DRONACHARYA
Group of Institutions

GreenTrace

Smart Carbon Footprint Tracker with Real-Time Personal Analytics

Project Synopsis

BACHELOR OF TECHNOLOGY
Branch: CSIT

SUBMITTED BY –
SUHANI (2202300110117)
ANUBHAV (2202300110022)

UNDER THE GUIDANCE OF
Dr. Aparna Mishra
(10-2025)

Department of CSIT
Dronacharya Group of Institutions, Gr. Noida (U.P)

Table of Contents

INTRODUCTION

In today's world, where climate change and environmental degradation have become global challenges, it is essential for individuals and organizations to take conscious steps toward sustainability. Every human activity—whether it's traveling, consuming food, using electricity, or buying products—contributes to the release of greenhouse gases, primarily carbon dioxide (CO_2). However, most people are unaware of how their daily habits impact the environment. To address this issue, **GreenTrace: Smart Carbon Footprint Tracker with Real-Time Personal Analytics** has been conceptualized as an intelligent web-based solution that empowers users to **track, analyze, and reduce their carbon footprint** effectively.

The primary goal of GreenTrace is to provide a user-friendly digital platform where users can monitor their environmental impact in real time. By entering their daily activities—such as travel mode, energy usage, or diet preferences—the system calculates corresponding carbon emissions using verified **emission factor APIs**. The application then generates detailed analytics through an interactive **dashboard**, displaying insights such as total emissions, category-wise breakdown, and weekly/monthly trends. This enables users to identify the major sources of their emissions and take actionable steps toward reducing them.

A key highlight of GreenTrace is its integration of **Artificial Intelligence (AI)** to provide **personalized recommendations** for minimizing carbon emissions. The AI module analyzes user behavior patterns and suggests lifestyle adjustments—for example, using public transport instead of personal vehicles, switching to renewable energy sources, or adopting plant-based meals. By leveraging machine learning, the system continuously improves its accuracy and provides more relevant tips as it learns from user data over time.

From a technical standpoint, GreenTrace is developed using the **MERN Stack (MongoDB, Express.js, React.js, and Node.js)**, a popular full-stack JavaScript framework. The use of React.js ensures a dynamic and responsive front-end interface, while Node.js and Express.js handle the server-side logic and API integration efficiently. Data related to user activities, emissions, and recommendations are securely stored in MongoDB, providing scalability and flexibility. Additionally, **OAuth 2.0 authentication** is implemented to allow secure login via Google or other accounts, ensuring a smooth and safe user experience.

The project falls under the domain of **Environmental Informatics and Sustainable Computing**, combining computer science with environmental science to drive behavioral change through technology. Technical concepts like **data analytics**, **API integration**, and **AI-based recommendation systems** make this project both innovative and relevant to current global needs. Through GreenTrace, users not only gain awareness of their personal carbon emissions but also become active contributors to global sustainability efforts.

In summary, **GreenTrace** is more than just a tracking application—it is a **smart environmental companion** designed to promote eco-friendly living through data-driven insights. It bridges the gap between awareness and action by helping users visualize their impact, receive intelligent advice, and participate in a community leaderboard that rewards sustainable behavior. By integrating modern web technologies with environmental data, this project aims to inspire individuals and small businesses to adopt greener practices, ultimately contributing to a healthier planet.

STUDY OF EXISTING SYSTEM

1. Climatiq — Carbon Accounting API

Summary. Climatiq provides a developer-focused API and a large audited emission-factor database that returns CO_2 estimates for activity- and spend-based inputs (e.g., distance traveled by car, electricity use, product purchases). The service is designed for integration into applications that need accurate, standardized emissions calculations and is compliant with major reporting approaches (GHG Protocol).

Strengths.

- Large, scientifically-audited emission factor catalogue (coverage for many activities).
- RESTful API, well-documented, designed for real-time calculations and scaling.

Limitations / Gaps.

- Tailored for developers/enterprises; may require paid tiers for large usage.
- Provides calculations but not end-user behaviour-change tools (visualization, personalized coaching).

Relevance to GreenTrace. Climatiq is a strong option for accurate backend emission calculations in GreenTrace; combining its data with a user-facing analytics + AI recommendation layer would add behavioural value beyond raw accounting.

