

Ecorise Encourage People to Protect Environment

Mr. Jindashah Maheeb Fakir, Mr. Manmath Virbhadra Murge, Mr. Pratik Nandkumar Kapse,
Mr. Sarthak Balaji Bhujbal

Student of Diploma In Information Technology
Vishweshwarayya Institute of Engineering and Technology, Almala, Maharashtra, India.

Abstract: *This paper presents the design and development of Ecorise: A Smart Environment-Focused Web Application for Promoting Sustainable Habits, aimed at encouraging individuals to adopt eco-friendly practices in their daily lives. Traditional environmental awareness methods often fail due to lack of user engagement, limited awareness, and absence of structured guidance.*

The proposed system is developed using modern web technologies including the MERN stack (MongoDB, Express.js, React.js, Node.js) and provides an interactive platform for users to participate in daily sustainability activities. The application offers personalized eco-friendly tasks, progress tracking, and a reward-based system to motivate continuous user participation.

A key feature of the system is its task-based engagement model, where users complete real-life environmental actions such as saving water, reducing plastic usage, and planting trees. Users can upload proof of task completion, which enhances authenticity and accountability. The system also supports user authentication using secure mechanisms and maintains data efficiently using MongoDB. Additionally, Ecorise promotes a community-driven approach by allowing users to track their progress and stay motivated through rewards and engagement features. The platform aims to bridge the gap between awareness and action by converting knowledge into daily habits.

The implementation demonstrates improved user engagement, increased environmental awareness, and effective habit-building through a structured and interactive system. This solution provides a scalable and practical approach to promote sustainability among individuals and communities using digital technology..

Keywords: Environment Protection, Sustainable Development, MERN Stack, Eco-Friendly Tasks, Habit Tracking System, Reward System, MongoDB, React.js, Node.js, User Engagement, Green Technology

I. INTRODUCTION

I. The increasing environmental challenges such as pollution, climate change, deforestation, and excessive resource consumption have made it essential for individuals to adopt sustainable practices in their daily lives. However, despite growing awareness, many people still rely on traditional and unstructured approaches toward environmental protection, which often results in limited impact. Individuals face difficulties in identifying practical eco-friendly actions, tracking their progress, and staying motivated over time.

II. Conventional awareness methods, such as campaigns and educational programs, primarily focus on spreading knowledge but fail to convert that knowledge into consistent daily habits. Additionally, there is a lack of digital platforms that provide structured guidance, personalized eco-friendly tasks, and progress tracking mechanisms. This gap leads to low engagement and reduced participation in sustainability initiatives.

III. With the rapid advancement of digital technologies, there is a need to develop an interactive and scalable solution that can bridge the gap between awareness and action. Existing applications in the market either focus on carbon footprint tracking or general environmental information, but they lack a comprehensive system that integrates daily task-based engagement, motivation through rewards, and community interaction.

IV. To address these challenges, this paper proposes Ecorise, a smart environment-focused web application developed using modern web technologies including the MERN stack (MongoDB, Express.js, React.js, and Node.js). The system



provides a user-centric platform that encourages individuals to adopt eco-friendly habits through daily tasks, progress tracking, and reward-based motivation.

V. The application enables users to perform real-life environmental activities such as saving water, reducing plastic usage, and planting trees, while allowing them to upload proof of task completion. The system also incorporates secure user authentication, efficient data management, and a user-friendly interface to enhance usability.

VI. By digitizing and structuring environmental actions into a task-based system, Ecorise improves user engagement, promotes sustainable behavior, and reduces the gap between environmental awareness and practical implementation. This solution offers a scalable and effective approach for individuals and communities to contribute toward environmental conservation through consistent daily efforts.

II. LITERATURE REVIEW

The growing concern for environmental sustainability has led to the development of various digital platforms aimed at promoting eco-friendly practices and raising awareness among individuals. Several existing systems focus on environmental monitoring, carbon footprint tracking, and information dissemination. However, many of these platforms fall short in ensuring continuous user engagement and practical implementation of sustainable habits in daily life.

