

# Carbon Footprint Project Documentation

---

## Table of Contents

1. Overview
2. Scope
3. System Architecture
4. Functionalities
5. Justification of Design Choices
6. Project Timeline
7. Conclusion

## 1. Overview

The Carbon Footprint Project is designed as a web-based tool that allows users to calculate their personal carbon footprint based on daily activities such as transportation, energy consumption, diet, and waste management. The primary goal of this project is to raise awareness of carbon emissions and inspire behavioral changes towards sustainable practices.

## 2. Scope

The project targets individual users interested in understanding their environmental impact. It covers several emission sources:

- - Transportation emissions (car, public transport, flights)
- - Household emissions (electricity, natural gas, heating)
- - Diet-related emissions (meat consumption, vegetarian or vegan diets)
- - Waste management and recycling practices

## 3. System Architecture

The project backend is built using Node.js and Express to create an efficient server that manages routes, forms, and views rendered using React templates. The modular structure ensures each function is separated, simplifying maintenance and facilitating testing. Calculations for carbon footprint are encapsulated in a separate module (`carbonCalculations.js`), enhancing modularity and maintainability. The frontend interface is managed using CSS and React templates to ensure a dynamic and interactive user experience.

## 4. Functionalities

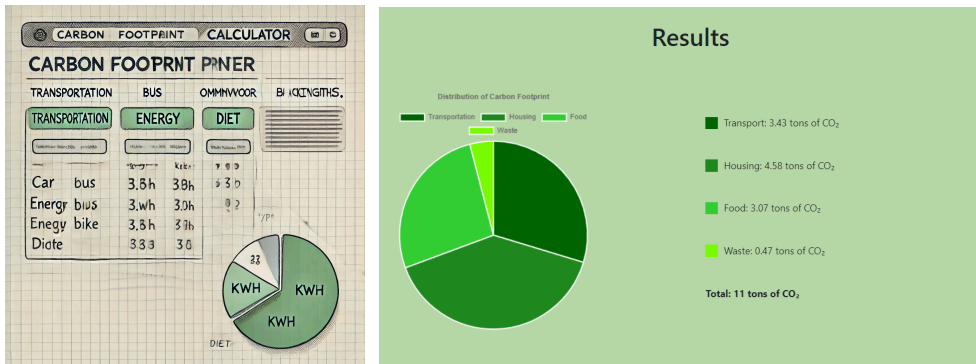
Key functionalities of the project include:

- - Real-time carbon footprint calculation based on user input
- - Modular breakdown of emissions from transport, household, diet, and waste
- - Dynamic rendering of results using EJS templates
- - Error handling for invalid input

## 5. Justification of Design Choices

The decision to use Node.js with Express was made to leverage the framework’s ability to manage routes and serve dynamic content effectively. Separating the calculation logic into `carbonCalculations.js` ensures easier debugging and future extensions. React was selected for rendering templates to provide a smooth, dynamic user experience. We aimed to create a better user experience (UX) through a simple and intuitive interface. Many carbon calculators are overly complex and hard to navigate. In contrast, our design guides users seamlessly to a detailed breakdown of their emissions upon completing the form.

## 6. Design Documents (sketches, diagrams, workflow, etc.)



## 7. Project Timeline

| Date & Time | Author | Commit Message |
|-------------|--------|----------------|
|-------------|--------|----------------|

|                     |        |                   |
|---------------------|--------|-------------------|
| 2024-11-29 14:11:51 | Omar A | Finished frontend |
|---------------------|--------|-------------------|

|                     |        |                      |
|---------------------|--------|----------------------|
| 2024-11-29 11:51:31 | Omar A | Erased deleted files |
|---------------------|--------|----------------------|

|                     |             |              |
|---------------------|-------------|--------------|
| 2024-11-29 11:21:32 | Kenzo Matoo | Parte3 Kenzo |
|---------------------|-------------|--------------|

|                     |             |              |
|---------------------|-------------|--------------|
| 2024-11-28 19:04:29 | Kenzo Matoo | Parte2 Kenzo |
|---------------------|-------------|--------------|

|                     |             |              |
|---------------------|-------------|--------------|
| 2024-11-28 18:58:58 | Kenzo Matoo | Parte1 Kenzo |
|---------------------|-------------|--------------|