---

2. Carbon Interface — Developer Carbon API

Summary. Carbon Interface offers a simple API for generating emissions estimates for common activities (travel by various modes, energy use, etc.). It targets fast integration into consumer and business apps that need quick per-activity CO_2 figures.

Strengths.

- Easy-to-use endpoints for common activities, suitable for MVPs and prototypes.

Limitations / Gaps.

- Focused on single-event calculations; additional logic is needed to aggregate, visualize, and personalize advice.
- Like other APIs, it does not provide behaviour-change mechanisms or AI personalization out of the box.

Relevance to GreenTrace. Carbon Interface is useful for rapid prototyping of activity-to-emission pipelines; GreenTrace would build on this by storing historical data, running analytics, and generating personalized reduction suggestions.

---

3. Oroeco — Personal Carbon Tracking & Gamification Platform

Summary. Oroeco is a consumer app/website that tracks lifestyle activities and spending to estimate individual carbon footprints. It emphasizes education, actionable suggestions, and social/gamified features (leaderboards, points). Studies and reports describe Oroeco as an early, practical example of combining tracking + behavioural incentives.

Strengths.

- Focus on behavioural engagement: tips, comparisons, and rewards to motivate users.
- Long-running project with user-facing design patterns for habit change.

Limitations / Gaps.

- Accuracy depends on quality of user input or connected data; may lack real-time API-backed precision for some activities.
- Limited personalization beyond rules or simple recommendations.

Relevance to GreenTrace. Oroeco demonstrates the importance of gamification and social comparison — components GreenTrace should adopt — but GreenTrace can add stronger API-backed accuracy and AI-driven personalization to improve recommendation relevance.

---

4. JouleBug — Employee/Community Sustainability Engagement

Summary. JouleBug is an app designed for organizations to engage employees in sustainable actions through challenges, bite-sized tips, and progress tracking. It's used in corporate and campus contexts to nudge group behaviour.

Strengths.

- Excellent model for programmatic engagement (challenges, team-based comparisons) and measuring participation.

Limitations / Gaps.

- Focused on engagement and campaigns rather than rigorous per-activity carbon accounting.
- Typically needs organizational onboarding and customization.

Relevance to GreenTrace. JouleBug provides patterns for leaderboard and challenge mechanics GreenTrace can use to increase user retention; combining those patterns with precise emissions calculations would give both motivation and credible impact metrics.

---

5. Academic Reviews: Carbon Calculators & Recommender Systems for Sustainability

Summary. Recent academic literature surveys show two important trends: (a) carbon footprint calculators vary widely in methodology and transparency, and (b) recommender systems and AI for sustainability can support personalized guidance but are still an emerging field requiring careful evaluation for trust and effectiveness. Reviews highlight the need for standardized emission factors, transparent methodology, and behavioural-design elements to turn awareness into action.

Key Findings.

- Standardization: calculators must document emission factors and assumptions to be credible.
- Personalization + Trust: AI-driven suggestions can increase relevance, but success depends on trust, explainability, and user acceptance.

Relevance to GreenTrace. These reviews justify GreenTrace's combined approach: use audited APIs (for standardization/accuracy) **and** layer AI-based, explainable recommendations plus UX features (dashboards, leaderboards) to encourage sustained behaviour change. The literature also suggests measuring and reporting methodology in the app (transparency) to build user trust.

Conclusion — Identified Gaps & How GreenTrace Fills Them

The surveyed systems provide either (a) accurate emission data via developer APIs (Climatiq, Carbon Interface) or (b) user-engagement and behavior-change mechanics (Oroeco, JouleBug). Academic reviews emphasize the need for standardized calculations plus trustworthy personalization. **GreenTrace** can bridge these gaps by combining:

1. **API-backed, auditable emission calculations** (for accuracy),
2. **AI-powered, explainable personalized suggestions** (for relevance), and
3. **Engagement features (dashboard, leaderboards, resale marketplace)** (for sustained behaviour change and measurable impact).

