# README.md

## Project Name

A web application with login, admin panel, APIs, and history tracking.

## Features

* **Login System**: Users can log in to access the application.
* **Admin Panel**: Admins can manage user accounts.
* **Weather API**: Fetch weather details by city.
* **Car API**: Fetch car details by model.
* **VIN API**: Decode Vehicle Identification Numbers (VIN).
* **History Tracking**: Logs API requests and responses.

## Setup Instructions

### Prerequisites

* Node.js (>=14.0.0)
* MongoDB instance (local or Atlas)

project-directory/

├── models/

│ ├── User.js # User model

│ ├── History.js # History model

├── views/ # EJS templates

│ ├── login.ejs # Login page

│ ├── main.ejs # Main page

│ ├── admin.ejs # Admin panel

│ ├── history.ejs # History page

│ ├── partials/ # Shared partials

│ ├── \_footer.ejs # Footer partial

├── public/ # Static assets

├── .env # Environment variables

├── server.js # Main server file

├── package.json # Dependencies

## API Endpoints

### Weather API

* **Endpoint**: /weather
* **Method**: GET
* **Parameters**: city
* **Example**: /weather?city=London

### Car API

* **Endpoint**: /cars
* **Method**: GET
* **Parameters**: model
* **Example**: /cars?model=camry

### VIN API

* **Endpoint**: /vin
* **Method**: GET
* **Parameters**: vin
* **Example**: /vin?vin=1HGCM82633A123456

### History

* **Endpoint**: /history
* **Method**: GET

## Troubleshooting

* Ensure Node.js and MongoDB are installed.
* Verify API keys in .env are valid.
* Check MongoDB connection string in .env.
* Restart the server after making changes.