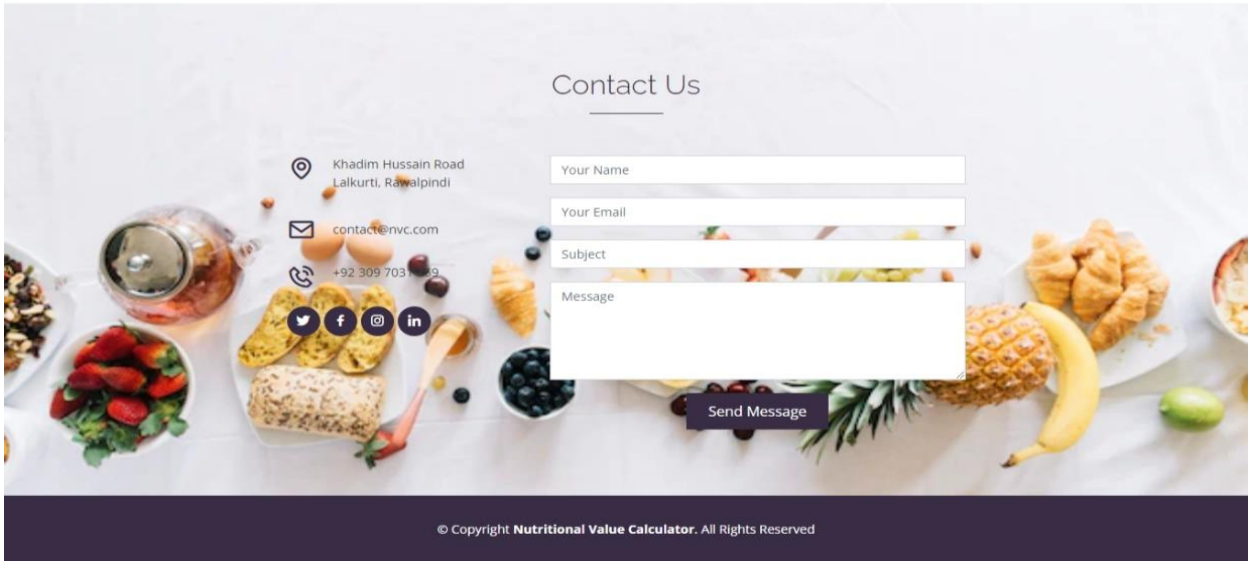


In today's world, Nutrition is a crucial part yet it's ignored.

Nutritional Vlue Calculator helps you to track your health more effeciently. It helps you:

- Check nutrition facts of a meal.
- Set daily routines and follow them to stay healthy.
- Get recommendations of meals and customize your diet plan.

Now you can chat in real-time and get help to make a diet plan that is more suitable for you without visiting expensive nutritionists or purchasing awfully expensive diet plans. So Try it now !!



Android Application

- For the mobile application, the interface will be same, but it will be scaled down to adjust to the mobile screen.

- For different screen sizes, the application's interface will be scaled accordingly.
- As the mobile screen has limited size, there will be a scrolling option available, but users will not be allowed to zoom in or zoom out.
- By default, the interface will be zoomed enough for the users to read text and view all the options from 1.5 feet distance.

Relevant Sustainable Development Goals

What is the Locally Relevant Socio-Economic Issue that the Project Addresses?

Becoming more conscious about diet, calorie intake, and predicting the nutrition facts of food has become a need of our society. People usually pay less attention to nutrition due to less awareness or high consultation fees of nutritionists and diet planners causing more harm to one's health and well-being.

Justify how particular SDG is related to your FYP?

Our FYP is based on the SDG "**Good Health and Well-being**". The aim of this project is to make people more aware of what they eat and whether it contributes to their overall health or plays a part in making them vulnerable to diseases and illnesses. Our project will allow the users to make customized diet plans, track daily progress based on nutrients intake, and set a personalized routine by selecting a suitable number of calories for each day.

Structure of Thesis

Chapter 2: literature review and the background and analysis study this thesis is based upon.

Chapter 3: Design and development of the project.

Chapter 4: Detailed evaluation and analysis of the code.

Chapter 5: Conclusion of the project.

Chapter 6: Future work needed to be done for the commercialization of this project.

2: Literature Review

A new product is launched by modifying and enhancing the features of previously launched similar products. Literature review is an important step for development of an idea to a new product. Likewise, for the development of a product, and for its replacement, related to food nutritional value calculator system, a detailed study regarding all similar projects is compulsory. Our research is divided into the following points.

- Industrial Background
- Existing solutions and their drawbacks
- Research Papers

Industrial background

In the last two decades, Pakistan has witnessed an increase in the trend of eating fast food which has given birth to many innovative ideas like food delivery at customer's doorstep. This kind of convenience has received a lot of praise from the customers as they don't need to step into the kitchen and cook anymore; their breakfast, lunch, and dinner are just one click away. With hundreds of restaurants and thousands of meals to choose from, the users will never run out of options which means that their health will never stop deteriorating. The same pattern is observed globally so finally some researchers with expertise in web development and machine learning have stepped in to propose good solutions.