FEASIBILITY STUDY

1. Introduction

The feasibility study is the initial and one of the most critical stages in the software development life cycle. It helps determine whether a proposed system is viable, practical, and beneficial for its intended users. Before investing significant time and resources, it is essential to analyze the **technical**, **economic**, and **operational feasibility** of the project. The feasibility study for **GreenTrace: Smart Carbon Footprint Tracker with Real-Time Personal Analytics** aims to evaluate whether the system can be successfully developed and implemented to help individuals and organizations track and reduce their carbon emissions efficiently.

---

2. Need for the Project

The growing concerns over **climate change**, **global warming**, and **rising carbon emissions** have made sustainability a top global priority. Many people wish to adopt eco-friendly habits but lack the tools to understand how their daily actions—such as travel choices, diet, and energy usage—affect their personal carbon footprint.

Traditional awareness campaigns or generic calculators fail to provide **personalized, data-driven insights**. Therefore, there is a strong need for a smart, interactive, and user-friendly system that can help users **quantify their environmental impact in real time** and **recommend actionable steps** to minimize it.

GreenTrace fulfils this need by offering an AI-powered, web-based platform that encourages sustainable living through measurable analytics and community participation.

---

3. Significance of the Project

GreenTrace holds significant value in both **technological innovation** and **environmental sustainability**:

- **Environmental Impact:** Promotes awareness and behavioural change by helping users visualize their carbon footprint and adopt greener habits.
- **Educational Purpose:** Acts as an educational platform that teaches individuals about emission sources and climate impact through interactive data visualization.
- **Technological Integration:** Showcases real-world application of the **MERN Stack**, API integration, and AI-based recommendation systems.
- **Community Engagement:** Features like leaderboards and challenges foster collective participation in environmental conservation.
- **AI-Driven Insights:** Provides intelligent, data-based recommendations that continuously adapt to user behaviour, making the system more engaging and effective.

This project not only supports **Sustainable Development Goal 13: Climate Action**, but also serves as a technological demonstration of how modern computing can contribute to real-world ecological problems.

---

4. Feasibility Analysis

a. Technical Feasibility

The project is technically feasible using readily available open-source technologies:

- **Frontend:** React.js for responsive UI
- **Backend:** Node.js with Express.js for API handling
- **Database:** MongoDB for flexible and scalable data storage
- **Integration:** Public emission factor APIs for accurate carbon data
- **AI Module:** Machine Learning / AI algorithms for personalized tips
  All technologies are cross-platform, well-documented, and supported by large developer communities, ensuring smooth implementation.

b. Economic Feasibility

GreenTrace can be developed using free or low-cost resources:

- Open-source MERN stack components
- Free API plans for carbon and environmental data
- Cloud hosting on platforms like Render, Vercel, or MongoDB Atlas
  Hence, the overall cost is minimal, making it economically viable for student or small-scale deployment.

c. Operational Feasibility

The system is designed to be intuitive and user-friendly, ensuring ease of use even for non-technical users. The dashboard layout, data visualization, and simple activity inputs enable users to easily track their emissions. The OAuth-based login system ensures secure access, while the leaderboard and tips encourage continuous engagement. Thus, it is operationally feasible for both individuals and small organizations.

---

5. Conclusion

After analysing the technical, economic, and operational aspects, **GreenTrace** is found to be **highly feasible and impactful**. It addresses a critical environmental challenge using accessible technologies and promotes sustainable living through intelligent analytics. The system not only serves as a valuable software solution but also contributes meaningfully toward environmental awareness and carbon reduction efforts. Therefore, the project is **both necessary and practical** to implement in the current global context.

OBJECTIVE

The main objective of the project **"GreenTrace: Smart Carbon Footprint Tracker with Real-Time Personal Analytics"** is to design and develop an intelligent web-based application that helps individuals and small organizations **monitor, analyse, and reduce their carbon footprint** through real-time data tracking and AI-driven insights.

The project aims to achieve the following specific objectives:

1. **To develop a full-stack web platform** using the **MERN (MongoDB, Express.js, React.js, Node.js)** framework that enables users to record their daily activities such as travel, energy consumption, and food habits, and automatically calculates their corresponding carbon emissions using verified APIs.
2. **To implement an AI-powered recommendation system** that analyses the user's activity data and provides personalized suggestions to help reduce carbon emissions, such as adopting energy-efficient alternatives, sustainable transport modes, and eco-friendly habits.
3. **To create an interactive user dashboard** that visually represents real-time analytics—displaying daily, weekly, and monthly emission trends—allowing users to better understand their environmental impact.
4. **To encourage sustainability through community engagement** by introducing a leaderboard and rewards system where users can compare their progress, share achievements, and motivate others to adopt greener practices.

