Flower Prediction: Predicting Flowers with Precision

Submitted in partial fulfillment of the requirements of the degree of

Bachelor of Engineering (Information Technology)

By

Laksh Sodhai Roll No – 57

Department of Information Technology

VIVEKANAND EDUCATION SOCIETY'S INSTITUTE OF TECHNOLOGY,
Chembur, Mumbai 400074
(An Autonomous Institute, Affiliated to University of Mumbai)
April 2024

Content	**Page No.**
Project Description	1-2
Requirement gathering	1-2
System requirements	2-3
Technologies used	2-3
Setup instructions	4-6
Project structure	6-7
Architectural diagrams	7-9
Features implemented	9-10
Screenshots of implementation	11-14
Future scope	14-15
Github link	14-15
Conclusion	15

FLOWER PREDICTION

Name of student	Laksh Sodhai
Class_Roll no	D15A_57
D.O.P	20/03/25
D.O.S	27/03/25
Sign and Grade	

Title : **Flower Prediction**

Project Description :

The Flower Prediction project is a machine learning-based web application designed to classify flowers into different species based on user input of sepal and petal dimensions. It uses the well-known Iris dataset and employs algorithms like Decision Tree or K-Nearest Neighbors for accurate predictions. The backend is built with Flask in Python, integrating a pre-trained model using scikit-learn. A simple, interactive frontend allows users to input flower measurements and view the predicted species instantly. The system provides real-time inference with minimal latency. It's ideal for educational demonstrations and botanical research. The app structure follows modular design principles, ensuring scalability and easy maintenance. The prediction logic is exposed through RESTful APIs for seamless frontend-backend communication.

Requirement gathering : To develop the Flower Prediction system, both functional and non-functional requirements were identified. Functionally, the system must allow users to input four key flower features: sepal length, sepal width, petal length, and petal width. It should process these inputs through a trained ML model and return the predicted flower species. A clean and intuitive interface is required for easy input and display of results.
On the technical side, the backend should support Python with Flask and scikit-learn, while the frontend may use basic HTML/CSS or any lightweight framework. The system must run smoothly on local machines with minimal configuration and ensure accurate, fast predictions. Optionally, logging or result history may be added using a simple database like SQLite or MongoDB.

System Requirements :

1.Hardware Requirements:

- **Processor:** Intel Core i5 / AMD Ryzen 5 or higher (dual-core, 2.0 GHz or faster)
- **RAM:** Minimum 8GB (16GB recommended)
- **Storage:** At least 1GB free space (256GB SSD recommended)
- **Network:** Stable internet connection (especially for MongoDB Atlas users)

2. Software Requirements:

- **Operating System:** Windows 10/11, macOS 10.15+, or Ubuntu 20.04+
- **Code Editor:** Visual Studio Code or compatible IDE
- **Version Control:** Git 2.25+
- **Python:** Version 3.8 or higher

Technologies Used :

Development	VS Code , Postman , Git
Frontend	HTML/CSS/Typescript(or Streamlit/Flask Templates)
Backend	Flask (Python 3.8+)
ML Model	Scikit-learn
Styling	SCSS / Bootstrap
APIs	RESTful Flask APIs

Setup Instructions :

1. **Python 3.8+:** To set up the Flower Prediction project, first install Python 3.8 or higher from the official Python website (python.org). During installation (especially on Windows), make sure to check the option "Add Python to PATH." After installation, verify it using python --version and pip --version in the terminal. For better environment management, you can create a virtual environment using python -m venv venv, and activate it with venv\Scripts\activate on Windows or source venv/bin/activate on macOS/Linux.
2. **Flask and Dependencies:** Once Python is set up, install the necessary packages by running pip install -r requirements.txt in your project directory. This will install Flask, scikit-learn, and other required libraries. Make sure the requirements.txt file is up to date with your project dependencies.
3. **ML Model Setup**: Ensure that your trained ML model (e.g., a .pkl file) is saved in the appropriate directory (e.g., a model/ folder). The Flask backend will load this model at runtime to perform predictions based on user inputs.