Existing solutions and their drawbacks

Different solutions are already being provided to deal with food-related ailments and to track the nutrients, but every product has some pros and cons. Following are some applications which are available online:

Macros – Calorie Counter & Meal Planner

This system is available as a mobile application. It simply tracks the calories intake and plans meals accordingly. User enters the name of food item, and the application displays its nutritional value. The drawback of this system is that it cannot predict the nutritional value of a meal because machine learning algorithms haven't been used in its development.

RecipeIQ: Recipe Organizer & Nutrition Calculator

This nutrition calculator app is meant for chefs and foodies. User can take the photo of any recipe to calculate nutrition facts and can also group different recipes to check their total nutritional value. User can also create custom recipes by editing ingredients to taste. However, it is not beneficial for the users who want to check the nutritional value of the meal instantly.

Food Calculator: Calories, Protein, Carbs, Fat

This app simply stores precalculated nutritional values of different food items and recipes. Neither it adds up the calories of different meals fused together, nor it allows users to make customized meal plans and get meal recommendations.

Research Papers

"A Framework to Estimate the Nutritional Value of Food in Real Time Using Deep Learning Techniques" [6] has been used as a reference to create this project.

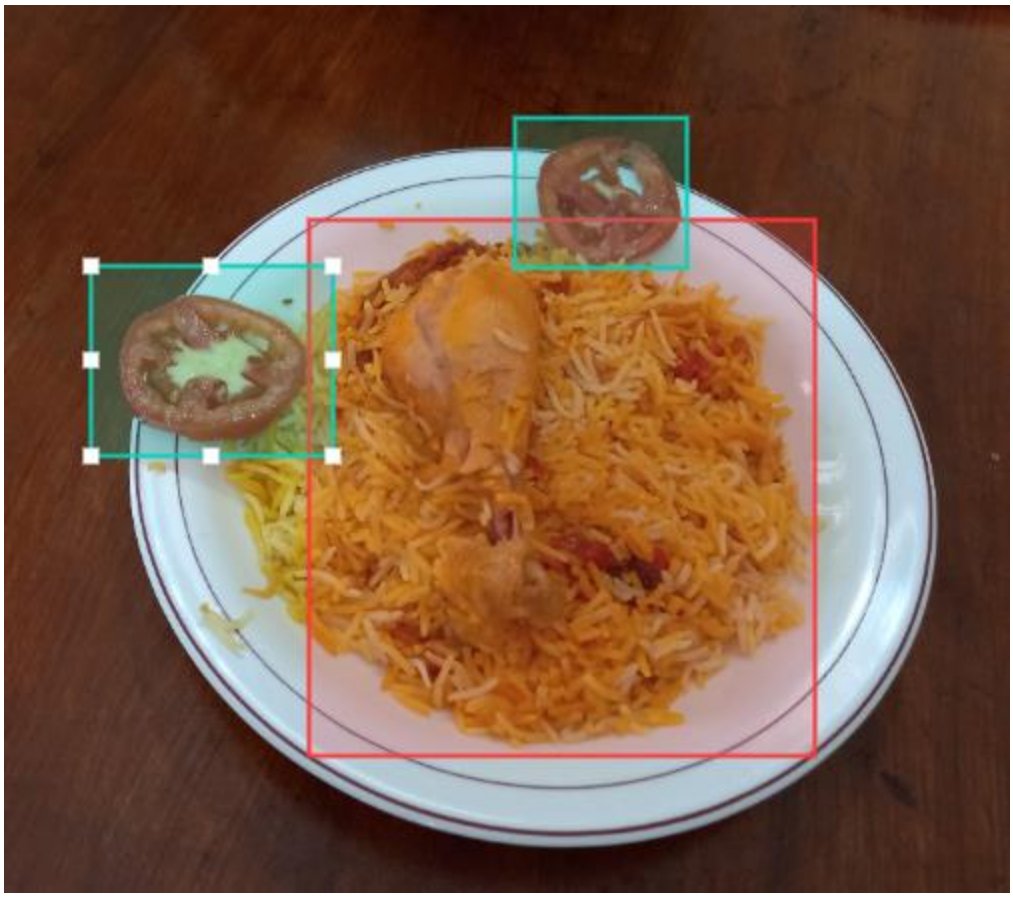
3: Interfacing and Detection

Food Item Detection

Pakistani cuisine is very diverse; curry, gravy, rice, meat, and lentils are combined to make hundreds of unique dishes. If we had included all these dishes, creating the dataset alone would have taken months to complete and the amount of memory occupied by the images would have been in terabytes. We decided to include only those meals which were unique and accurately represented a specific meal category. To represent a combination of rice and meat, we included chicken biryani and following the same pattern, we added Kofta curry, steamed chicken etc. Pakistani cuisine is incomplete without salad, so we included tomatoes, cucumbers, and green chilies to represent this part of our meal.

Preparing Dataset

The most important component of this project is the dataset comprising of images of different food items. All the images have been captured through a phone's camera and then labelled using an online tool "Makesense.ai". For each image, bounding boxes are drawn according to the labels identified in the image. The labels have been downloaded in VOC XML format which is a requirement for the " **detecto**" library.