Platforms such as environmental awareness websites and mobile applications provide general information about climate change, pollution, and conservation methods. These systems are effective in spreading knowledge but often lack interactive features that encourage users to take consistent action. As a result, users may gain awareness but fail to translate it into real-life behavioral changes.

Some applications, such as carbon footprint calculators and sustainability tracking tools, allow users to measure their environmental impact. While these systems provide useful insights, they are mostly analytical and do not actively guide users with daily actionable tasks. Additionally, they often lack motivation mechanisms such as rewards, gamification, or progress tracking, which are essential for long-term habit formation.

Certain eco-friendly applications introduce challenges and tips for sustainable living. However, many of them offer limited personalization and do not provide structured task management systems. Furthermore, these platforms generally lack features like proof-based task validation, which reduces accountability and authenticity of user participation.

Social and community-based platforms encourage environmental discussions and awareness campaigns, but they do not provide a unified system that combines awareness, action, tracking, and motivation in a single platform. This fragmentation reduces their effectiveness in creating measurable environmental impact.

Based on the analysis of existing systems, it is evident that there is a gap in providing a comprehensive digital solution that not only spreads awareness but also ensures consistent user engagement through structured daily actions. The proposed system, Ecorise, addresses these limitations by integrating features such as daily eco-friendly tasks, proof-based task completion, reward and motivation system, and progress tracking within a single platform.

By combining awareness with action-oriented features, Ecorise provides a more practical and user-centric approach compared to existing generalized environmental applications. This makes it highly effective in promoting sustainable habits and encouraging individuals to actively participate in environmental conservation.

III. METHODOLOGY

The proposed system, Ecorise, is designed using a modular and scalable approach based on modern web technologies including the MERN stack (MongoDB, Express.js, React.js, and Node.js). The system follows a client-server architecture, where the frontend and backend operate independently and communicate through RESTful APIs.

I. System Architecture

The system is divided into three main layers:



1) Frontend (Client Layer):

Developed using React.js (with Next.js), responsible for user interaction and dynamic UI rendering. It provides an interactive dashboard where users can view daily eco-tasks, track progress, and upload proof of completed tasks.

2) Backend (Application Layer):

Built using Node.js and Express.js, it handles business logic, user authentication, API routing, task management, and reward calculation.

3) Database (Data Layer):

MongoDB is used as a NoSQL database to store user information, tasks, completed activities, rewards, and uploaded proof data in a flexible format.

II. Functional Modules

The system is divided into the following core modules:

4) User Management Module:

Handles user registration and login using secure authentication (JWT-based). It manages user profiles and session handling.

5) Task Management Module:

Provides daily eco-friendly tasks to users such as saving water, reducing plastic usage, and planting trees. Tasks are assigned dynamically.

6) Task Completion Module:

Allows users to mark tasks as completed and upload proof (images/videos), ensuring authenticity and accountability.

7) Reward & Gamification Module:

Awards points to users for completing tasks. It motivates continuous participation through a reward-based system.

8) Progress Tracking Module:

Tracks user performance including completed tasks, earned points, and activity streaks, helping users monitor their improvement.

9) Community/Engagement Module:

Encourages users to stay motivated by creating a sense of participation and environmental responsibility.

III. Data Flow Process

1. The user interacts with the frontend interface (React/Next.js).
2. API requests are sent to the backend using HTTP/Axios.
3. The backend validates requests using middleware (JWT authentication).
4. Business logic is executed in controllers.
5. Data is stored or retrieved from MongoDB using Mongoose models.
6. The server sends a response back to the client.
7. The frontend updates the UI dynamically.

IV. Security Mechanism

- User passwords are encrypted using hashing techniques before storage.



- Authentication is handled using JSON Web Tokens (JWT).
- Protected routes ensure that only authorized users can access specific features.