Backend Setup :

1. **Navigate to backend folder:**

```
cd project
```

2. **(Optional) Create a virtual environment:**

```
python -m venv venv
venv\Scripts\activate   # For Windows
```

3. **Install dependencies:**

```
pip install -r requirements.txt
```

4. **Start the Flask server:**

```
cd api
python app.py
```

Backend will run at: http://localhost:5000

Frontend Setup

1. **Navigate to frontend folder:**

```
cd project
```

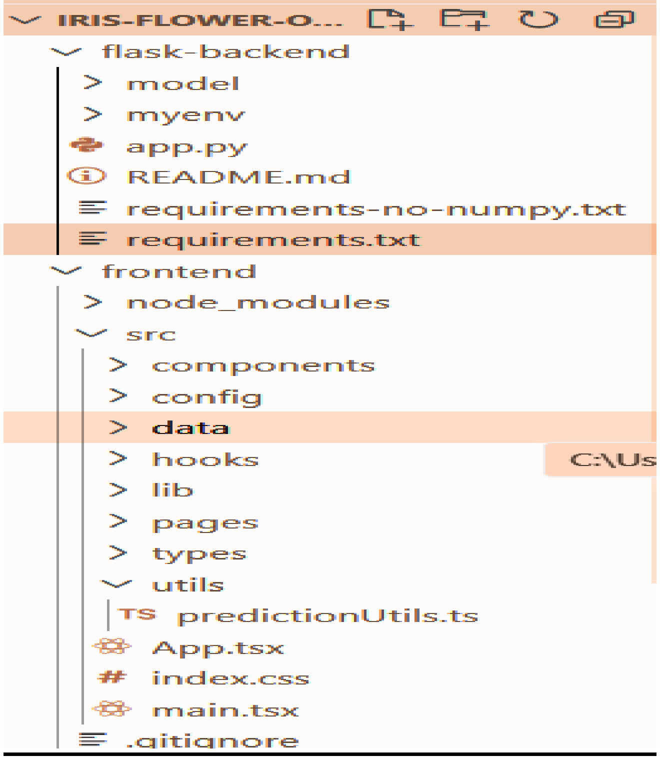
2. **Install dependencies:**

```
npm install
```

3. **Npm run dev**

Frontend will run at: http://localhost:8081

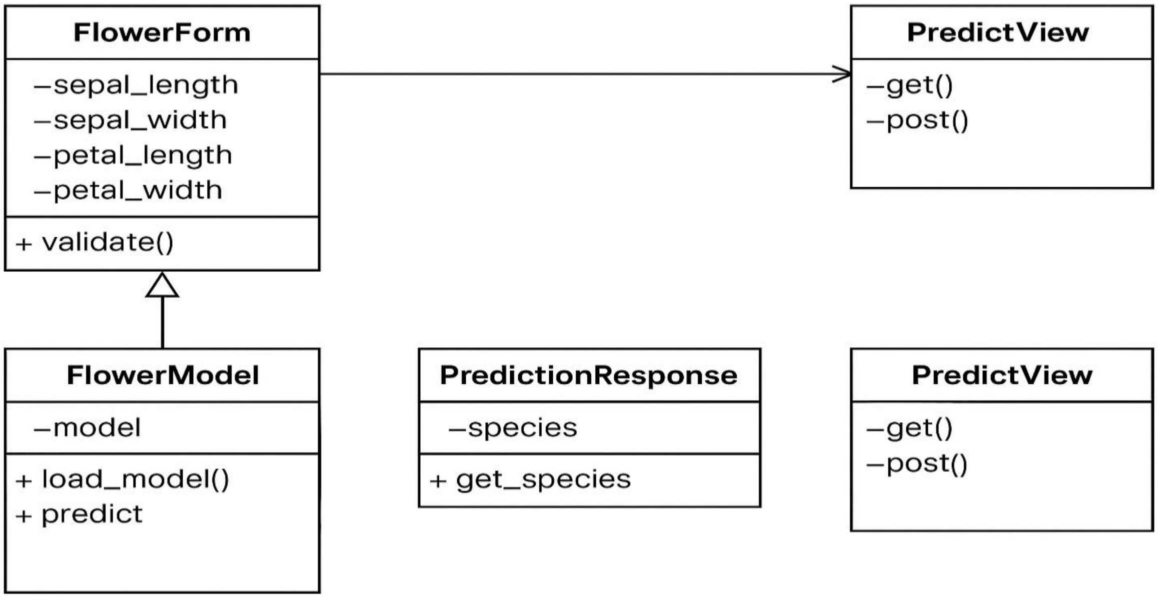
Project Structure:



```
IRIS-FLOWER-O...
  flask-backend
    > model
    > myenv
    app.py
    README.md
    requirements-no-numpy.txt
    requirements.txt
  frontend
    > node_modules
    src
      > components
      > config
      > data
      > hooks                C:\Us
      > lib
      > pages
      > types
      utils
        predictionUtils.ts
      App.tsx
      index.css
      main.tsx
    .gitignore
```

