

## ASSIGNMENT- 2

|                 |                                 |
|-----------------|---------------------------------|
| Assignment Date | 25 September 2022               |
| Project Name    | Nutrition Assistant Application |
| Team ID         | PNT2022TMID00181                |
| Maximum Marks   | 2 Marks                         |

### 1.Create a Flask App

```
1 from flask import Flask, render_template, flash, redirect, url_for, session, request, logging
2 #from data import Articles
3 from flask.mysql import MySQL
4 from wtforms import Form, StringField, TextAreaField, PasswordField, validators
5 from passlib.hash import sha256_crypt
6 from functools import wraps
7
8 app = Flask(__name__)
9
10 # Config MySQL
11 app.config['MYSQL_HOST'] = 'localhost'
12 app.config['MYSQL_USER'] = 'root'
13 app.config['MYSQL_PASSWORD'] = '123456'
14 app.config['MYSQL_DB'] = 'myflaskapp'
15 app.config['MYSQL_CURSORCLASS'] = 'DictCursor'
16 # init MySQL
17 mysql = MySQL(app)
18
19 #Articles = Articles()
20
21 # Index
22 @app.route('/')
23 def index():
24     return render_template('home.html')
25
26 # About
27 @app.route('/about')
28 def about():
29     return render_template('about.html')
30
31 # Articles
```

```
35 def articles():
36     # Create cursor
37     cur = mysql.connection.cursor()
38
39     # Get articles
40     result = cur.execute("SELECT * FROM articles")
41
42     articles = cur.fetchall()
43
44     if result > 0:
45         return render_template('articles.html', articles=articles)
46     else:
47         msg = 'No Articles Found'
48         return render_template('articles.html', msg=msg)
49     # Close connection
50     cur.close()
51
52 #Single Article
53 @app.route('/article/<string:id>')
54 def article(id):
55     # Create cursor
56     cur = mysql.connection.cursor()
57
58     # Get article
59     result = cur.execute("SELECT * FROM articles WHERE id = %s", [id])
60
61     article = cur.fetchone()
62
63     return render_template('article.html', article=article)
64
65 # Articles Form class
```

