Submitted By - Bhagyadeep Mahawar

◆ Project Name

BlogNest – Blogging Platform

◆ Project Description

BlogNest is a dynamic, role-based blogging platform that enables users to interact with blog content based on their assigned roles: **Reader** or **Author**. The project is built using the **Node.js** backend with **Express.js**, **MongoDB** for data persistence, and **EJS** for server-side rendering.

Authors have full control over their content, with the ability to **create**, **edit**, and **delete** their blog posts through a dedicated dashboard. On the other hand, **Readers** can explore all published blogs, read full content, and personalize their experience by adding or removing blogs from a **favorites** list.

This project emphasizes:

- **Authentication and session management**

- **Role-based access control (RBAC)**

- **Secure CRUD operations**

- A user-friendly interface and clean blog viewing experience

Whether used as a learning platform or content-sharing site, **BlogNest** offers an excellent example of how modern web applications manage content, roles, and personalized user features.

◆ Tech Stack Used

Layer	Technologies
Frontend	HTML, CSS, JavaScript, Bootstrap, EJS
Backend	Node.js, Express.js
Database	MongoDB with Mongoose
Template Engine	EJS

Storage	Cloudinary (for images)
Authentication	bcryptjs, express-session (or JWT if used)
File Upload	Multer / Multer-Cloudinary
Deployment	Render / Vercel / Railway (optional)

👤 User Roles:

◆ **Reader:**

- Can register and login

- Can **view all blog posts**

- Can **read full blog content**

- Can **add blogs to favorites**

- Can **remove blogs from favorites**

◆ **Author/Host:**

- Can register and login

- Can **create new blog posts**

- Can **edit** their own blog posts

- Can **delete** their own blog posts

✅ Key Features:

- 📃 Role-based access control (Author vs Reader)

- 📝 Blog creation with title, content, tags, and timestamps

- 💁 Author dashboard to manage blogs

- ❤️ Readers can mark blogs as favorite

- ❌ Option to remove blogs from favorites

- 🔐 User authentication with session handling

- 📦 MongoDB-based persistent storage

- 📄 Clean EJS templating for dynamic content rendering

🔷 Screenshots