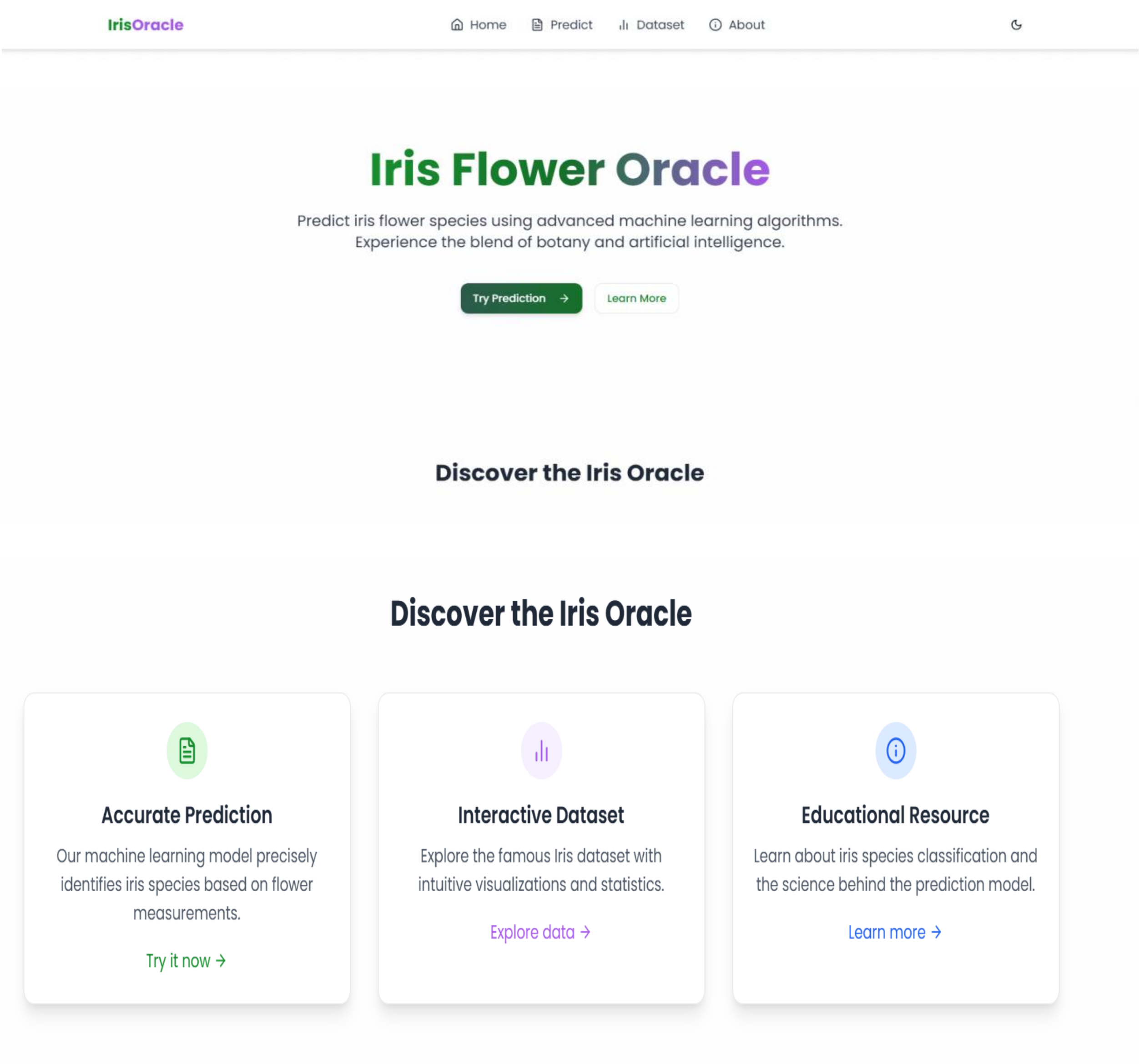
Architectural Diagrams :

a) Class Based Diagram -



Screenshots of implementation:

Home Page:



Iris Flower Oracle

Predict iris flower species using advanced machine learning algorithms.
Experience the blend of botany and artificial intelligence.

