|  |
| --- |
| **Team ID :** LTVIP2025TMID32402 |
| Project Name : Enchanted Wings: Marvels of Butterfly Species |

# Architecture

## 1. Input Layer

* Input: Butterfly images (RGB)
* Image Preprocessing:
* - Resizing images (e.g., 224×224 pixels)
* - Normalizing pixel values
* - Data augmentation techniques (rotation, zoom, horizontal flip, etc.) to improve model generalization

## 2. Feature Extraction Using Pre-trained CNN

* Model: A CNN pre-trained on ImageNet (e.g., ResNet50, VGG16, InceptionV3, EfficientNet)
* Purpose: Extracts deep features from butterfly images
* Freezing Layers: Initially, the convolutional layers of the pre-trained model are frozen to retain learned weights

## 3. Classification Head

* Global Average Pooling Layer: Reduces the dimensionality of the feature maps
* Dense (Fully Connected) Layer: Learns species-specific patterns from extracted features
* Dropout Layer: Helps in preventing overfitting by randomly deactivating neurons during training
* Output Layer:
* - Activation Function: Softmax
* - Number of Nodes: 75 (representing 75 butterfly species)
* - Output: Probability distribution across species classes

## 4. Training Setup

* Loss Function: Categorical Crossentropy
* Optimizer: Adam
* Evaluation Metrics: Accuracy, Precision, Recall, F1-Score
* Callbacks: Early stopping, learning rate reduction on plateau to improve training efficiency

## 5. Deployment (Optional)

* User Interface: A web or mobile-based app for uploading butterfly images
* Backend Service: REST API built with Flask or FastAPI
* Model Hosting: TensorFlow Serving, or similar, for real-time inference

## Flow of Architecture:

* Input Image
* ↓
* Preprocessing and Augmentation
* ↓
* Pre-trained CNN (e.g., ResNet50)
* ↓
* Global Average Pooling
* ↓
* Dense + Dropout Layers
* ↓
* Softmax Output Layer (75 Classes)
* ↓
* Predicted Butterfly Species
* **Architecture:**
* 