# ArtificialIntelligenceandMachine Learning

# ProjectDocumentation

## 1. Introduction

- **ProjectTitle:EnchantedWings:MarvelsofButterflySpecies**

- **TeamMembers:**
  Team Leader **: Iswarya**

  Teammember**:Dusanapudi naveen**

  Team member **:Esha bhargavi Jalluri**

  Team member : **Chandika Bala naga Vamsi**

## 2. ProjectOverview

**• Purpose:**

Thegoalofthisprojectisto developanintelligent web-basedbutterflyclassification tool thatidentifiesbutterflyspeciesfromuploadedimagesusingdeeplearning(CNN/VGG16). It aims to assist researchers, educators, and enthusiasts in accurately identifying species, promotingbiodiversityawareness,andsupportingconservationeffortsthroughan accessible AI-powered interface.

---

**• Features:**

1. **ImageUploadInterface**
   – Simpleandresponsiveformtouploadbutterflyimagesfor classification.

2. **SpeciesPredictionwithPre-trainedCNNModel**
   – Usesatraineddeeplearningmodel(e.g.,VGG16)toidentifybutterflyspeciesfrom images.

3. **PredictionOutputwithName&Insights**
   – Displayspredictedbutterflynamewithaestheticdesignandarelevantquoteorfun fact.

4. **ResponsiveMulti-PageWebInterface**
   – User-friendly,mobile-compatiblesitewithnavigationbetweenhome,input,and result pages.

5. **EducationalContenton ButterflyDiversity**
   – Includesbutterflyfacts,specieslist,andconservationmessagestospreadecological awareness.

3. Architecture–EnhancedWings:TheMarvelsofWings

🖥️Frontend:

DevelopedusingHTMLandCSS,withoptionalBootstrapforstyling. The interface includes multiple pages such as:

- Home–introducingthe projectanditspurpose
- Input–allowinguserstouploadbutterfly images
- Result–displayingthepredictedbutterflyspecies
- About–providingeducationalinformationandproject context

ThisprojectdoesnotuseframeworkslikeReact;alightweight,staticfrontendapproachwas preferred for simplicity and compatibility.

---

⚙️Backend:

PoweredbyFlask(Python),thebackendhandles:

- Imageuploadandpreprocessing
- ServingthetrainedCNN model (VGG16)
- Predictingbutterflyspeciesfromtheinputimage
- Routingbetweenwebpages

Themodelisloaded froma.h5 fileandusedin real-timeduringuser interaction.

---

🗃️Database:

Nopermanentdatabaseisintegratedinthecurrentversion.

All operations aresession-based and processed in-memory during runtime.

InthefutureintegrationwithMongoDBorPostgreSQLcouldbeaddedto:

- Loguser predictions
- Trackmostfrequentlyidentifiedspecies
- Enableuser-specifichistoryor insights

## 4. SetupInstructions

🛠️Prerequisites&Installation–EnhancedWings

---

✅Prerequisites:

- Python 3.9+
- Flask(forbackendserver)
- JupyterNotebook/Google Colab (fortraining andtestingthemodel)
- RequiredPythonLibraries:
    - tensorflow/keras(fordeeplearning)
    - numpy
    - pillow
    - flask
    - werkzeug

- gunicorn*(optionalfor deployment)*

---

InstallationInstructions:

1. ClonetheRepository

bash

CopyEdit

gitclonehttps://github.com/your-repo/enhanced-wings cd

enhanced-wings

2. SetUpVirtualEnvironment(optionalbutrecommended)

bash

CopyEdit

python-m venv venv

source venv/bin/activate #ForLinux/macOS

venv\Scripts\activate # For Windows

3. InstallRequiredLibraries

bash

CopyEdit

pipinstall -r requirements.txt

4. RuntheFlaskApp

bash

CopyEdit

pythonapp.py

Openyourbrowserandnavigateto:http://127.0.0.1:5000

## 5. FolderStructure

enhanced-wings/

├── model/

│ ├── butterfly_model.h5 #TrainedKeras/TensorFlowmodel

│ └── class_labels.json #Mappingofclassindicestobutterfly names

│

```
├──static/
│   ├──images/                    #Uploaded andbackground images
│   │ └── butterfly-bg.jpg
│   └──css/
│       └──style.css              #Optionalexternalstylesheet
│
├── templates/
│   ├──index.html                 #Homepage(Welcomepage)
│   ├──input.html                 #Fileupload/ Prediction form
│   └──output.html                #Predictionresult page
│
├── app.py                        #Flaskbackendapplication
├──requirements.txt               #Pythondependencyfile
├── README.md                     #Projectdocumentationandusageguide
└── utils.py                      #Optionalhelperfunctions(e.g.,imagehandling,class
mapping)
```