Discover the Iris Oracle

Discover the Iris Oracle

Accurate Prediction

Our machine learning model precisely identifies iris species based on flower measurements.

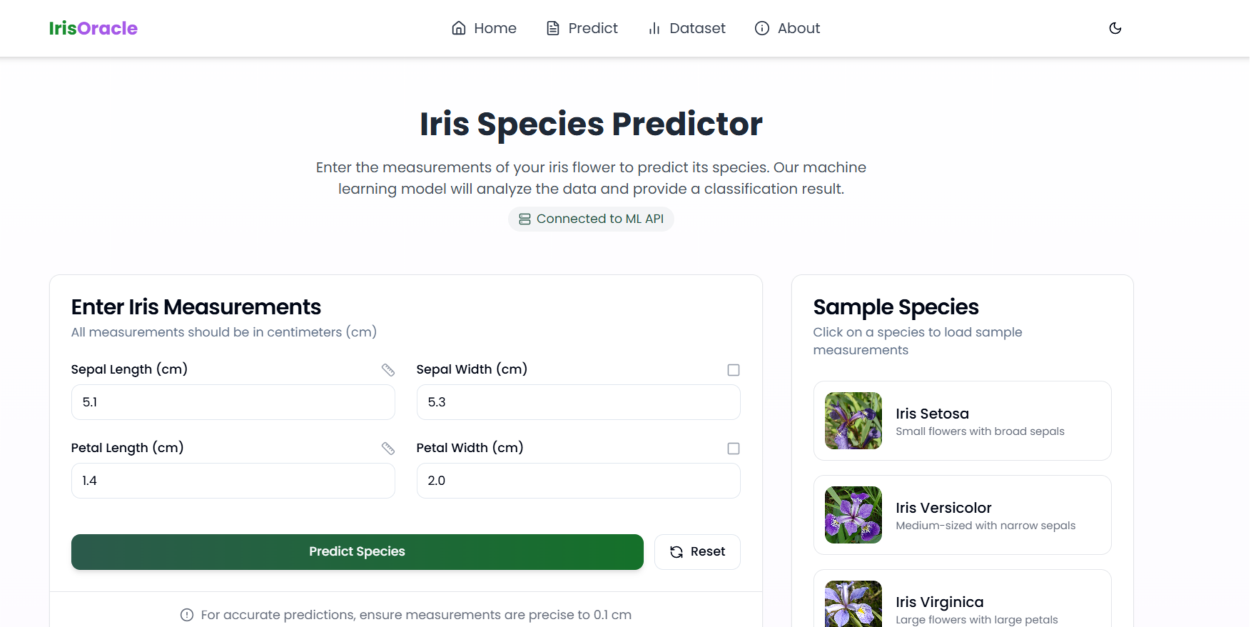
Interactive Dataset

Explore the famous Iris dataset with intuitive visualizations and statistics.

Educational Resource

Learn about iris species classification and the science behind the prediction model.

Prediction Page:



Iris Species Predictor

Enter the measurements of your iris flower to predict its species. Our machine learning model will analyze the data and provide a classification result.

Connected to ML API

Enter Iris Measurements

All measurements should be in centimeters (cm)

