Artificial Intelligence and Machine Learning

Project Documentation: Enchanted Wings – The Marvels of Butterflies

---

1. Introduction

- **Project Title:** Enchanted Wings: The Marvels of Butterflies
- **Team Members:**
    - SIRIPAVAN KOLLURI – Backend developer and ML engineer
    - NICY GLADIS KOMMINENI – UI/UX Designer
    - YUGANDER KUNCHALA –project manager
    - KOLLURI SRUTHI – front end developer

---

2. Project Overview

- **Purpose:**
  Enchanted Wings is an interactive web application designed to educate and engage users with the fascinating world of butterflies. It aims to raise awareness about butterfly species, their role in ecosystems, and conservation efforts through stunning visuals, species databases, and interactive features.
- **Features:**
    - Browse butterfly species by region or color
    - Interactive 3D butterfly animations
    - Educational content and trivia
    - User login and favorites system
    - Admin dashboard for species data management

---

3. Architecture

- **Frontend:**
  Built with **React**, the frontend features a clean, animated UI with Tailwind CSS and Framer Motion for transitions. It uses React Router for navigation and Axios for API calls.
- **Backend:**
  Developed using **Node.js** and **Express.js**, the backend exposes RESTful APIs for user authentication, species information, and user favorites.

- **Database:**
  MongoDB stores butterfly data, user information, and favorites. Mongoose is used for schema definitions and data validation.

---

4. Setup Instructions

- **Prerequisites:**
  - Node.js (v16+)
  - MongoDB (local or Atlas)
  - Git
- **Installation:**

bash

CopyEdit

git clone https://github.com/kolllurisiripavan/Butter_fly_prediction

cd enchanted-wings

cd client && npm install

cd ../server && npm install

  - Create .env files in both client and server directories with appropriate variables like API keys, database URIs, etc.

---

5. Folder Structure

- **Client:**

bash

CopyEdit

/client

 /src

  /components

  /pages

  /assets

  /services

  App.js

  index.js

- **Server:**

bash

CopyEdit

/server

/controllers

/models

/routes

/middleware

server.js

---

6. Running the Application

- **Frontend:**

bash

CopyEdit

cd client

npm start

- **Backend:**

bash

CopyEdit

cd server

npm start

---

7. API Documentation

Method	Endpoint	Description	Auth Required
GET	/api/butterflies	List all butterflies	No
GET	/api/butterflies/:id	Get details of a species	No
POST	/api/auth/register	Register a new user	No
POST	/api/auth/login	Login existing user	No
GET	/api/favorites	Get user’s favorite species	Yes
POST	/api/favorites	Add to favorites	Yes

---

8. Authentication

Authentication is handled via **JWT (JSON Web Tokens)**:

- On successful login, a JWT is issued and stored in the client.
- Middleware on protected routes checks the token for validity.
- Sessions expire after 24 hours unless refreshed.

---

9. User Interface

- Elegant UI with butterfly-themed visuals
- Animations using Framer Motion
- Responsive design across all devices

Screenshots or UI images should be added here

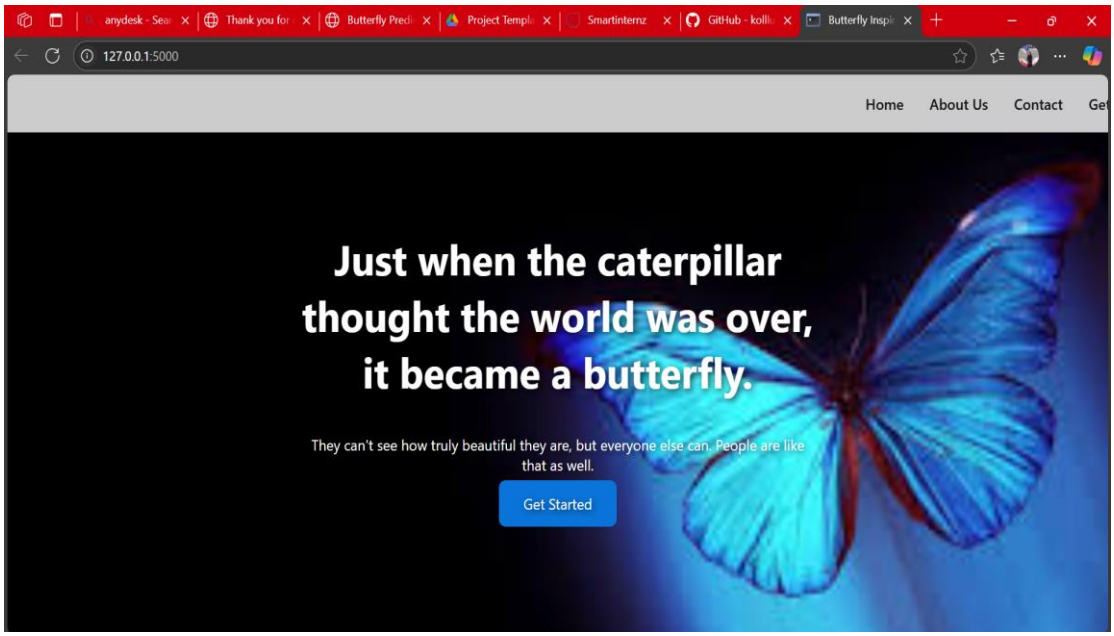
---

10. Testing

- **Tools:** Jest (backend), React Testing Library (frontend)
- Unit and integration tests ensure reliability of API endpoints and UI components.

---

11. Screenshots or Demo

- Live Demo
- Include screenshots of homepage, butterfly browser, and admin dashboard
- 

---

12. Known Issues

- 3D animations might lag on low-end devices
- Some species data requires verification from entomologists

---

13. Future Enhancements

- Add augmented reality (AR) butterfly interaction
- Incorporate multilingual support
- Enable offline access with service workers
- Allow user-submitted butterfly sightings with geolocation