The model's accuracy is directly proportional to the size of dataset and degree of variety in it. To fulfil these conditions, we captured at least one hundred training images and twenty test images for each food item. Object detection models yield better results if high-quality images are used in the dataset, so we captured all the photos using a 48-megapixel camera. Every image has a balanced contrast, high level of detail, and optimal focus which played a great role in minimizing loss and maximizing accuracy during the training process.

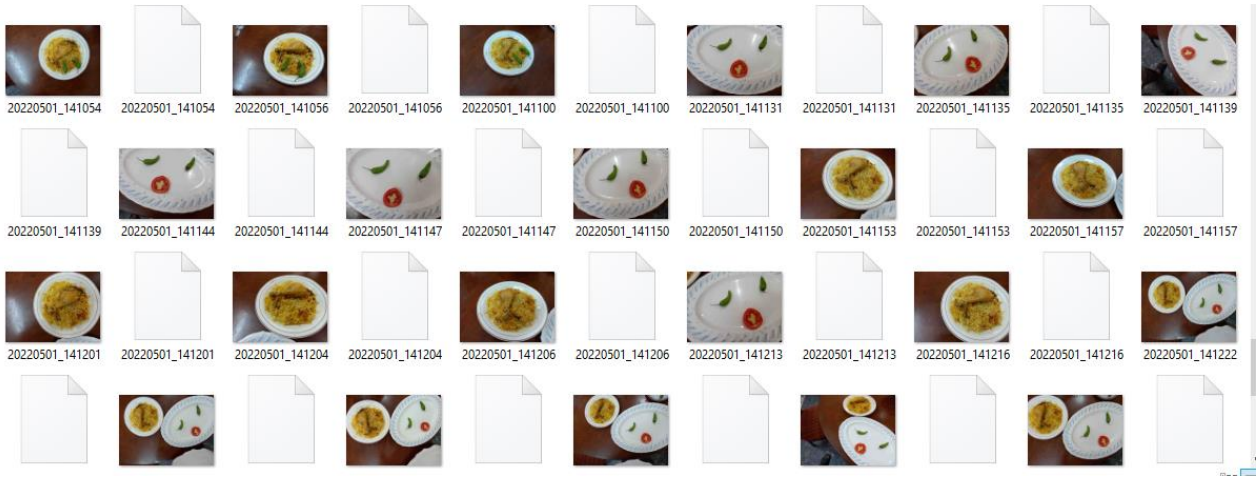


Figure 2: Proposed Dataset

Dataset Classes

Class	Number of Images
Chicken Biryani	120
Kofta Curry	110
Steamed Chicken	280
Cucumber	240
Tomatoes	
Green Chili	
Samosa	240
Total	990

Faster R-CNN Model

Faster R-CNN model [7] is a convolutional neural network model developed by researchers at Microsoft. It employs a region proposal network (RPN) to generate region proposals, which is timesaving as compared to algorithms like Selective Search.

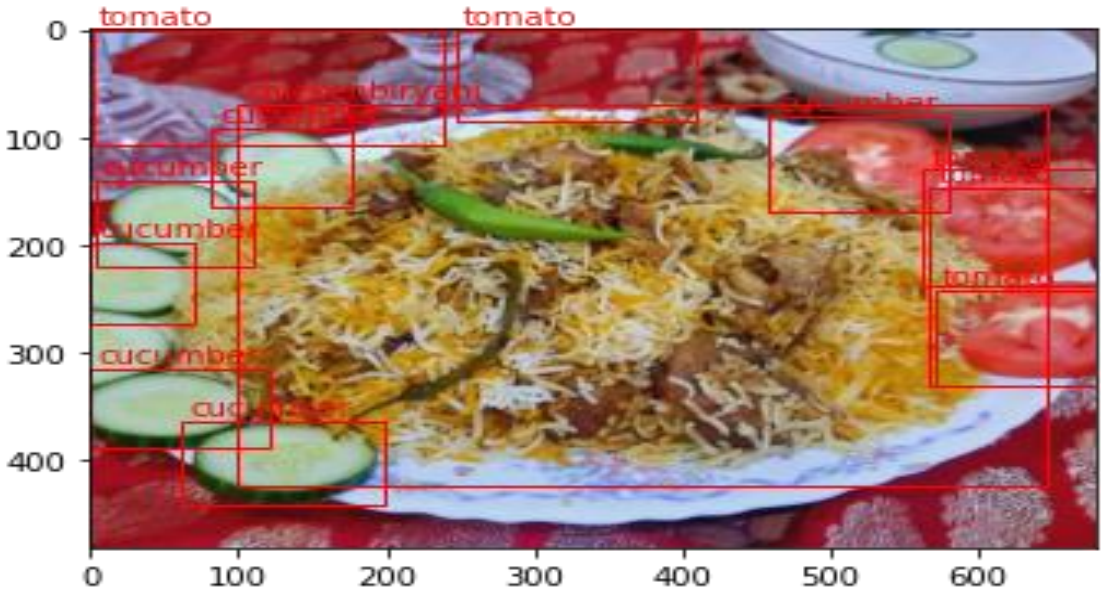
Once the model got trained, we saved '**model_weights.pth**' file which can be loaded afterwards to predict labels without having to train the model all over again. To check the model's accuracy, we downloaded a random image of Biryani from Google and ran the code.

```
model = torch.load('/content/drive/MyDrive/Detecto/model_weights.pth')
```