Sepal Length (cm): 5.1

Sepal Width (cm): 5.3

Petal Length (cm): 1.4

Petal Width (cm): 2.0

For accurate predictions, ensure measurements are precise to 0.1 cm

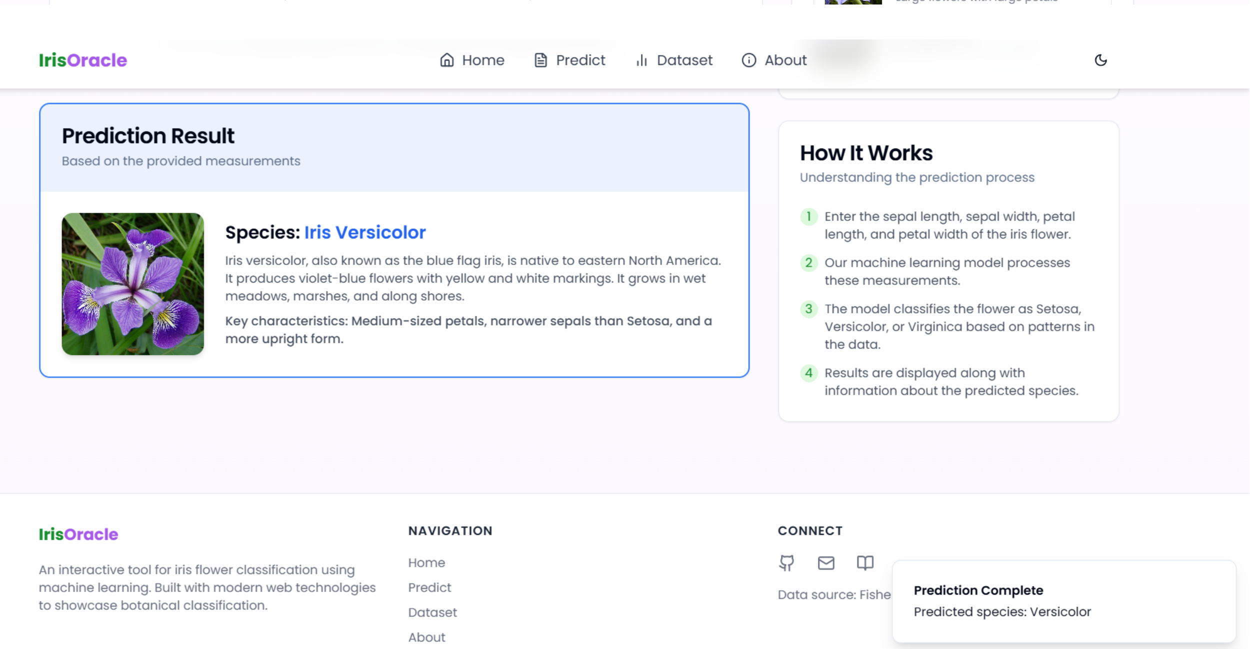
Sample Species

Click on a species to load sample measurements

Iris Setosa
Small flowers with broad sepals

Iris Versicolor
Medium-sized with narrow sepals

Iris Virginica
Large flowers with large petals



Prediction Result

Based on the provided measurements

Species: Iris Versicolor

Iris versicolor, also known as the blue flag iris, is native to eastern North America. It produces violet-blue flowers with yellow and white markings. It grows in wet meadows, marshes, and along shores.

Key characteristics: Medium-sized petals, narrower sepals than Setosa, and a more upright form.

How It Works

Understanding the prediction process

1. Enter the sepal length, sepal width, petal length, and petal width of the iris flower.
2. Our machine learning model processes these measurements.
3. The model classifies the flower as Setosa, Versicolor, or Virginica based on patterns in the data.
4. Results are displayed along with information about the predicted species.