V. Media Handling Strategy

- User-uploaded proof (images/videos) is handled efficiently.
- Files are either stored locally or on cloud storage (if integrated).
- File references (URLs/paths) are stored in the database.

IV. IMPLEMENTATION

The implementation of Ecorise focuses on developing a fully functional web-based application that promotes eco-friendly habits through task-based engagement. The system is built using modern web technologies (MERN stack) and is designed to handle real-world user interactions such as task assignment, progress tracking, reward management, and user authentication.

1. Frontend Implementation

The frontend of the system is developed using React.js (with Next.js), following a component-based architecture. The user interface is designed to be interactive, responsive, and user-friendly.

- Pages: Login, Registration, Dashboard, Task List, Progress Tracking, Profile
- State Management: React Hooks such as useState and useEffect are used to manage dynamic data and API calls
- API Integration: Axios/Fetch is used to send HTTP requests to the backend
- Routing: Next.js routing is used for smooth page navigation

The frontend provides a dashboard where users can view daily eco-tasks, upload proof, and track their progress efficiently.

2. Backend Implementation

The backend is developed using Node.js and Express.js, which handle server-side operations and business logic.

- Routing: API endpoints are defined for users, tasks, rewards, and authentication
- Controllers: Handle logic such as task assignment, task completion, and reward calculation
- Middleware: Used for authentication (JWT verification) and request validation
- Error Handling: Implemented using try-catch blocks and HTTP status codes

The backend ensures secure and efficient communication between the frontend and the database.

3. Database Implementation

The system uses MongoDB with Mongoose for data modeling.

- Collections: Users, Tasks, CompletedTasks, Rewards
- Flexible Structure: Allows storing different types of eco-tasks and user activities
- Relationships: User data is linked with completed tasks and rewards using unique identifiers This structure ensures efficient storage and retrieval of dynamic environmental activity data.

4. Task Management System

The task system is the core component of the application.

- Users receive daily eco-friendly tasks
- Tasks include activities like saving water, reducing plastic use, and planting trees
- Users can mark tasks as completed
- Proof (image/video) can be uploaded for validation

The system ensures that users actively participate in real-life environmental actions.

5. Reward & Progress System

To maintain user engagement, the system includes a reward mechanism.

- Users earn points for completing tasks
- Progress is tracked based on completed tasks and points



- Activity streaks can be maintained

This gamification approach motivates users to continue using the platform regularly.

6. Authentication and Security

User authentication is implemented using JWT (JSON Web Tokens).

- On login, a token is generated and stored on the client side
- The token is sent with every protected API request
- Passwords are encrypted using hashing techniques This ensures secure access and protects user data.

7. Media Upload and Storage

The system supports uploading proof of task completion.

- Images/videos are uploaded by users
- Files are stored either locally or on cloud storage (if integrated)
- File paths/URLs are stored in the database

This feature increases authenticity and accountability of user actions.

8. User Interface and Experience

- Clean and responsive UI design
- Dashboard-based workflow for users
- Easy navigation between modules
- Simple and engaging interface

The UI is designed to provide a smooth user experience and encourage regular usage.

V. RESULTS AND DISCUSSION

The implementation of Ecorise was tested across multiple scenarios to evaluate its functionality, usability, and performance in promoting eco-friendly habits among users. The system was able to successfully handle core operations such as user authentication, task assignment, task completion with proof upload, reward allocation, and progress tracking.

1. Functional Results

The system was tested with different users to ensure proper functionality of all modules.

• Users were able to:

Register and login securely View daily eco-friendly tasks

Complete assigned tasks and upload proof (images/videos) Earn points and rewards for completed tasks

Track their progress and activity history

• System capabilities:

Dynamic task assignment Real-time progress updates

Reward calculation based on completed tasks

These results confirm that the system successfully achieves its objective of promoting sustainable habits through structured digital engagement.

2. System Performance

The application demonstrated smooth and efficient performance under normal usage conditions:

- Fast UI rendering due to React/Next.js component-based architecture
- Efficient API response handling using Node.js and Express.js
- Quick data storage and retrieval using MongoDB

The system performs effectively for small to medium-scale usage and ensures a responsive user experience.