Input Image:

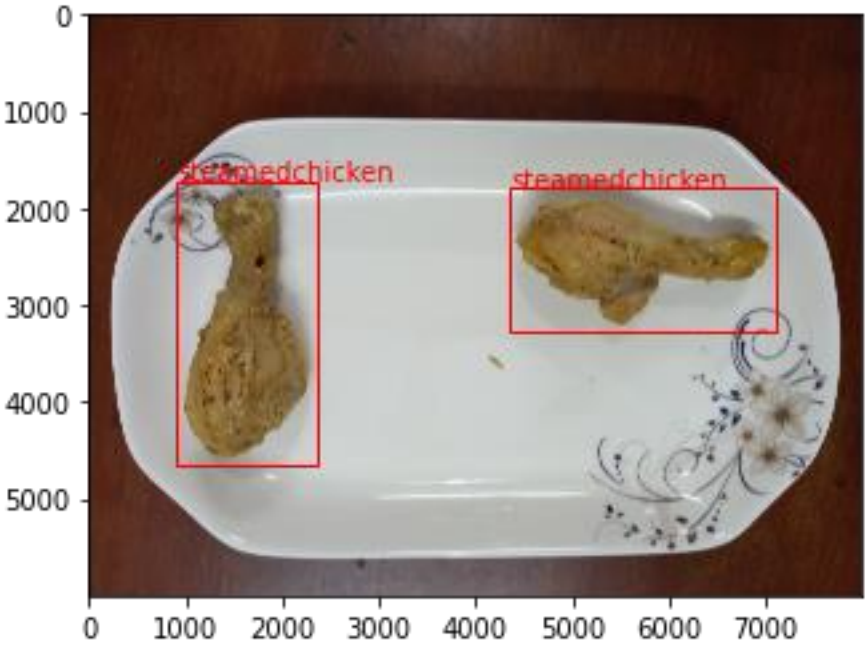
Output:

The output shows that biryani, cucumbers, and tomatoes have been detected by the program which is evident by the bounding boxes around them with correct labels.



Input Image:

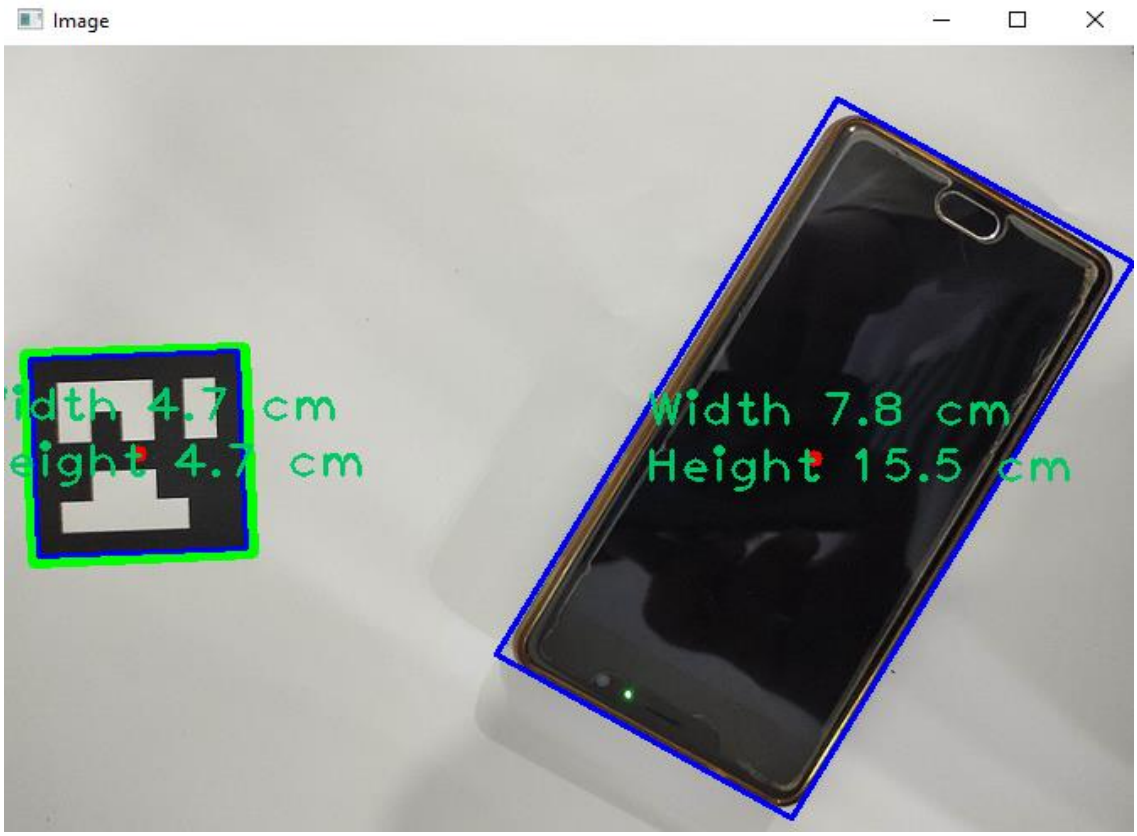
Output:



Predicting the object's size

The estimation of the meal's size is crucial for predicting its nutritional value because the user will eat a variety of meals having different quantities. We have used a computer vision library called "OpenCV" [8] to estimate the object's size.