METHODOLOGY

The development of **GreenTrace** follows a systematic and modular approach to ensure efficient design, implementation, and testing of all features. The project adopts an **Agile Development Methodology**, allowing iterative progress through continuous integration and testing. The overall goal is to build a responsive, data-driven web application that helps users calculate, track, and reduce their carbon footprint through real-time analytics and AI-powered suggestions.

◈ Project Development Steps

The following steps outline the methodology adopted to achieve the objectives of the project:

1. **Problem Identification and Research:**
   Study the global issue of carbon emissions and analyze existing carbon tracking applications. Identify their limitations and define how GreenTrace can improve user awareness and engagement using AI and analytics.
2. **Requirement Analysis:**
   Gather both functional and non-functional requirements. Define system needs such as user authentication, data visualization, API integration for emission data, and AI-based suggestion generation.
3. **System Design:**
   Develop the **system architecture**, **ER diagram**, and **data flow models** to represent the working of the system. The frontend–backend interaction is designed using RESTful APIs, and the data is structured using MongoDB collections.
4. **Module Division and Assignment:**
   Divide the project into interconnected modules. Each module is assigned to a team member based on their technical expertise.
5. **Frontend Development (React.js):**
   Design and develop a clean, responsive, and interactive user interface using React.js. Implement user dashboards, login screens, forms for data input (travel, food, energy usage), and charts for analytics.
6. **Backend Development (Node.js + Express.js):**
   Create RESTful APIs to handle data exchange between the client and the server. Implement logic for emission calculations using APIs and store user data securely in MongoDB.
7. **Database Management (MongoDB):**
   Design the database schema to store user profiles, daily activity logs, emissions data, and AI recommendations. Ensure data consistency and security using proper indexing and validation.
8. **Integration of Emission Factor APIs:**
   Connect third-party APIs to retrieve real-time emission factors for transportation, food, and energy categories. The APIs help in accurately computing the user's carbon footprint.
9. **AI Module Implementation:**
   Develop a recommendation system using machine learning or AI logic that analyzes user data patterns and provides personalized suggestions for reducing carbon emissions.
10. **Testing and Debugging:**
    Conduct functional and performance testing of all modules. Use both unit and integration testing to ensure system reliability and accuracy of calculations.

11. **Deployment and Maintenance:**
    Deploy the application on a cloud platform (such as **Render**, **Vercel**, or **AWS**) for public use. Post-deployment maintenance includes bug fixes, feature enhancements, and performance monitoring.

◈ Module Structure

Module Name	Description
User Authentication Module	Implements secure login and registration using OAuth (Google Login). Handles user sessions and access control.
Data Input Module	Allows users to enter their daily activities – travel details, energy consumption, and food habits.
Emission Calculation Module	Fetches emission factors via APIs and calculates the user's carbon footprint based on the input data.
Dashboard & Analytics Module	Displays real-time charts, graphs, and summaries using libraries like Chart.js or Recharts to visualize emissions data.
AI Recommendation Module	Uses AI/ML algorithms to suggest personalized tips for reducing emissions and improving sustainability habits.
Community Leaderboard Module	Encourages eco-friendly competition by ranking users based on their reduced carbon footprint.
Database Management Module	Manages all user data, analytics, and recommendation storage using MongoDB. Ensures scalability and security.
Admin Module	Provides administrative access for monitoring users, managing emission datasets, and updating AI logic.

◈ Team Structure and Role Distribution

Team Member	Role
Member 1 – Project lead, Backend developer & AI/ML developer	Responsible for backend logic using Node.js and Express.js, API integrations, and database management. Builds the AI-based recommendation system and integrates it with user analytics.
Member 2 – Frontend developer, tester & documentation lead	Develops the UI/UX using React.js, implements dashboards, and handles API calls from the frontend. Conducts system testing, identifies bugs, prepares project documentation, and ensures quality assurance.

