

Flask Deployment

Name: Mahnoor Farhat

Date: 28th January, 2025

Internship Batch: LISUM41: 30 December, 2024 – 31 March, 2025

URL: <https://github.com/mahnoor-farhat/flask-deployment>

Deployment Steps:

```
<> index.html X
templates > <> index.html > ...
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Iris Prediction</title>
5  </head>
6  <body>
7      <h1>Predict Iris Species</h1>
8      <form action="{url_for('/predict')}}" method="POST">
9          <label>Sepal Length:</label>
10         <input type="text" name="sepal_length"><br>
11         <label>Sepal Width:</label>
12         <input type="text" name="sepal_width"><br>
13         <label>Petal Length:</label>
14         <input type="text" name="petal_length"><br>
15         <label>Petal Width:</label>
16         <input type="text" name="petal_width"><br>
17         <button type="submit">Predict</button>
18     </form>
19     {% if result %}
20         <h2>Prediction: {{ result }}</h2>
21     {% endif %}
22 </body>
23 </html>
24
```

iris.py X

iris.py > ...

```
1 import pickle
2 from sklearn.datasets import load_iris
3 from sklearn.ensemble import RandomForestClassifier
4 from sklearn.model_selection import train_test_split
5
6 data = load_iris()
7 X, y = data.data, data.target
8 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
9
10 model = RandomForestClassifier()
11 model.fit(X_train, y_train)
12
13 with open("iris_model.pkl", "wb") as file:
14     pickle.dump(model, file)
15
16 print("Model saved as iris_model.pkl")
```

app.py X

app.py > home

```
1 import pandas as pd
2 import numpy as np
3 import pickle
4 from flask import Flask, request, render_template
5
6 app = Flask(__name__)
7 model = pickle.load(open('iris_model.pkl', 'rb'))
8
9 @app.route('/')
10 def home():
11     return render_template('index.html')
12
13 @app.route('/predict', methods=['POST'])
14 def predict():
15     try:
16         # Log form data
17         print("Form Data Received:", request.form)
18
19         sepal_length = float(request.form['sepal_length'])
20         sepal_width = float(request.form['sepal_width'])
21         petal_length = float(request.form['petal_length'])
22         petal_width = float(request.form['petal_width'])
23
24         input_data = np.array([[sepal_length, sepal_width, petal_length, petal_width]])
25         print("Input Data for Prediction:", input_data)
```

```
app.py ×
app.py > home
14 def predict():
26
27     prediction = model.predict(input_data)
28     species = ["Setosa", "Versicolor", "Virginica"][prediction[0]]
29     print("Prediction Result:", species)
30
31     return render_template("index.html", result=species)
32 except Exception as e:
33     print("Error in Prediction:", e)
34     return "Error: " + str(e)
35
36
37 if __name__ == '__main__':
38     app.run(port=5000, debug=True)
```

app

← → ↑ ↺ Error > ... Desktop > app > Search app 🔍

+ New ✂ Copy Paste Print Share Delete Sort View ... Details

	Name	Status	Date modified	Type
Gallery	templates	🔄	28/01/2025 19:08	File folder
> Mahnoor - Perso	app	❌	28/01/2025 19:49	PY File
	iris	❌	28/01/2025 19:02	PY File
	iris_model.pkl	❌	28/01/2025 19:02	PKL File

Desktop Downloads Documents

```
C:\Windows\System32\cmd.e × + ▾  
Microsoft Windows [Version 10.0.26100.2894]  
(c) Microsoft Corporation. All rights reserved.  
  
C:\Users\SAAD COMMUNICATION\OneDrive\Desktop\app>python app.py  
* Serving Flask app 'app'  
* Debug mode: on  
WARNING: This is a development server. Do not use it in a production  
n deployment. Use a production WSGI server instead.  
* Running on http://127.0.0.1:5000  
Press CTRL+C to quit  
* Restarting with stat  
* Debugger is active!  
* Debugger PIN: 267-230-033
```