## 6. RunningtheApplication

1. **Activatevirtualenvironment(optionalbutrecommended):**

   bash

   Copycode

   ```
   python-m venv venv

   venv\Scripts\activate#OnWindows #
   ```

   OR

   ```
   sourcevenv/bin/activate#On Mac/Linux
   ```

2. **Installdependencies:**

   bash

   Copycode

pipinstall -r requirements.txt

3. **StarttheFlaskserver:**

bash

Copy code

pythonapp.py

4. **Accessyourweb application:**

cpp

Copycode

http://127.0.0.1:5000/

## 7. APIDocumentation

http://127.0.0.1:5000/

## 8. Authentication

Currently,the**EnhancedWings**applicationdoes**notimplementauthentication**,asitisa publicly accessible educational tool.

🔐**FutureAuthenticationScope:**

- Role-basedaccessfor:
  - 🔬Researchers(touploadnewspecies)
  - 🏫Teachers/Students(toviewhistoryofpredictions)
- JWTorOAuth2integrationforsecureAPI access
- Admindashboardtomanagebutterflyclassesanduserlog

## 9. UserInterface

- **Clean,multi-pagelayout(Home,Prediction,Info,About)**
- **Intuitivenavigationbar**
- **Visualbadgesforclassificationresults**
- **Inputformwithvalidationforimageupload**

**10. Testing:**

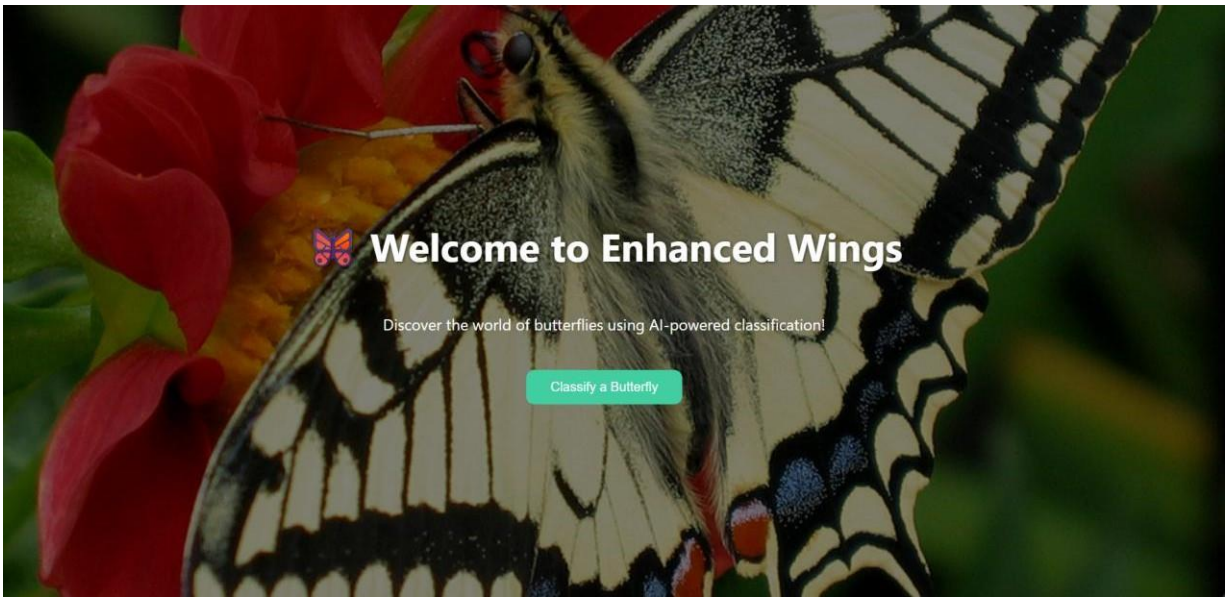
**1. ModelTesting:Usedtrain_test_split,confusionmatrix,andclassification report (via scikit-learn)**

