

# Iris Categorization

## Flask Application with PostMan Evaluation

Name: Chris Donaton  
Internship Batch: LISUM16  
Date: 29/12/2022  
Submitted to: Data Glacier

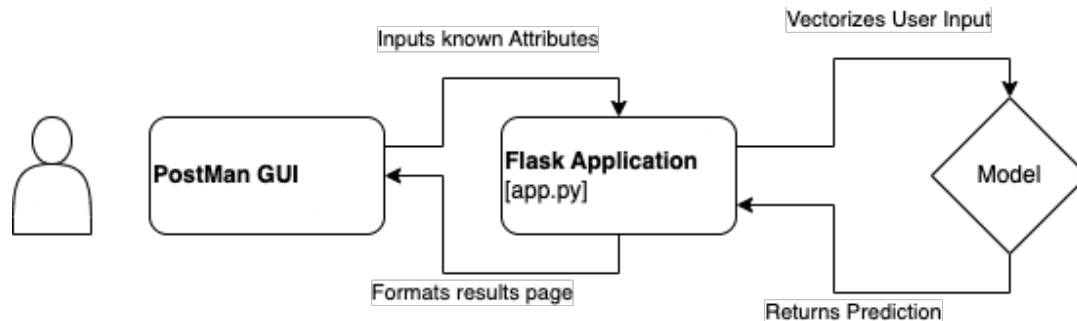


**Data Glacier**

Your Deep Learning Partner

This application incorporates the classic dataset, the classification of Iris flowers. The dataset consists of 50 samples, each with five attributes: the sepal length and width, the petal length and width, and the actual species of each flower. Each flower is one of four species.

The application uses a Decision Tree Classifier to predict the species of Iris based on the length and width of its sepal and petal. The following diagram details the front and back-end workflow for the application.

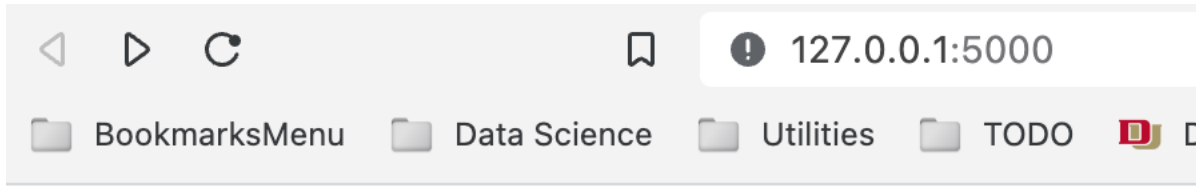


A key learning point here was understanding the difference between POST and GET methods as they pertain to the HTML, FLASK API, and POSTMAN interface. It is necessary all components in the pipeline are interoperable for maximum success.

# Getting it running

```
week04 — -zsh — 80x24
[(base) pumablade@PumaBlade-2 week04 % python3 -m venv simpleFlask
[(base) pumablade@PumaBlade-2 week04 % . simpleFlask/bin/activate
[simpleFlask] (base) pumablade@PumaBlade-2 week04 % python app.py
python          python3-intel64    python3.10-config  python3.11-intel64
python3         python3.1         python3.11
python3-config  python3.10        python3.11-config
```

```
week04 — python app.py — 80x24
[simpleFlask] (base) pumablade@PumaBlade-2 week04 % python app.py
* 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
```



## Iris Classification

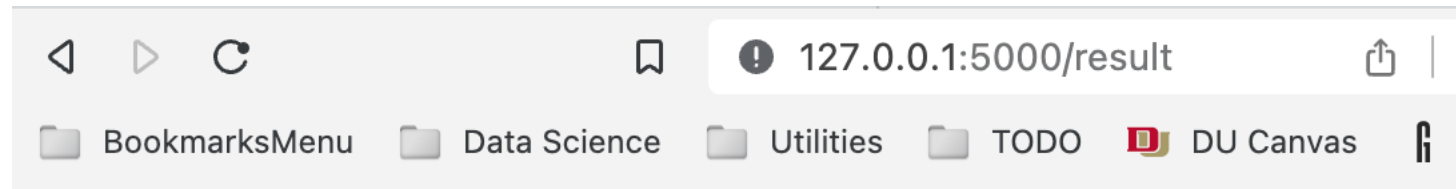
Sepal Length

Sepal Width

Petal Length

Petal Width

# Testing it out



## You have found a Versicolor!