**Enchanted Wings: Marvels of Butterfly Species**

**Team Details**

* **Team ID:** LTVIP2025TMID35942

**Team Members:**

M Asha Sree Latha (**Team Leader)**

Bagirannagari Anil,

Basineni Bharath Kumar,

Modem Bharath

**Project Overview**

**Enchanted Wings** is a deep learning-powered web application that classifies butterfly species from uploaded images.  
The project leverages **Transfer Learning** with a pre-trained Convolutional Neural Network (CNN), coupled with a **Flask-based web application** to provide real-time predictions.

This system aims to simplify butterfly identification for biodiversity conservation, education, and hobbyist learning.

**Phase 1: Brainstorming & Ideation**

**1.1 Objective**

To develop a smart, accurate, and user-friendly application capable of recognizing butterfly species using AI.

**1.2 Problem Statement**

Manual identification of butterfly species is complex and time-consuming, requiring expert knowledge. This project solves that by providing an automated image classification solution.

**1.3 Proposed Solution**

* Utilize a **pre-trained CNN model (MobileNetV2/EfficientNet)** for transfer learning.
* Apply preprocessing steps like **resizing, normalization, and augmentation**.
* Build a **Flask API** for model inference.
* Provide a **responsive web UI** for image upload and result visualization.

**1.3.1 Dataset**

* **Data Source:** Curated dataset of butterfly species.
* **Format:** JPEG/PNG images organized into classes.
* **Files:** Training\_set.csv and Testing\_set.csv map images to species labels.
* **Preprocessing:** Images are resized to **224x224 pixels** and normalized for training.

**1.3.2 Workflow**

1. **Data Preprocessing:** Resize, normalize, augment.
2. **Model Training:** Transfer learning on a pre-trained backbone (MobileNetV2).
3. **Evaluation:** Accuracy, confusion matrix, per-class performance.
4. **Model Export:** .keras or .h5 format with class mapping.
5. **Deployment:** Flask-based web application.

**1.4 Target Users**

| **User** | **Need** | **Benefit** |
| --- | --- | --- |
| Researchers | Quick species ID | Saves time and effort |
| Conservation NGOs | Species monitoring | Assists in biodiversity studies |
| Students | Learning & education | Provides interactive learning |
| Hobbyists | Identifying captures | Enhances photography experience |

**Expected Outcome**

* **Accurate classification (90-95% validation accuracy).**
* **Lightweight web app with real-time predictions.**
* **Deployment-ready Flask app with interactive UI.**

**Phase 2: Requirement Analysis**

**2.1 Prerequisites**

* **Languages/Frameworks:** Python 3.9+, Flask 2.0+, TensorFlow 2.11+.
* **Libraries:** NumPy, Pandas, Pillow, scikit-learn, Matplotlib.
* **Environment:** Anaconda or venv for environment management.

**2.2 Functional Requirements**

* Upload an image (JPG/PNG).
* Preprocess and feed it to the CNN.
* Return predicted class with confidence.
* Display result on the web page.

**Phase 3: Architecture & UI Design**

**3.1 Architecture**

rust

CopyEdit

User -> [Upload Image] -> Flask Server

-> Preprocessing

-> CNN Model (Transfer Learning)

-> Prediction (Class + Confidence)

-> UI Display

**3.2 Flask API**

* **/** (GET): Homepage.
* **/predict** (POST): Accepts image and returns prediction.
* **/health** (GET): Confirms model readiness.

**3.3 UI**

* Built using **HTML, CSS, and JavaScript**.
* Features **image preview, loading spinner, and responsive design**.

**Phase 4: Model Development**

* **Base Model:** MobileNetV2 / EfficientNet (ImageNet weights).
* **Custom Head:** GlobalAveragePooling → Dense(512, relu) → Dropout → Dense(75, softmax).
* **Training:** Augmentation (flip, rotation, zoom), Adam optimizer, and fine-tuning of top layers.
* **Export:** butterfly\_classifier.keras and class\_indices.json.

**Phase 5: Deployment & Testing**

* **Backend:** Flask handles image upload and prediction.
* **Testing:** Validation accuracy measurement, confusion matrix, and latency tests (1–2 sec/image).
* **Frontend:** User-friendly upload form with instant result display.

**Phase 6: Challenges & Fixes**

| **Challenge** | **Solution** |
| --- | --- |
| Class imbalance | Data augmentation |
| Similar species patterns | Fine-tuning and high-resolution inputs |
| Deployment issues | Exception handling in Flask |

**Phase 7: Future Enhancements**

1. Show top-3 predictions with probabilities.
2. Grad-CAM visualization for model explainability.
3. Cloud deployment on Render/Heroku.
4. Progressive Web App (PWA) for offline use.

**Conclusion**

The **Enchanted Wings** project delivers a powerful AI tool for butterfly species identification with a simple web interface.  
This solution is practical for both research and educational purposes.