3. Usability and User Experience

The system is designed with a user-friendly and intuitive interface:

- Dashboard-based navigation for easy access to tasks and progress
- Simple and clear task instructions for better understanding
- Easy proof upload system for task verification
- Minimal learning curve for new users

The reward system and progress tracking features increase user motivation and engagement.

4. Discussion

The results indicate that the proposed system effectively bridges the gap between environmental awareness and practical implementation. By converting awareness into daily actionable tasks, the system ensures consistent user participation.

The use of MongoDB allows flexible storage of different types of tasks and user activities. The modular architecture of the system ensures scalability and future enhancements such as advanced analytics, mobile application integration, and community features.

However, the system is currently optimized for small-scale deployment and may require further improvements in scalability, performance optimization, and advanced tracking mechanisms for large-scale usage.

VI. CONCLUSION

The development of Ecorise demonstrates an effective approach to promoting environmental sustainability through a modern web-based application. The project successfully addresses key challenges such as lack of awareness, low user engagement, and absence of structured guidance in adopting eco-friendly habits.

By implementing a MERN stack-based architecture, the system provides a scalable and efficient platform for users to participate in daily environmental activities. Features such as task-based engagement, proof-based task completion, reward and gamification system, and progress tracking significantly improve user motivation and consistency.

The system proves to be a practical solution for individuals and communities by converting environmental awareness into real-life actions. It enhances user engagement, encourages sustainable behavior, and simplifies the process of adopting eco-friendly habits in daily life.

In conclusion, the project not only achieves its intended objectives but also establishes a strong foundation for future enhancements such as mobile application development, advanced analytics, community-based features, and integration with environmental impact tracking systems. This makes Ecorise a scalable and impactful solution in the domain of environmental sustainability and green technology.

VII. ACKNOWLEDGMENT

I would like to express my sincere gratitude to my project guide for their valuable guidance, continuous support, and encouragement throughout the development of this project. Their insights and suggestions have played a significant role in shaping this work.

I am also thankful to the Head of the Department and all the faculty members of the Information Technology Department for providing the necessary resources and a supportive environment to successfully complete this project.

I extend my appreciation to my teammates for their cooperation, teamwork, and valuable contributions during the development of the Ecorise project.

Finally, I would like to thank my family and friends for their constant support, motivation, and encouragement, which helped me complete this project successfully.



REFERENCES

- [1]. MongoDB Documentation, “MongoDB Manual”. Available: <https://www.mongodb.com/docs/>
- [2]. Express.js Documentation, “Fast, unopinionated, minimalist web framework for Node.js”. Available: <https://expressjs.com/>
- [3]. React.js Documentation, “React – A JavaScript library for building user interfaces”. Available: <https://reactjs.org/>
- [4]. Node.js Documentation, “Node.js Official Documentation”. Available: <https://nodejs.org/>
- [5]. Mongoose Documentation, “MongoDB Object Modeling for Node.js”. Available: <https://mongoosejs.com/>
- [6]. Next.js Documentation, “The React Framework for Production”. Available: <https://nextjs.org/docs>
- [7]. Tailwind CSS Documentation, “Utility-first CSS framework”. Available: <https://tailwindcss.com/docs>
- [8]. JSON Web Token (JWT), “JWT Introduction”. Available: <https://jwt.io/introduction>
- [9]. Axios Documentation, “Promise-based HTTP client for the browser and Node.js”. Available: <https://axios-http.com/>
- [10]. Bcrypt Documentation, “A library to hash passwords”. Available: <https://www.npmjs.com/package/bcrypt>
- [11]. Environmental Protection Agency (EPA), “Sustainability and Environmental Protection”. Available: <https://www.epa.gov/>
- [12]. United Nations, “Sustainable Development Goals (SDGs)”. Available: <https://sdgs.un.org/goals>