We generated a 5cm x 5cm Aruco marker using an online tool and printed it on an A4 plain paper. Using this marker as a reference object with known dimensions, we estimated the size of different objects to check the accuracy of our results.



4: Code Analysis and Evaluation

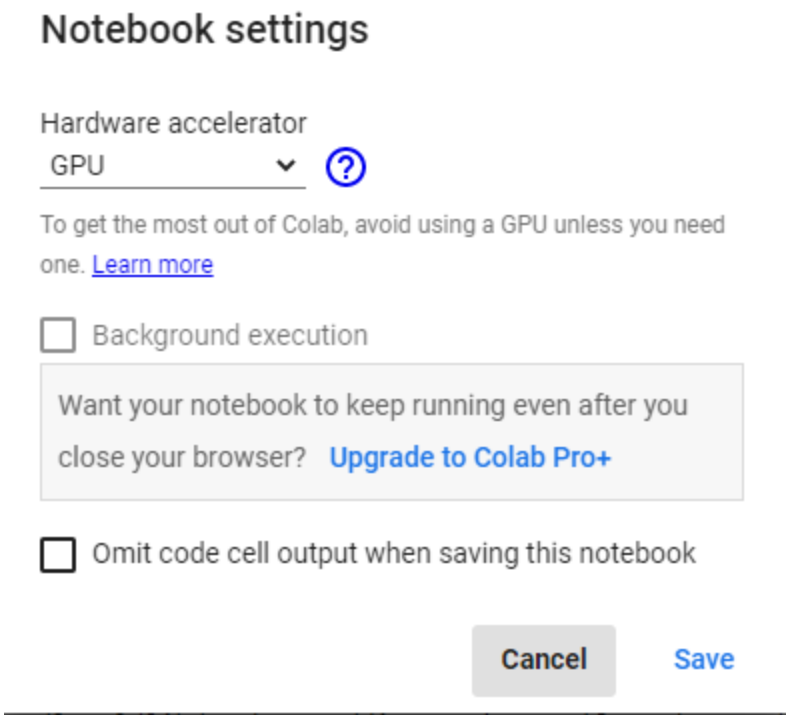
Object Detection

Before loading the dataset from Google drive, it is important to install all the required dependencies and libraries essential for training the model.

```
import torch
print(torch.cuda.is_available())
```

```
True
```

Executing the block of code shown above will return the output = “True” if your runtime type is set as **GPU**. If the output is false, it means that we need to change our runtime type to **GPU** from the Notebook settings.



After completing these steps, we will install "**Detecto**" library which is the most essential component of this project.

```
[ ]  !pip install detecto
```

Now, we need to import packages/libraries required for plotting the losses occurred during model training, detecting objects, and performing mathematical operations.

```
[ ]  from detecto import core, utils, visualize
     from detecto.visualize import show_labeled_image, plot_prediction_grid
     from torchvision import transforms
     import matplotlib.pyplot as plt
     import numpy as np
```

To introduce uniqueness in our dataset for better results, we will transform our images dataset by performing horizontal/vertical flip, resizing, and changing saturation level.

```
custom_transforms = transforms.Compose([
transforms.ToPILImage(),
transforms.Resize(900),
transforms.RandomHorizontalFlip(0.5),
transforms.ColorJitter(saturation=0.2),
transforms.ToTensor(),
utils.normalize_transform(),
])
```

Now, we have reached the main part of our code where the model will be trained using the dataset stored in Google drive. We have the option to adjust the hyperparameters (Learning rate and the number of epochs) to achieve the best results.

```
Train_dataset=core.Dataset('/content/drive/MyDrive/Detecto-Updated Dataset/Detecto-UpdatedDataset/Train/',transform=custom_transforms)#L1
Test_dataset = core.Dataset('/content/drive/MyDrive/Detecto-Updated Dataset/Detecto-UpdatedDataset/Test/')#L2
loader=core.DataLoader(Train_dataset, batch_size=2, shuffle=True)#L3
```

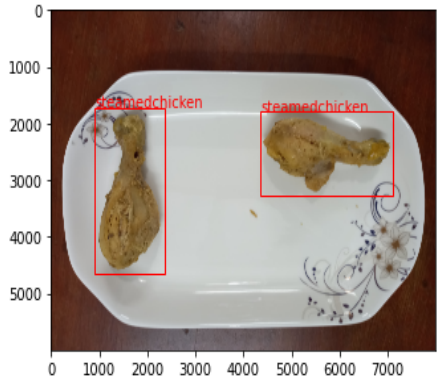
```
losses = model.fit(loader, Test_dataset, epochs=15, lr_step_size=5, learning_rate=0.001, verbose=True)#L5
```