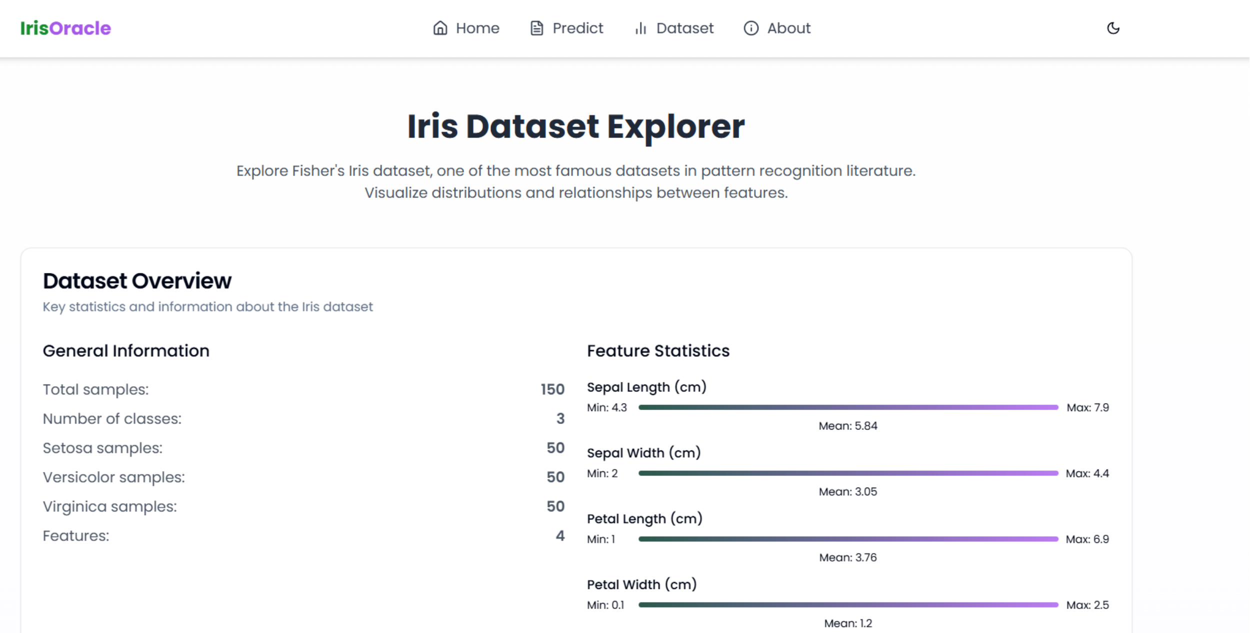
IrisOracle

An interactive tool for iris flower classification using machine learning. Built with modern web technologies to showcase botanical classification.

Data source: Fishe

Prediction Complete
Predicted species: Versicolor

Dataset Page:



Iris Dataset Explorer

Explore Fisher's Iris dataset, one of the most famous datasets in pattern recognition literature.
Visualize distributions and relationships between features.

Dataset Overview

Key statistics and information about the Iris dataset

General Information

Total samples:	150
Number of classes:	3
Setosa samples:	50
Versicolor samples:	50
Virginica samples:	50
Features:	4

