ENCHANTED WINGS: MARVELS OF BUTTERFLY SPECIES

1. INTRODUCTION

Project Title: Enchanted Wings: Marvels of Butterfly Species
Team Members:

Team ID – LTVIP2025TMID59407

- Sasi Kumar Palaparthi – Team Lead
- Polavarapu Ramani – Frontend Developer
- Parisa Rajeev – Tester & Data Analyst
- Pasupuleti Anand Rama Naidu – UI/UX Designer , Model Developer , & Backend Developer

---

2. PROJECT OVERVIEW

Purpose:
This project aims to develop a robust image classification system for identifying butterfly species using deep learning. It uses transfer learning with VGG16 and a Flask-based web app to deliver real-time predictions from user-uploaded images.

Features:

- Upload butterfly image and classify its species
- Display prediction confidence
- Web-based interface using Flask and Bootstrap
- Responsive layout for desktop and mobile use
- Visual performance metrics (accuracy graph)
- Integrated model training in Google Colab

---

3. ARCHITECTURE

Frontend:

- Built with HTML5, Bootstrap 5, and JavaScript
- Responsive UI for displaying prediction results and forms for uploading images
- Templates for homepage (carousel), input form, and result output

Backend:

- Flask (Python) backend
- TensorFlow/Keras used for loading and running VGG16-based classifier
- Pretrained model stored as `.h5` file and loaded into memory on app start

Database:

- No persistent storage used. All images are processed temporarily in-memory during prediction
- Option to add MongoDB in future for storing user submissions and prediction logs

---

4. SETUP INSTRUCTIONS

Prerequisites:

- Python 3.8+
- Flask
- TensorFlow
- Keras
- Google Colab (for model training)

Installation:

1. Clone the project repo
2. Place your `vgg16_butterfly_final.h5` model file in the project root
3. Install dependencies using `pip install -r requirements.txt`
4. Run `python app.py` to start the Flask server

---

5. FOLDER STRUCTURE

```
PROJECT_FOLDER/
│
├── main.py
├── vgg16_butterfly_model.keras
├── butterfly_classification.ipynb
│
├── templates/
│   ├── index.html
│   ├── result.html
├── requirements.txt
```

---

6. RUNNING THE APPLICATION

Frontend: Automatically rendered by Flask from `/templates/index.html`
Backend: Start with:

```
python app.py
```

Access the application at `http://localhost:5000`

7. API DOCUMENTATION

Endpoint: `/predict`
Method: POST
Request: Multipart Form Data (image upload)
Response: JSON with prediction result

Example:

```
{
  "prediction": "Monarch Butterfly",
  "confidence": 0.94
}
```

8. AUTHENTICATION

This version does not require user authentication. Future enhancements may include:

- User registration/login
- Token-based authentication
- Prediction history for logged-in users

9. USER INTERFACE

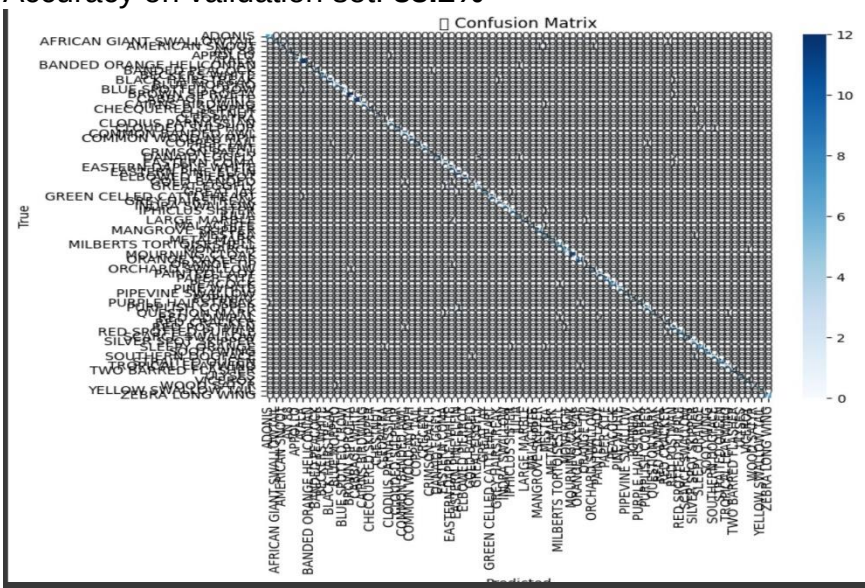
- Homepage: Carousel with “Get Started” button leading to the input form
- Input Page: Users upload a butterfly image and submit
- Output Page: Displays the predicted species and confidence