| Date & Time         | Author       | Commit Message         |
|---------------------|--------------|------------------------|
| 2024-11-28 18:50:52 | Kenzo Matoo  | Parte1 Kenzo           |
| 2024-11-27 19:15:45 | AlexisVarBB  | Add files via upload   |
| 2024-11-25 20:44:53 | AlexisVarBB  | Add files via upload   |
| 2024-11-25 12:18:03 | Omar A       | Logo changed           |
| 2024-11-23 23:11:42 | Omar A       | React Backend 1st Part |
| 2024-10-30 00:19:07 | omarsanlop05 | Update README.md       |
| 2024-10-30 00:10:41 | omarsanlop05 | Update README.md       |
| 2024-10-30 00:07:18 | Omar A       | Corrected Styles       |
| 2024-10-29 23:06:08 | Omar A       | Finished project       |
| 2024-10-29 21:53:01 | Omar A       | Design corrections     |
| 2024-10-29 20:12:46 | Kenzo Matoo  | KenzoParte4            |
| 2024-10-29 19:31:35 | Kenzo Matoo  | KenzoParte3            |
| 2024-10-29 18:44:20 | Omar A       | Arreglos calculadora   |
| 2024-10-29 18:14:30 | Kenzo Matoo  | KenzoParte2            |
| 2024-10-29 12:28:46 | AlexisVarBB  | Add files via upload   |
| 2024-10-29 10:47:38 | AlexisVarBB  | Add files via upload   |
| 2024-10-29 07:07:00 | AlexisVarBB  | Add files via upload   |
| 2024-10-25 19:07:39 | Kenzo Matoo  | Kenzo parte1           |
| 2024-09-06 21:09:02 | omarsanlop05 | Initial commit         |

## 9. Personal experiences.

### Omar:

I participated in every part of the project, having different roles at each time of the project.

First delivery: Set the final details for the static initial page, such as adding a few bootstrap elements and fixing some styles.

Second delivery: connection to a mongo database, participation in backend logic and fixing final styles and changes and publication on Render.

Final delivery: main page components in React, "About" page design and implementation, fixing general style of the page, part of backend team, connection to user database and publication on Render.

### **Kenzo:**

Here are the key areas where I participated:

#### **Backend Development:**

- I designed and implemented the register, component, all the logic behind the calculator and login/Register parts
- I did all the connections between frontend and backend enabling the communication, things like the handleSubmit and HandleLogin besides fixing them i made them better

#### **Results Visualization:**

- I worked on few components to enhance the aesthetics of the website, very small things like color text or containers to make the website look better

#### **Collaborative Problem Solving:**

- I helped with the implementation of mongoose and the database of mongoDB

### **Personal Experience**

This project push me to be a lot better in programming i am truly grateful for all of this and i think i liked it a lot because of its complexity

### **Alexis:**

Here are the key areas where I participated:

#### **Calculator Component Development:**

- I designed and implemented the Calculator component, which serves as the core feature of the website, enabling users to calculate their carbon footprint.
- I ensured seamless communication with the backend by setting up the handleSubmit function to send user inputs via a POST request and display the results dynamically.

### **Results Visualization:**

- I worked on the Results component to display the calculated data using a dynamic pie chart.

### **Adding Informative Sections**

### **External Links Integration**

### **Styling and User Interface:**

- I worked on the CSS for various components, ensuring a cohesive design across sections while maintaining responsiveness and accessibility.

### **Collaborative Problem Solving:**

- I debugged and optimized parts of the project, ensuring proper state management, dynamic data rendering, and smooth backend integration.

### **Personal Experience**

This project allowed me to strengthen my skills in React, state management, and backend communication while contributing to a cause I deeply care about: environmental sustainability. Working on the calculator and results visualization was particularly rewarding as it combined technical problem-solving with a meaningful impact.

## **8. Conclusion of the Work**

### **Challenges Encountered During the Project**

- Transitioning from EJS to React: One of the main challenges was transitioning an existing EJS-based project into a React-based application. This required a complete rethinking of the structure and componentization of the project, as React's approach is centered around reusable components and state management.
- Connecting Frontend with Backend: Integrating the frontend React components with the backend for dynamic data exchange posed a challenge. Ensuring the seamless transmission of user inputs from forms and correctly processing backend responses to display the results dynamically was critical.

### **Proposed Solutions and Reasoning Behind Changes**

- Componentization and State Management: The project was refactored into reusable components such as Header, Info, Calculator, and Results. This modular approach simplified development, testing, and future scalability of the project.
- Use of Bootstrap for Styling: Bootstrap was used to ensure consistent design and responsiveness across all components. Custom CSS was added to align with the project's theme and provide visual appeal while maintaining performance.

- **Dynamic Backend Integration:** The frontend was connected to the backend using fetch for form submissions and results fetching. Error handling mechanisms were incorporated to provide feedback to the user in case of backend failures.

### **Final Reflection on the Solution**

The proposed solution effectively demonstrates how modern technologies like React and Bootstrap can be leveraged to create a dynamic and interactive web application. The modular design enables easy updates and future enhancements. Integrating data visualization allows users to better understand their carbon footprint, making the solution both informative and engaging.

### **Real-Life Application of the Solution**

This application can be used in real-world scenarios to educate individuals and organizations about their carbon footprint and encourage them to adopt more sustainable practices. By integrating resources from leading environmental organizations, the project serves as both a tool for self-assessment and a gateway to actionable steps for climate advocacy.