

Carbon Footprint Tracking Application Database Documentation

Vinay Kumar Satrasala,Mohammed Shahad

September 3, 2024

Contents

1	Introduction	2
2	NoSQL Database Schema	2
2.1	Collections and Fields	2
2.1.1	Users Collection	2
2.1.2	Carbon Emissions Collection	2
2.1.3	Emission History Collection	3
2.2	Document Relationships	3
3	NoSQL Database Schema Representation	4
3.1	Collections and Fields	4
3.2	Entity-Relationship Diagram	6
4	Database Explanation	6
4.1	Users Collection	6
4.2	Carbon Emissions Collection	6
4.3	Emission History Collection	7
4.4	Document Relationships	7
5	Conclusion	7

1 Introduction

This document provides a detailed analysis of the NoSQL database schema for the Carbon Footprint Management System. It outlines the collections, their fields, and the relationships between different documents within the collections. The goal is to offer a clear understanding of the NoSQL database structure to facilitate development, maintenance, and future enhancements.

2 NoSQL Database Schema

The database schema is designed to be flexible and scalable, typical of NoSQL databases, which are often schema-less and allow for rapid iteration and changes. Below is the detailed schema, including collection definitions, fields, and relationships.

2.1 Collections and Fields

2.1.1 Users Collection

The ‘Users’ collection stores information about individuals using the system.

- **_id** (*ObjectId*): Unique identifier for each user.
- **username** (*string*): The username of the user.
- **email** (*string*): Email address of the user.
- **password** (*string*): Encrypted password for user authentication.
- **country** (*string*): Country of residence of the user.
- **state** (*string*): State of residence of the user.

2.1.2 Carbon Emissions Collection

The ‘Carbon Emissions’ collection captures detailed carbon emissions data associated with each user.

- **_id** (*ObjectId*): Unique identifier for each emission record.
- **user_id** (*ObjectId*): Reference to the user in the ‘Users’ collection.
- **month** (*string*): The month of the emission record.
- **electricity_kwh_used** (*double*): The amount of electricity used in kWh.
- **electricity_emission** (*double*): The emission generated from electricity usage.
- **travel_public_transport** (*double*): Emissions from public transport.

- **travel_personal_vehicle** (*double*): Emissions from personal vehicle travel.
- **travel_emission** (*double*): Total travel-related emissions.
- **water_liters_used** (*double*): Amount of water used in liters.
- **water_emission** (*double*): Emission generated from water usage.
- **dietary_meat_consumption** (*double*): Meat consumption in the diet.
- **dietary_vegetable_consumption** (*double*): Vegetable consumption in the diet.
- **dietary_emission** (*double*): Emission generated from dietary habits.
- **waste_recycled_waste** (*double*): Amount of recycled waste.
- **waste_non_recycled_waste** (*double*): Amount of non-recycled waste.
- **waste_emission** (*double*): Emission generated from waste.
- **lpg_firewood_lpg_used** (*double*): Amount of LPG used.
- **lpg_firewood_firewood_used** (*double*): Amount of firewood used.
- **lpg_firewood_emission** (*double*): Emission generated from LPG and firewood usage.

2.1.3 Emission History Collection

The ‘Emission History’ collection stores historical emission data for users, enabling them to track their carbon footprint over time.

- **_id** (*ObjectId*): Unique identifier for each history record.
- **user_id** (*ObjectId*): Reference to the user in the ‘Users’ collection.
- **carbon_emissions_id** (*ObjectId*): Reference to the associated record in the ‘Carbon Emissions’ collection.
- **month** (*string*): The month of the emission history record.
- **total_emission** (*double*): The total emission recorded for that month.

2.2 Document Relationships

In NoSQL databases, relationships are often embedded or referenced rather than enforced via foreign keys as in relational databases. Below are the primary relationships:

- **Users to Carbon Emissions: One-to-Many**
 - Each user can have multiple carbon emissions records.

- **Users to Emission History:** One-to-Many
 - Each user can have multiple emission history records.
- **Carbon Emissions to Emission History:** One-to-One
 - Each emission history record references one carbon emission record.

3 NoSQL Database Schema Representation

3.1 Collections and Fields

Users

- `_id` (ObjectId)
- `username` (string)
- `email` (string)
- `password` (string)
- `country` (string)
- `state` (string)

Stores user information including personal details and authentication credentials.

