**📘 Full Stack MERN Project Documentation Format**

**🟦 1. Cover Page**

* **Project Title:Enchanted wings of butterfly species**
* **Submitted by:**
* **Team Members:Vidya Reddy**
* Syed Mohammed Siddiq
* S Supraja
* **Team ID (if any):** **LTVIP2025TMID42176**
* **Institution / Organization:**
* **Supervisor / Guide Name:**
* **Date of Submission:**

**2. Table of Contents**

* Introduction
* Problem Statement
* Objectives
* Technology Stack
* System Architecture
* Module Description
* Database Schema
* Screenshots / UI Wireframes
* Challenges Faced
* Future Enhancements
* Conclusion

**🟦 3. Introduction**

* Brief description of the project
* Relevance and need of the project
* Scope and application area

**🟦 4. Problem Statement**

* What problem are you solving?
* Why is it important?
* Who are the intended users?

**🟦 5. Objectives**

* Clear list of goals the project aims to achieve  
  Example:
  + Allow users to register and log in
  + Enable posting and viewing of butterfly data
  + Admin management for content moderation

**🟦 6. Technology Stack**

| **Layer** | **Technology** |
| --- | --- |
| Frontend | React.js |
| Backend | Node.js + Express.js |
| Database | MongoDB |
| Styling | TailwindCSS / Bootstrap |
| Authentication | JWT / OAuth2 / Firebase Auth |
| Deployment | Vercel (Frontend) / Render, Railway or Heroku (Backend) |

**🟦 7. System Architecture**

Include an architecture diagram showing how frontend, backend, and database interact.

lua

CopyEdit

+------------+ HTTP +--------------+ DB Query +------------+

| React UI | <------------> | Node/Express | <-------------> | MongoDB |

+------------+ +--------------+ +------------+

▲

|

▼

[JWT Auth, API Calls, UI Rendering]

**🟦 8. Module Description**

Break down the system into main modules:

* **User Authentication**
  + Register/Login using JWT
* **Dashboard**
  + Display data in cards/tables
* **Admin Panel**
  + View, Delete or Modify user content
* **API Integration**
  + RESTful API endpoints for CRUD operations
* **Real-Time Features (Optional)**
  + Chat, Notification using Socket.io

**🟦 9. Database Schema (MongoDB Collections)**

Describe each schema:

javascript

CopyEdit

// Example: ButterflySpecies Schema

{

speciesName: String,

location: String,

imageUrl: String,

description: String,

postedBy: mongoose.Schema.Types.ObjectId,

createdAt: Date

}

**🟦 10. Screenshots / UI Wireframes**

Include screenshots of key interfaces:

* Login/Signup Page
* Dashboard
* Admin Panel
* Mobile Responsiveness (if applicable)

**🟦 11. Challenges Faced**

* State technical and non-technical issues faced and how you resolved them.
  + Example: Handling image upload with multer
  + Ensuring authentication persists across pages

**🟦 12. Future Enhancements**

* Suggestions for further improvement:
  + Add push notifications
  + Implement AI-based butterfly identification
  + Add multilingual support

**🟦 13. Conclusion**

* Summarize what was achieved
* Reflect on team experience and learning outcomes

**🟦 14. References**

* Mention any APIs, documentation, libraries used
* Example:
  + [MongoDB Docs](https://www.mongodb.com/docs/" \t "_new)
  + [React Documentation](https://reactjs.org/" \t "_new)
  + [Express Documentation](https://expressjs.com/" \t "_new)