10. TESTING

- Functional Testing for all endpoints
- UI Testing for responsiveness and layout
- Model evaluation using metrics like Accuracy, Precision, Recall
- Accuracy on training set: **96.4%**

```
              accuracy                           0.88       500
             macro avg       0.90      0.88      0.87       500
          weighted avg       0.90      0.88      0.88       500
```

- Accuracy on validation set: **88.2%**

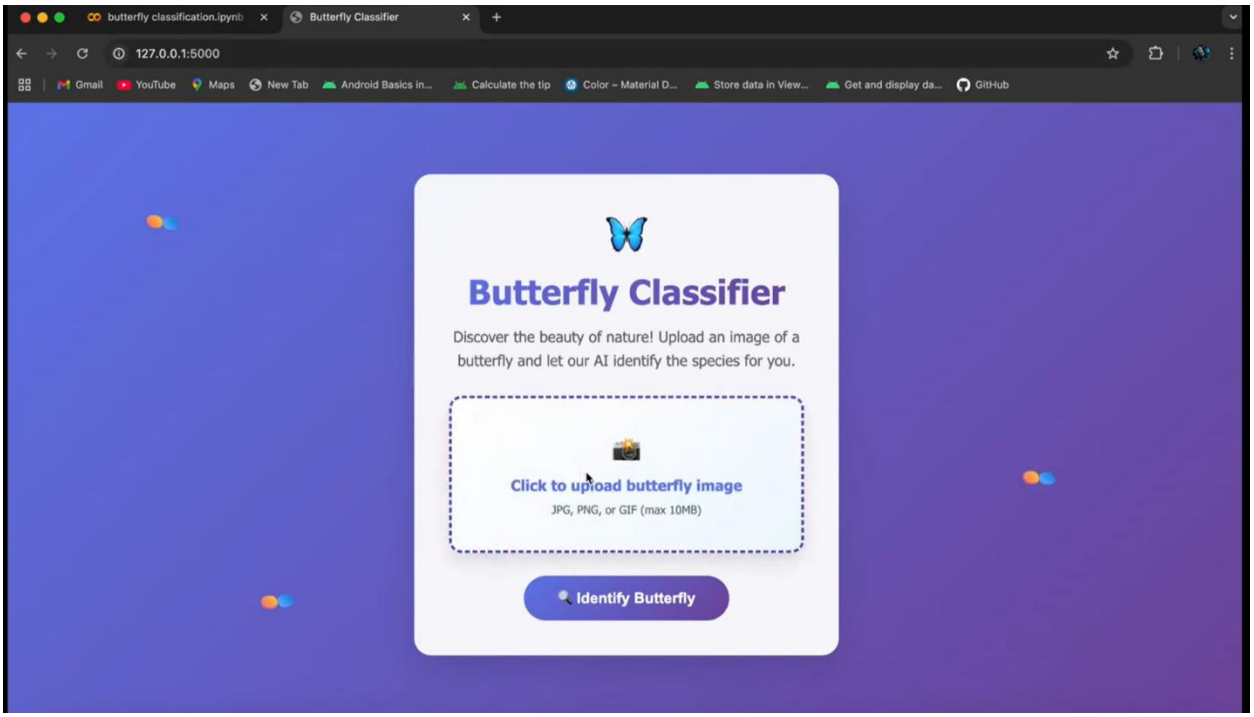



```
16/16 ━━━━━━━━━━━━━━━━━━━━ 11s 693ms/step
✅ Accuracy: 87.80%

📊 Classification Report:

                          precision    recall  f1-score   support

                  ADONIS       0.88      1.00      0.93         7
AFRICAN GIANT SWALLOWTAIL      1.00      0.80      0.89         5
          AMERICAN SNOOT       1.00      0.50      0.67         6
                   AN 88       1.00      1.00      1.00         4
                APPOLLO        1.00      0.83      0.91         6
                   ATALA       0.85      1.00      0.92        11
 BANDED ORANGE HELICONIAN      1.00      0.80      0.89         5
          BANDED PEACOCK       1.00      0.80      0.89         5
           BECKERS WHITE       1.00      1.00      1.00         7
        BLACK HAIRSTREAK       0.90      0.82      0.86        11
            BLUE MORPHO        0.86      1.00      0.92         6
       BLUE SPOTTED CROW       1.00      0.88      0.93         8
          BROWN SIPROETA       0.80      0.92      0.86        13
           CABBAGE WHITE       1.00      1.00      1.00        11
         CAIRNS BIRDWING       1.00      0.86      0.92         7
      CHECQUERED SKIPPER       1.00      0.80      0.89         5
                CHESTNUT       1.00      1.00      1.00         3
```