Once the training process is done, we need to save the **mode_weights.pth** file which we can simply load afterwards to predict labels from an image.

```
model.save('model_weights.pth')
model = core.Model.load('model_weights.pth', ['chickenbiryani','tomato',
'greenchilli','cucumber','kofta','steamedchicken'])
```

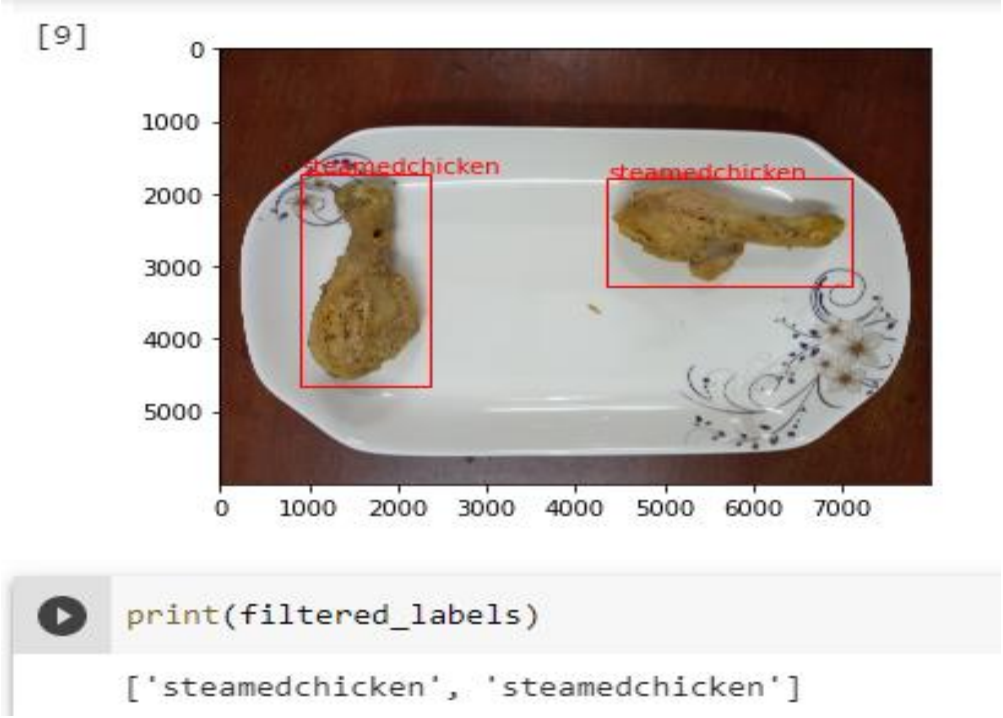
After saving the weights, we can enter the path of any random image either downloaded from Google or belonging to the **Test** folder of our dataset to check the accuracy of our model

```
[ ]  image = utils.read_image('/content/drive/MyDrive/Detecto-Updated Dataset/Detecto-UpdatedDataset/Test/20220516_201938.jpg')
     predictions = model.predict(image)
     labels, boxes, scores = predictions
     show_labeled_image(image, boxes, labels)
```



To calculate the nutritional value of the food items, we will return an array of labels detected in the image. Two pieces of steamed chicken were detected so **filtered_labels** will print an array containing these labels.

These labels will be used to fetch the nutritional value of the detected food items from the database and finally, the values will be summed up to return the total number of calories and nutrients present in the meal.




```
print(filtered_labels)
```

```
['steamedchicken', 'steamedchicken']
```

Web Application

Web Application is created using the MERN stack with RESTful API to deal with the interactions and changes, made during the run-time. The Front-end is made using ReactJS and Bootstrap.

Following dependencies are needed to be installed to successfully run the application:

- Express
- Mongoose (for MongoDB)
- NodeJs
- Nodemon
- Dotenv
- JSON-webtokens

- Axios
- Crypto-js(to encrypt and decrypt the user credentials)

```
"dependencies": {
  "crypto-js": "^4.1.1",
  "dotenv": "^10.0.0",
  "express": "^4.17.1",
  "jsonwebtoken": "^8.5.1",
  "mongoose": "^6.0.12",
  "multer": "^1.4.4",
  "nodemon": "^2.0.14",
```

Uploading an Image

User will upload the image which will be wrapped into FormData and will be sent to the server using API-calls through Axios.

```
fileUploadHandler = () => {
  this.handleChange();

  const fd = new FormData();
  fd.append('productImage', this.state.selectedFile, this.state.selectedFile.name)
  axios.post('http://localhost:5000/api/calculator/calculate-nutrients', fd).then(res => {
    console.log(res.data);
  });
}
```