## 2) Add Home Page and About Page

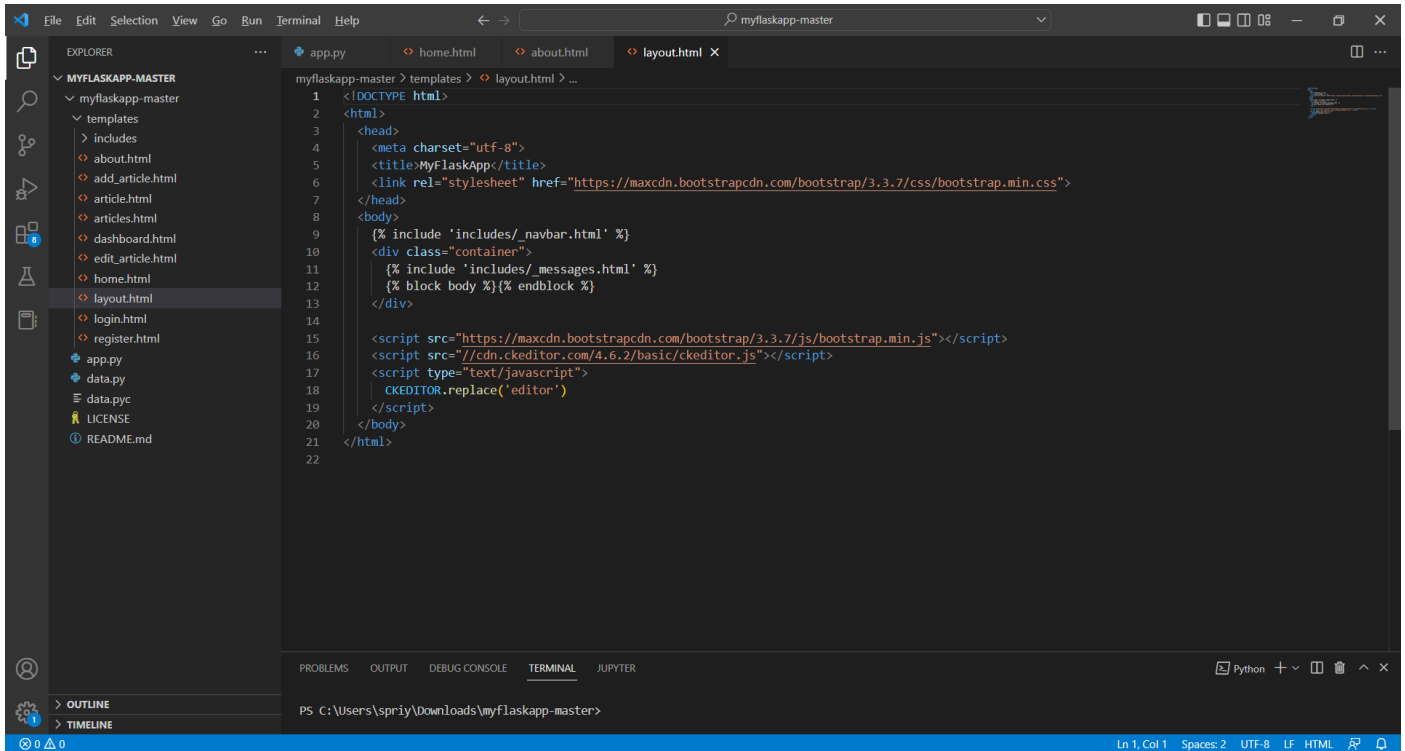


### About Us

We are working on nutrition assistant applications

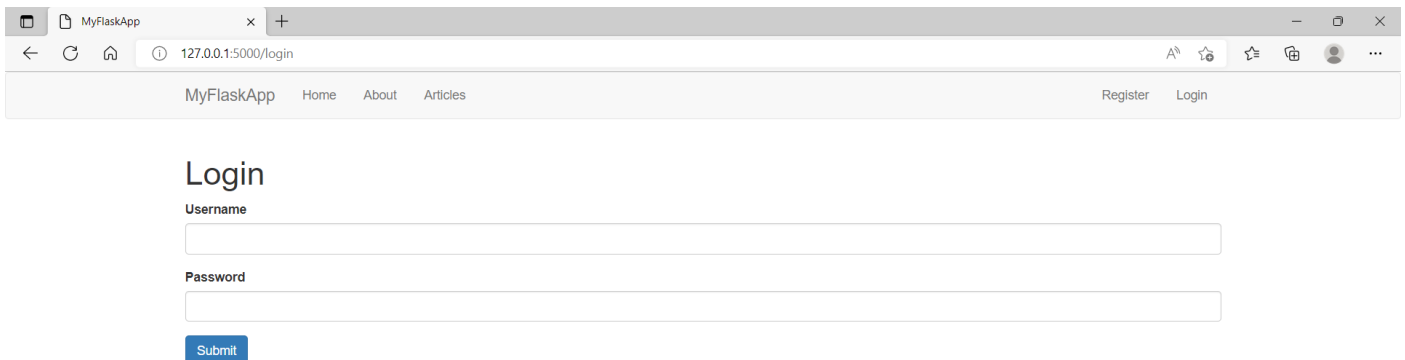
This project aims at building a web App that automatically estimates food attributes such as ingredients and nutritional value by classifying the input image of food. Our method employs Clarifai's AI-Driven Food Detection Model for accurate food identification and Food API's to give the nutritional value of the identified food.