FACILITIES REQUIRED FOR PROPOSED WORK

The development and deployment of the project **"GreenTrace: Smart Carbon Footprint Tracker with Real-Time Personal Analytics"** require a set of software and hardware facilities to ensure smooth implementation, testing, and performance. The following resources will be utilized during various phases of development such as design, coding, integration, and testing.

---

1. Software Requirements

Category	Software / Tool	Description
Frontend Development	React.js (JavaScript Framework)	For building dynamic, responsive, and modular user interfaces.
Backend Development	Node.js & Express.js	To handle server-side logic, routing, and API integration
Database	MongoDB (NoSQL Database)	For storing user data, activity logs, emission records, and analytics.
Authentication	OAuth 2.0 / Firebase Auth	For secure user login using Google or other social accounts.
AI / Analytics Integration	Python API / AI Model	To generate personalized emission reduction tips using AI.
API Integration	Carbon Emission Factor APIs (e.g., Climatic, Carbon Interface)	To fetch real-time emission data for various activities (travel, food, utilities).
Version Control	Git & GitHub	For project version management, collaboration, and deployment tracking.
IDE / Code Editor	Visual Studio Code	For code development, debugging, and testing.
Web Browser	Google Chrome / Edge / Firefox	For testing and previewing the web application.
Design Tools	Figma / Canva	For designing UI/UX wireframes and presentation materials.

2. Hardware Requirements

Component	Minimum Specification	Recommended Specification
Processor	Intel Core i3 or equivalent	Intel Core i5
RAM	4 GB	8 Gb or higher for faster performance
Storage	250 Gb HDD	500 GB SSD or more for quick access and storage
Operating System	Windows 10 /macOS /Linux	Latest version preferred for compatibility
Internet Connection	Stable broadband connection	Required for API calls, package installation and deployment

REFERENCES (IEEE FORMAT)

[1] Intergovernmental Panel on Climate Change (IPCC), *Climate Change 2023: Synthesis Report — Summary for Policymakers*. Geneva, Switzerland: IPCC, 2023. [Online]. Available: https://www.ipcc.ch/report/ar6/syr/

[2] M. A. Abdullah, M. S. Hossain, and G. Muhammad, "A smart IoT-based system for monitoring and controlling the carbon footprint," *IEEE Access*, vol. 8, pp. 225527–225540, 2020. doi: 10.1109/ACCESS.2020.3045478

[3] P. P. Shinde and S. S. Shah, "A study on MERN stack for web application development," *International Research Journal of Engineering and Technology (IRJET)*, vol. 7, no. 5, pp. 3529–3533, 2020.

[4] J. Smith, "Integrating AI and sustainability: Machine learning approaches to carbon footprint estimation," *IEEE Transactions on Sustainable Computing*, vol. 6, no. 3, pp. 455–464, Jul.–Sept. 2021. doi: 10.1109/TSUSC.2021.3057324

[5] P. K. Mishra, R. K. Singh, and A. Choudhary, "Green computing and sustainable software engineering: A roadmap," *IEEE Access*, vol. 9, pp. 110123–110138, 2021. doi: 10.1109/ACCESS.2021.3103504

[6] Carbon Interface API Documentation, "Carbon emission calculation APIs for developers," [Online]. Available: https://www.carboninterface.com/docs

[7] M. B. Tahir and M. H. A. Shah, "The role of artificial intelligence in achieving sustainable development goals," *Sustainability*, vol. 13, no. 24, pp. 13641–13657, Dec. 2021. doi: 10.3390/su132413641

[8] Google Developers, "OAuth 2.0 for Web Server Applications," [Online]. Available: https://developers.google.com/identity/protocols/oauth2

[9] A. K. Gupta and S. Bansal, "Personal carbon footprint estimation and awareness model using mobile applications," *Proceedings of the IEEE International Conference on Smart Technologies for Smart Nation (SmartTechCon)*, Bangalore, India, pp. 1115–1120, 2017. doi: 10.1109/SmartTechCon.2017.8358559

[10] MongoDB Inc., "MongoDB: The application data platform," [Online]. Available: https://www.mongodb.com/