Server-Side Working: The call will be directed to main index.js file of the server where routes are already defined for the corresponding actions.

```
JS index.js > ...
1   const express = require('express');
2   const app = express();
3   const mongoose = require('mongoose');
4   const dotenv = require('dotenv');
5   const userRouter = require('./routes/user');
6   const authRouter = require('./routes/auth');
7   const nutri_calcRouter = require('./routes/nutri_calc');
8   const userStatsRouter = require('./routes/user_stats');
9   dotenv.config();  //Tp use dotenv this fn needs to be called.
10
11  mongoose.connect(
12      process.env.MONGO_URL
13  ).then(
14      () => console.log(
15          "DB is connected!"
16      )).catch((err) => { console.log(err) });
17
18  app.use(express.json()); //To enable Server to process JSON FILES
19  app.use('/api/auth', authRouter);
20  app.use('/api/users', userRouter);
21  app.use('/api/calculator', nutri_calcRouter);
22  app.use('/api/userstats', userStatsRouter);
23  app.listen(process.env.PORT_NO || 5000, () => {
24      console.log('Server is Connected!');
25  });
```

Then it will be further guided to the method which deals with the call.

```
1  const router = require('express').Router();
2  const { verifyToken, verifyTokenAndAuthorization, verifyTokenAndAdmin } = require('./verifyToke
3  const multer = require('multer');
4  const { spawn } = require('child_process');
5
```

```
    const childPython = spawn('python', ['codespace.py', req.file, "My name is Husnain"]);
    try {

        await childPython.stdout.on('data', async (data) => {

        data1 = JSON.parse(data);
        // data3 = await data1[2];
        // console.log(`stdout: ${data1[2]}`);
        res.status(200).json(data1);
     }) ;

    } catch (err){
        res.status(500).json(err);
    }



    childPython.stderr.on('data', (data) => {
        console.log(`stdout: ${data}`);
```

Using spawn component, a child process will be created to use the python script in which Image-preprocessing techniques and machine learning algorithms are defined to be applied on the image.

Authentication And Authorization: For authentication and authorization, JSON-webtokens are used and for encryption of user credentials crypto-js is used.

```
const router = require('express').Router();
const User = require('../models/User');
const CryptoJS = require('crypto-js'); //first install crypto-js...
const jwt = require('jsonwebtoken');
```

Android Application

```
package com.example.nutritionalvaluecalculator;

import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;
```

```
public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

       Button Upload = findViewById(R.id.btnUploadPhoto);
       Upload.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View view) {
        ImagePicker.Companion.with( activity: MainActivity.this)
                .crop()                    //Crop image(Optional), Check Customization for more option
                .compress(1024)            //Final image size will be less than 1 MB(Optional)
                .maxResultSize( width: 1080,  height: 1080)  //Final image resolution will be less than 1080 x 1080(Optional)
                .start();
```

- The upload button is created which allows the user to upload a photo.
- ImagePicker API is used which allows the user to crop, compress, and adjust the image size.

```
Button Camera = findViewById(R.id.btnTakePhoto);
Camera.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View view) {
        ImagePicker.Companion.with( activity: MainActivity.this)
                .cameraOnly()
                .crop()                    //Crop image(Optional), Check Customization for more option
                .compress(1024)            //Final image size will be less than 1 MB(Optional)
                .maxResultSize( width: 1080,  height: 1080)  //Final image resolution will be less than 1080 x 1080(Optional)
                .start();
    }
```

The Camera button is created which allows the user to take a photo. When user uploads or takes a photo, processing is initiated to display the nutritional value.

Additional Features

Signup

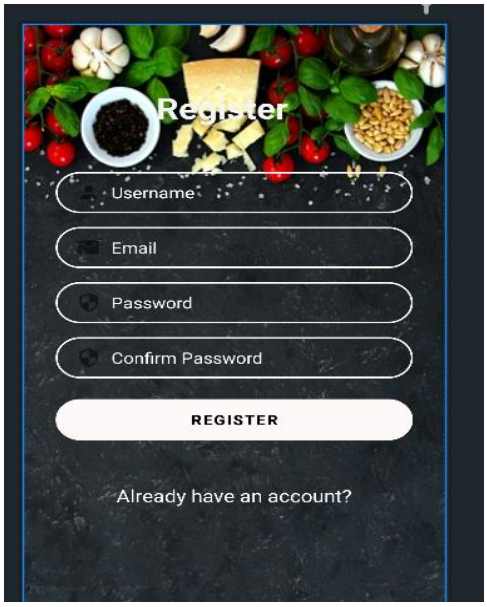
```
package com.example.nutritionalvaluecalculator;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.TextView;
```

```
public class SignUp extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_sign_up);
        TextView btn=findViewById(R.id.alreadyhaveacc);
        btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) { startActivity(new Intent( packageContext: SignUp.this,Login.class)); }
        });
```



Login

```
public class Login extends AppCompatActivity {
    GoogleSignInOptions gso;
    GoogleSignInClient gsc;
    Button googleBtn;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);
        TextView username =(TextView) findViewById(R.id.inputUserName);
        TextView password =(TextView) findViewById(R.id.inputPassword);
        TextView login =  findViewById(R.id.btnlogin);
        TextView btn=findViewById(R.id.textViewSignUp);
        btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) { startActivity(new Intent( packageContext: Login.this,SignUp.class)); }
        });
```