### 3) Add a Bootstrap



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta charset="utf-8">
5 <title>MyFlaskApp</title>
6 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
7 </head>
8 <body>
9 {% include 'includes/_navbar.html' %}
10 <div class="container">
11 {% include 'includes/_messages.html' %}
12 {% block body %}{% endblock %}
13 </div>
14
15 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>
16 <script src="//cdn.ckeditor.com/4.6.2/basic/ckeditor.js"></script>
17 <script type="text/javascript">
18     CKEDITOR.replace('editor')
19 </script>
20 </body>
21 </html>
22
```

### 4) Add sign -in page and database connectivity



MyFlaskApp Home About Articles Register Login

## Login

Username

Password

Submit

You are now logged in

No Articles Found

## Dashboard

Welcome allan123

Add Article

| ID | Title | Author | Date |
|----|-------|--------|------|
|----|-------|--------|------|

## Register

Name

Email

Username

Password

Confirm Password

Submit

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

hotel

Tables

Views

Stored Procedures

Functions

hr\_database

myflaskapp

Tables

articles

users

Views

Stored Procedures

Functions

northwind

sakila

sakthi

enl hr

Administration Schemas

Information

Schema: myflaskapp

SQL File 2

```
1 • create database myflaskapp;
2 • use myflaskapp;
3 • create table users (id int(11) auto_increment primary key, name varchar(100), email varchar(100),
4 • username varchar(100), password varchar(100), register_date
```

Result Grid

|     | id     | name      | email    | username                    | password            | register_date |
|-----|--------|-----------|----------|-----------------------------|---------------------|---------------|
| ▶ 1 | sakthi | saksdsdgv | sakthi   | \$5\$rounds=535000\$J.Pn... | 2022-10-28 20:45:50 |               |
| ▶ 2 | ram    | abc@123   | ram123   | \$5\$rounds=535000\$6L4...  | 2022-10-28 20:50:27 |               |
| ▶ 3 | allan  | allan123  | allan123 | \$5\$rounds=535000\$y2R...  | 2022-10-28 20:59:46 |               |

users 2

Output

Action Output

| #  | Time     | Action  | Message  | Duration / Fetch      |
|----|----------|---|--|-----------------------|
| 9  | 20:58:46 | create table articles(id int(11) auto_increment primary key , title varchar(100) , author varchar(100) , body text, ... | 0 row(s) affected, 1 warning(s): 1681 Integer display width is deprecated and will be removed in a future release. | 0.031 sec             |
| 10 | 21:35:27 | select * from users LIMIT 0, 1000   | 3 row(s) returned  | 0.000 sec / 0.000 sec |

Object Info Session

File Edit Selection View Go Run Terminal Help

myflaskapp-master

EXPLORER

MYFLASKAPP-MASTER

myflaskapp-master

templates

includes

about.html

add\_article.html

article.html

articles.html

dashboard.html

edit\_article.html

home.html

layout.html

login.html

register.html

app.py

data.py

data.pyc

LICENSE

README.md

app.py

```
44 if result > 0:
45     return render_template('articles.html', articles=articles)
46 else:
47     msg = 'No Articles Found'
48     return render_template('articles.html', msg=msg)
49 # Close connection
50 cur.close()
51
52
53 #Single Article
54 @app.route('/article/<string:id>/')
55 def article(id):
56     # Create cursor
57     cur = mysql.connection.cursor()
58
59     # Get article
60     result = cur.execute("SELECT * FROM articles WHERE id = %s", [id])
61
62     article = cur.fetchone()
63
64     return render_template('article.html', article=article)
65
66
67 # Register Form Class
68 class RegisterForm(Form):
69     name = StringField('Name', [validators.Length(min=1, max=50)])
70     username = StringField('Username', [validators.Length(min=4, max=25)])
71     email = StringField('Email', [validators.Length(min=6, max=50)])
72     password = PasswordField('Password', [
73         validators.DataRequired(),
74         validators.EqualTo('confirm', message='Passwords do not match')
75     ])
76     confirm = PasswordField('Confirm Password')
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

Python

PS C:\Users\spriy\Downloads\myflaskapp-master>

Ln 53, Col 4 (302 selected) Spaces: 4 UTF-8 LF Python 3.10.4 64-bit