**2. Tools:TensorFlow,Keras**

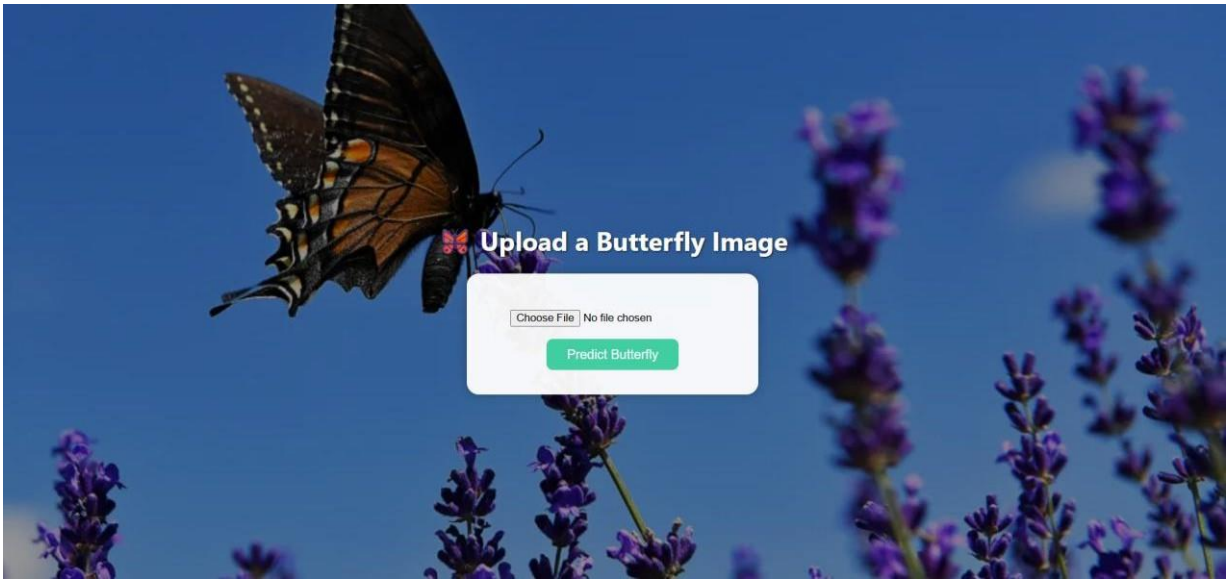
**3. UITesting:Manuallytestedpageflow,formbehavior,andprediction response**

**11. screenshotsorDemo:**

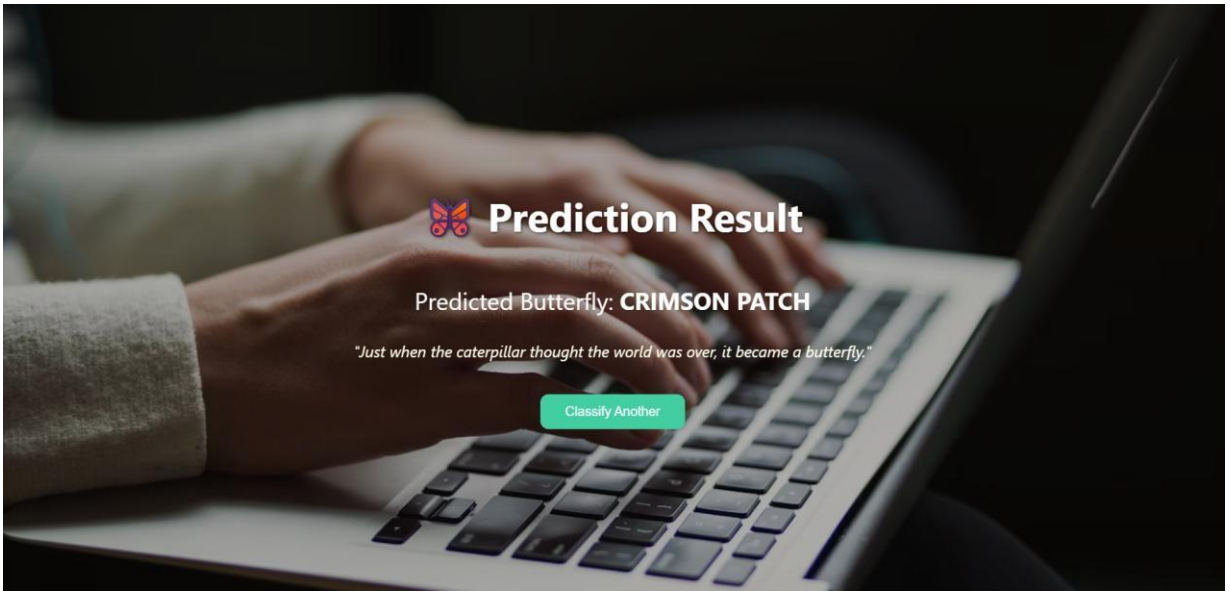
Homepage screenshot:




Predectionpagescreenshot:




PredictionResultscreenshot:




## 10. KnownIssues

- ❌ Nodatabase–predictionsare**notstored**after use
- ⚠ Basicerrorhandling–**inputvalidation**needsenhancement
- 🔓 Noauthentication–**openaccess**totheprediction

## 11. FutureEnhancements

- 🔐 Add**doctorlogin**forsecure access
- 💾 Store results in**MongoDB**forrecord-keeping
- 📄 Enable**PDFreportdownloads**afterprediction
- 🤖 Integratea**chatbot**orlaunch a **mobileversion**forwider accessibility