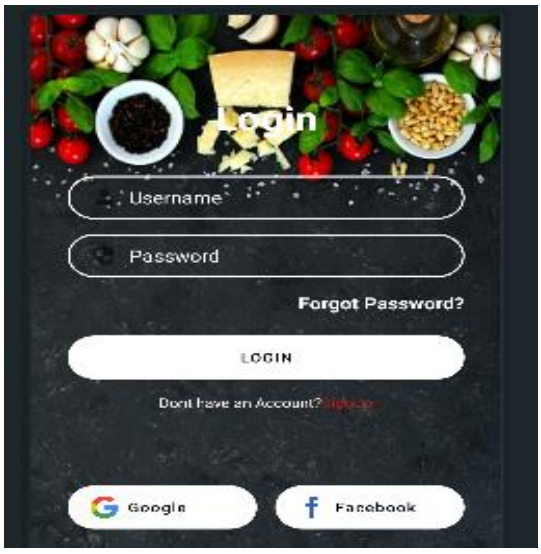
- Username and Password fields are created where user enters his/her credentials in the respective fields.
- If user clicks on Login button, he is redirected to a page where he can take or upload a photo and if user clicks on sign-up button, he is redirected to a sign-up page.

```
login.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if(username.getText().toString().equals("admin") && password.getText().toString().equals("admin")){
            //correct
            // Toast.makeText(Login.this,"LOGIN SUCCESSFUL",Toast.LENGTH_SHORT).show();
            startActivity(new Intent( packageContext: Login.this, MainActivity.class));
        }else
            //incorrect
            Toast.makeText( context: Login.this, text: "LOGIN FAILED !!!",Toast.LENGTH_SHORT).show();
```

- Credentials are checked and if they are correct, user can go to next page but if the credentials are incorrect, a prompt appears with the message “login failed!”

```
@Override
protected void onActivityResult(int requestCode, int resultCode,Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    if(requestCode == 1000){
        Task<GoogleSignInAccount> task = GoogleSignIn.getSignedInAccountFromIntent(data);

        try {
            task.getResult(ApiException.class);
            navigateToSecondActivity();
        } catch (ApiException e) {
            Toast.makeText(getApplicationContext(),  text: "Something went wrong", Toast.LENGTH_SHORT).show();
```



5: Conclusion

In this thesis, we discussed a nutritional value calculator system which can assist the users in dealing with diet-related illnesses by recommending customized diet plans and calculating nutritional value of meals. Our proposed system has an advantage over other traditional systems due to the latest algorithms used for the detection of food items. Additionally, the objectives of tracking daily calorie intake and controlling food-related ailments are attained efficiently. Hence, this project saves time as well as the precious lives of its users.

Our proposed system is easy to use as the user is not required to be skilled in any of the technologies used for its development. Moreover, "**Nutritional Value Calculator**" can be deployed on a larger scale without the requirement of providing product training to its users.

6: Future Work

Future milestones that need to be achieved to commercialize this project are the following:

Expanding the Dataset:

The main objective of this project is to predict the nutritional value of a meal using its image. Initially, the dataset of food images fed to the system is smaller. For its implementation on a larger scale, we need to work on a dataset which incorporates a greater number of cuisines.

7: References and Work Cited

1. https://www.kaggle.com. 2018. Food 101 dataset containing 101 classes of food items [online] Available at: https://www.kaggle.com/datasets/dansbecker/food-101

2. https://www.niddk.nih.gov/. 2018. Prevalence of Overweight and Obesity. [online] Available at: https://www.niddk.nih.gov/health-information/health-statistics/overweight-obesity

3. WHO.int. 2020. Malnutrition Key Facts [online] Available at: https://www.who.int/news-room/fact-sheets/detail/malnutrition

4. https://www.aa.com.tr/ . 2022. Available at: https://www.aa.com.tr/en/asia-pacific/40-of-children-in-pakistan-suffer-from-obesity/2522481

5. Unicef.org. Pakistan. National Nutrition Survey 2018 [online] Available at: https://www.unicef.org/pakistan/media/1871/file/KeyFindings---NationalNutritionSurvey2018.pdf

6. https://ieeexplore.ieee.org/. 2018. A Framework to Estimate the Nutritional Value of Food in Real Time Using Deep Learning Techniques. [online] Available at: https://ieeexplore.ieee.org/abstract/document/8590712

7. https://www.ncbi.nlm.nih.gov/. 2020. Object Detection Based on Faster R-CNN Algorithm with Skip Pooling and Fusion of Contextual Information. [online] Available at: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7582940/

8. https://ieeexplore.ieee.org/. 2012. A brief introduction to OpenCV. [online] Available at: https://ieeexplore.ieee.org/document/6240859

Nutrition Value Calculator

ORIGINALITY REPORT

7%	4%	1%	6%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Higher Education Commission Pakistan Student Paper	4%
2	Submitted to Asia Pacific University College of Technology and Innovation (UCTI) Student Paper	1%
3	www.coursehero.com Internet Source	1%
4	www.unicef.org Internet Source	<1%
5	Submitted to University of Ulster Student Paper	<1%
6	www.karger.com Internet Source	<1%
7	mafiadoc.com Internet Source	<1%

Exclude quotes	On	Exclude matches	Off