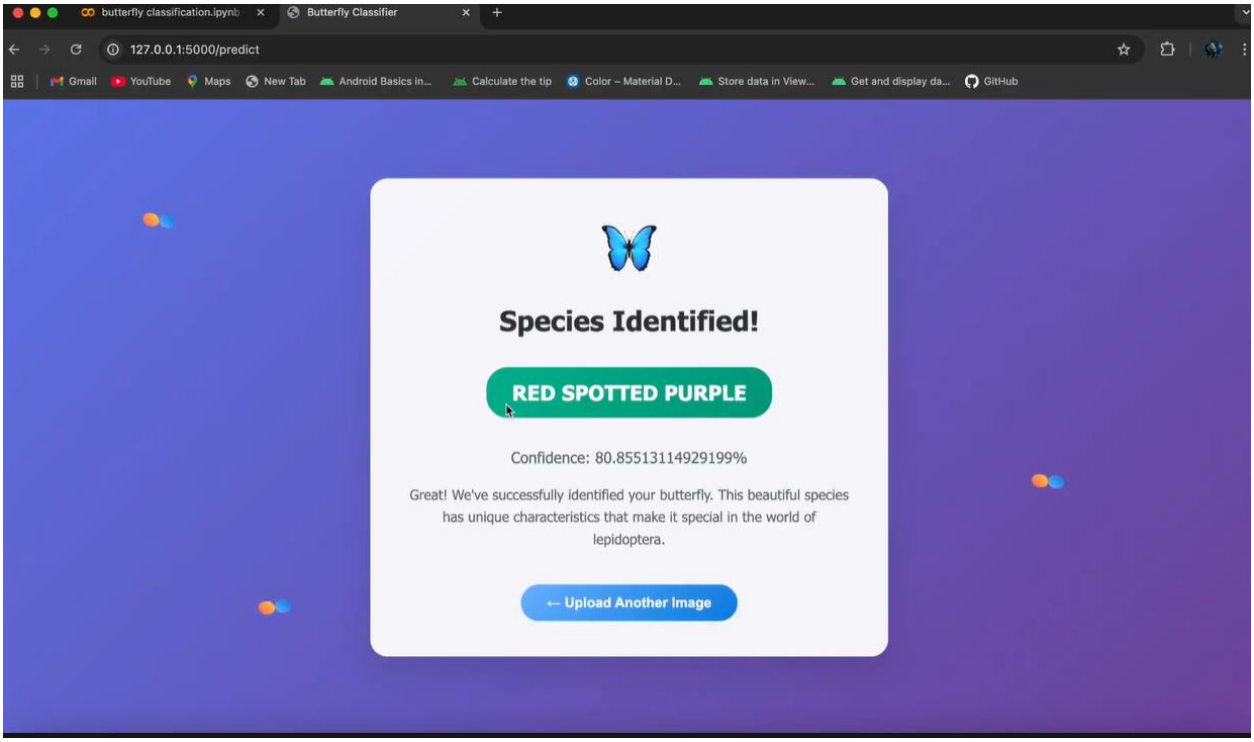
---

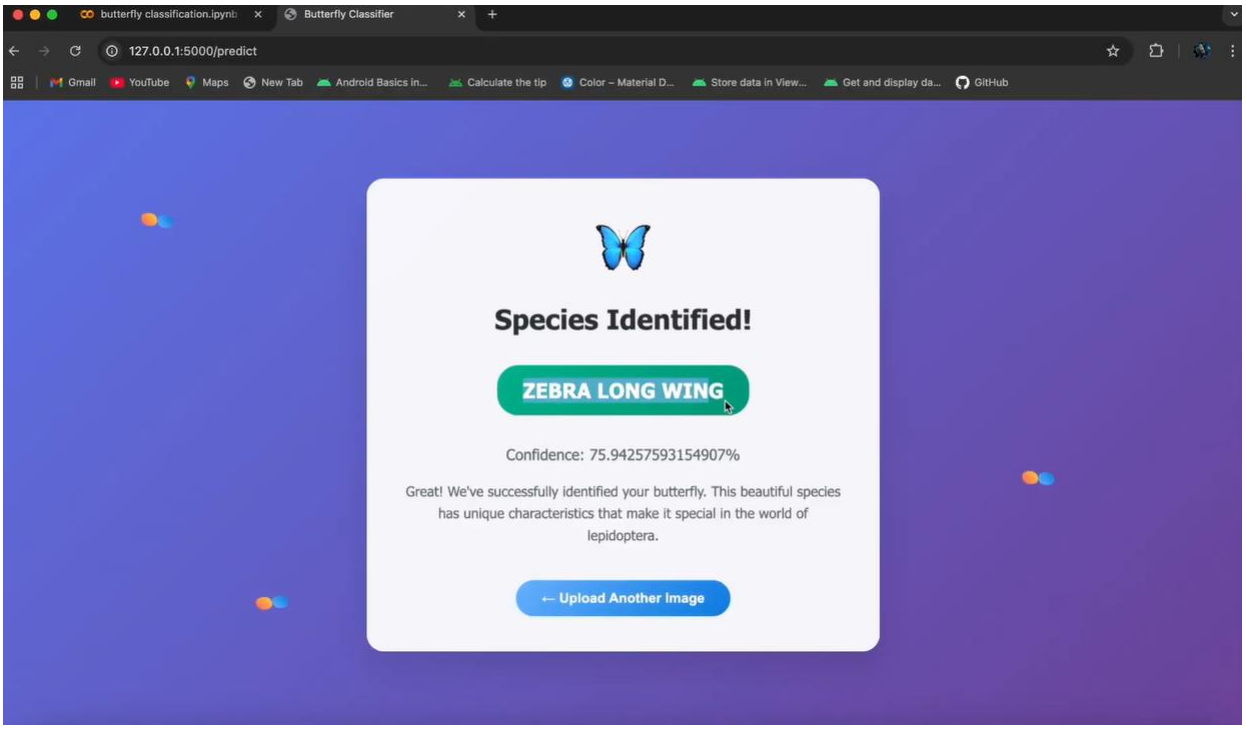
11. SCREENSHOTS OR DEMO

Homepage: Carousel layout with navigation bar



Output Page: Prediction result with formatted species name and confidence bar





Demo Video link-- https://drive.google.com/file/d/17mWkBUcEhKNNNr-gzKDizvduP7wpkjh7/view?usp=drive_link

---

12. KNOWN ISSUES

- Limited real-time camera input support
- Misclassification for rare species
- Model performance depends heavily on image quality
- Prediction is slightly slower on low-end hardware

---

13. FUTURE ENHANCEMENTS

- Add mobile camera integration
- Migrate backend to FastAPI for better performance
- Use cloud inference with Google Cloud Functions or AWS Lambda
- Add voice-based interaction (speech to species)
- Enhance dataset with community submissions
- Add visualization for prediction confidence overlaid on the image
- Include more species and retrain the model for wider accuracy

---