Carbon Emissions

- `_id` (ObjectId)
- `user_id` (ObjectId)
- `month` (string)
- `electricity_kwh_used` (double)
- `electricity_emission` (double)
- `travel_public_transport` (double)
- `travel_personal_vehicle` (double)
- `travel_emission` (double)
- `water_liters_used` (double)
- `water_emission` (double)
- `dietary_meat_consumption` (double)
- `dietary_vegetable_consumption` (double)

- dietary_emission (double)
- waste_recycled_waste (double)
- waste_non_recycled_waste (double)
- waste_emission (double)
- lpg_firewood_lpg_used (double)
- lpg_firewood_firewood_used (double)
- lpg_firewood_emission (double)

Tracks detailed carbon emission data across different categories such as electricity, travel, water, diet, and waste management.

Emission History

- _id (ObjectId)
- user_id (ObjectId)
- carbon_emissions_id (ObjectId)
- month (string)
- total_emission (double)

Stores historical emission data to track changes over time. Each record is associated with a specific carbon emission entry.

3.2 Entity-Relationship Diagram

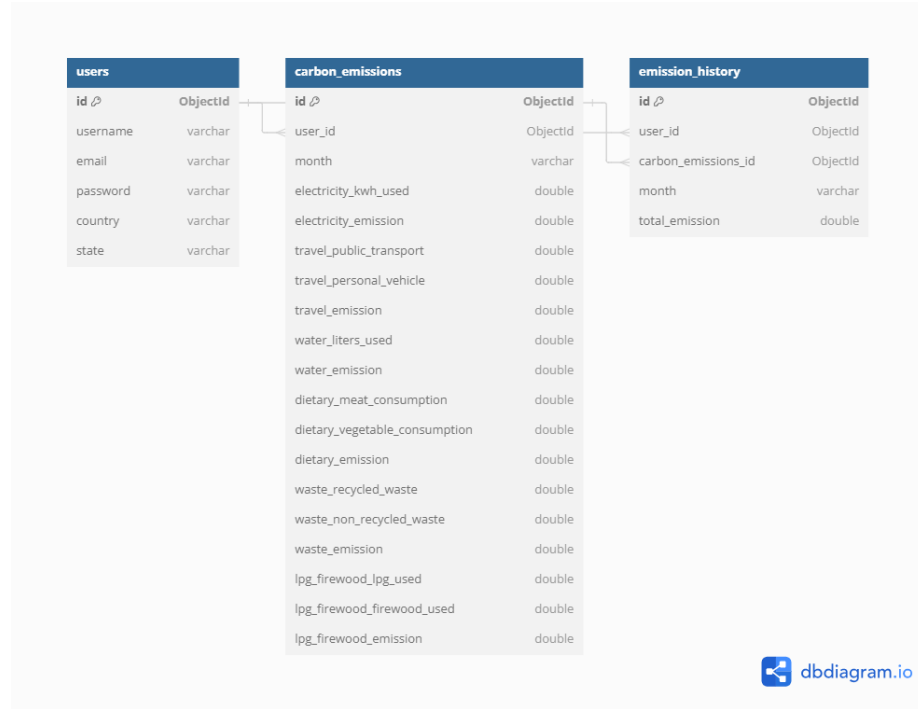


Figure 1: Entity-Relationship Diagram of the Carbon Footprint Management System

4 Database Explanation

4.1 Users Collection

The ‘Users’ collection is the core entity representing individuals who interact with the system. Each user has a unique identifier (‘`id`’), and the collection stores essential information such as username, email, password, country, and state.

4.2 Carbon Emissions Collection

The ‘Carbon Emissions’ collection records detailed carbon emissions data for each user. Each document is uniquely identified by ‘`id`’ and references the associated user via ‘`user.id`’. The collection includes fields for various sources of emissions, such as electricity usage, travel, water consumption, dietary habits, and waste management.

4.3 Emission History Collection

The ‘Emission History’ collection maintains historical records of emissions for each user, enabling them to track their carbon footprint over time. Each document references the associated user and carbon emissions record.

4.4 Document Relationships

The relationships between the documents are managed through references rather than enforced constraints. The primary relationships are as follows:

- ****One-to-Many (Users to Carbon Emissions)****: A single user can have multiple carbon emissions records.
- ****One-to-Many (Users to Emission History)****: A single user can have multiple emission history records.
- ****One-to-One (Carbon Emissions to Emission History)****: Each emission history document references one carbon emission document, ensuring accurate tracking of historical data.

5 Conclusion

This documentation outlines the NoSQL database schema for the Carbon Footprint Management System based on the provided ER diagram. The schema is designed to be flexible, scalable, and efficient, allowing for rapid development and easy modification. Each collection is structured to capture essential data points, and the relationships between documents facilitate efficient data retrieval and analysis. This approach supports the system’s objectives of tracking, analyzing, and reporting carbon emissions effectively.