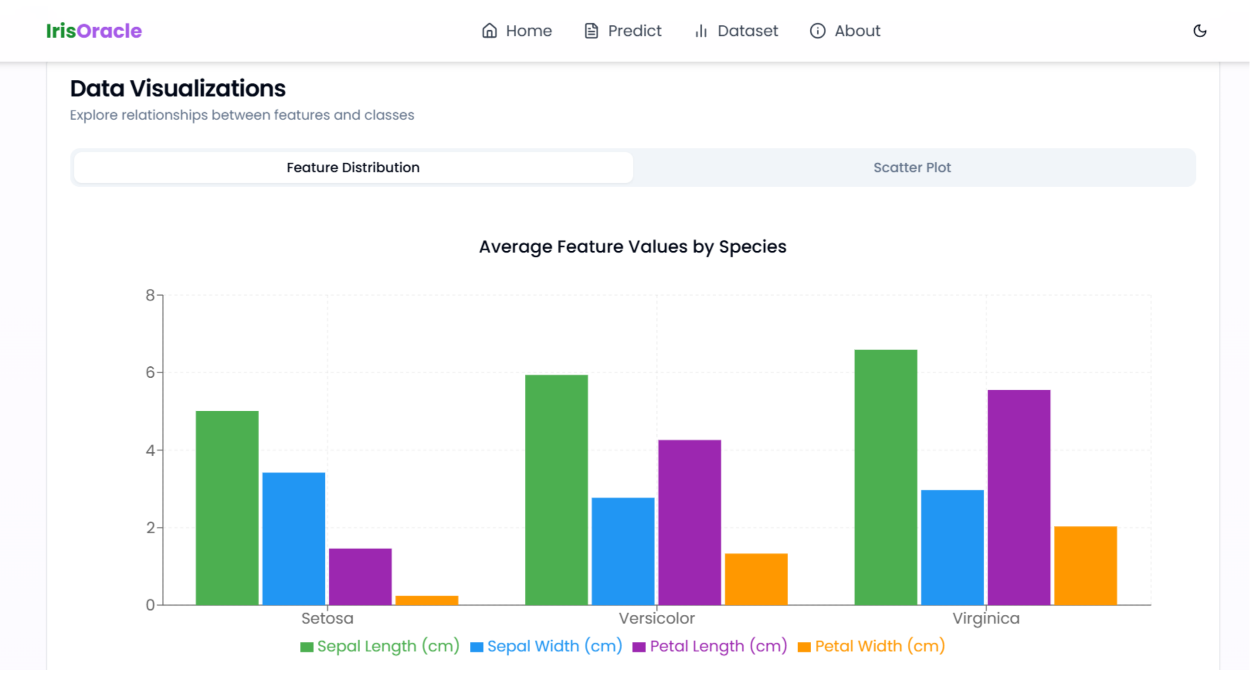
Feature Statistics

Sepal Length (cm)
Min: 4.3 Max: 7.9
Mean: 5.84

Sepal Width (cm)
Min: 2 Max: 4.4
Mean: 3.05

Petal Length (cm)
Min: 1 Max: 6.9
Mean: 3.76

Petal Width (cm)
Min: 0.1 Max: 2.5
Mean: 1.2

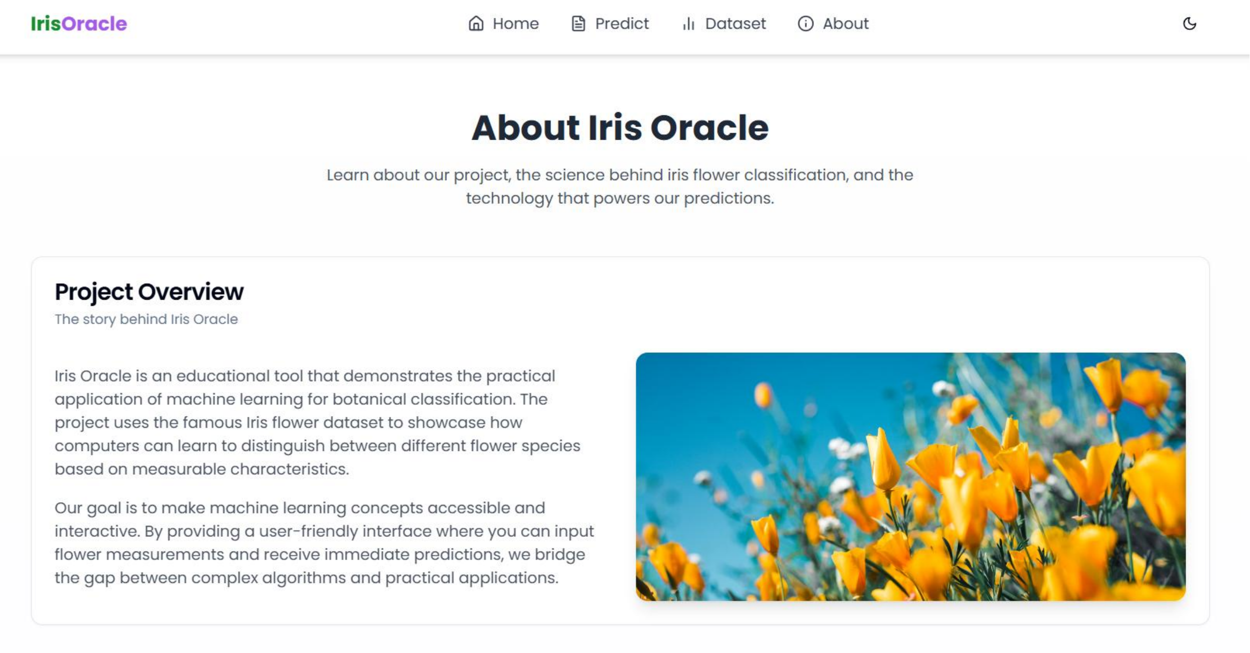


Data Visualizations

Explore relationships between features and classes

Average Feature Values by Species

About Page:



About Iris Oracle

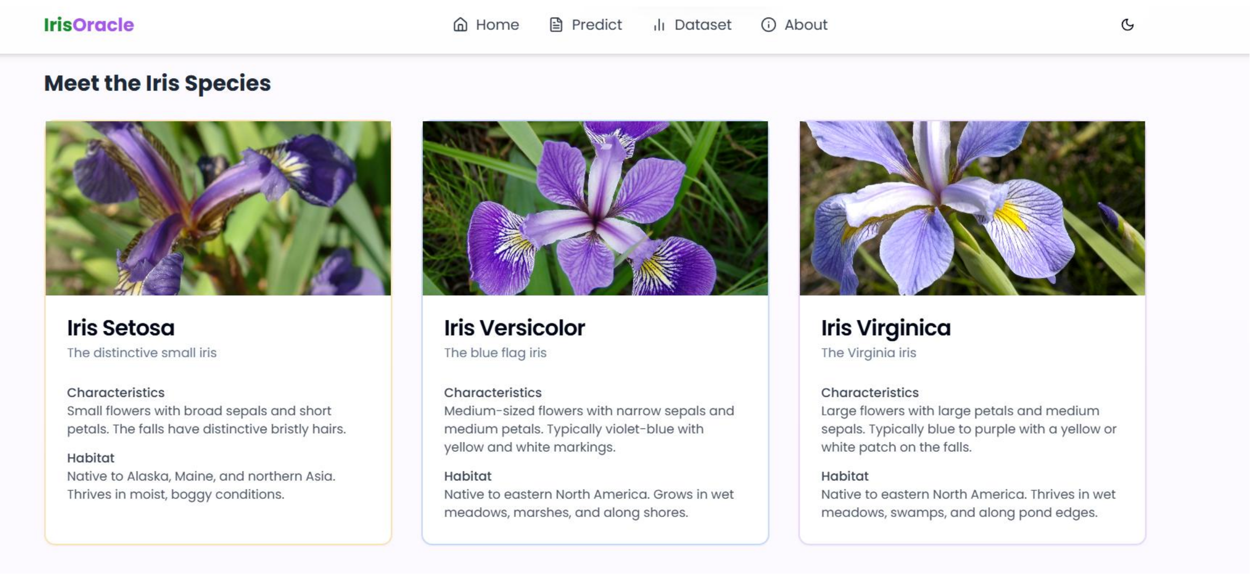
Learn about our project, the science behind iris flower classification, and the technology that powers our predictions.

Project Overview

The story behind Iris Oracle

Iris Oracle is an educational tool that demonstrates the practical application of machine learning for botanical classification. The project uses the famous Iris flower dataset to showcase how computers can learn to distinguish between different flower species based on measurable characteristics.

Our goal is to make machine learning concepts accessible and interactive. By providing a user-friendly interface where you can input flower measurements and receive immediate predictions, we bridge the gap between complex algorithms and practical applications.



Meet the Iris Species

Iris Setosa

The distinctive small iris

Characteristics

Small flowers with broad sepals and short petals. The falls have distinctive bristly hairs.

Habitat

Native to Alaska, Maine, and northern Asia. Thrives in moist, boggy conditions.

Iris Versicolor

The blue flag iris

Characteristics

Medium-sized flowers with narrow sepals and medium petals. Typically violet-blue with yellow and white markings.

Habitat

Native to eastern North America. Grows in wet meadows, marshes, and along shores.

Iris Virginica

The Virginia iris

Characteristics

Large flowers with large petals and medium sepals. Typically blue to purple with a yellow or white patch on the falls.

Habitat

Native to eastern North America. Thrives in wet meadows, swamps, and along pond edges.

Future Scope : The Flower Prediction system can be expanded in several ways. Integration with mobile applications could make it accessible on-the-go for botanists, researchers, and students. Advanced models like deep learning (e.g., CNNs) could be used to predict species from flower images, enhancing accuracy and usability. The system can be extended to include a wider variety of plant species beyond the Iris dataset. Additionally, real-time data collection through IoT sensors in smart gardens can be incorporated. Integration with cloud platforms can allow for scalable deployment and analytics.

Github Link : **https://github.com/laksh-59/WebX-project**

Conclusion:

The Flower Prediction project demonstrates how machine learning can be effectively applied to classify flower species based on measurable features. By utilizing the Iris dataset and implementing a user-friendly web interface, the system provides quick and accurate predictions. It bridges the gap between data science and real-world applications in botany and education. The integration of Flask as a backend framework ensures lightweight, efficient performance. This project not only highlights the power of predictive analytics but also emphasizes the importance of user-centric design. Through modular development, the system is easy to maintain and extend. With further enhancements, such as image-based predictions and mobile compatibility, the project holds strong potential for future growth. Overall, it serves as a practical introduction to applied machine learning.