Real-Time Vehicle Tracking System Documentation

Table of Contents

1. Project Overview

Title: *Real-Time Vehicle Tracking System*

Objective: To develop a system that tracks the location of vehicles in real time, providing updates and history of vehicle movements to improve fleet management, security, and operational efficiency.

Scope:

- Track multiple vehicle simultaneously.

- Provide real-time location updates.

- Store historical movement data.

- Allow users to view the location on a map.

- Generate reports on vehicle movements.

2. **Process Model** (eg. Agile or Water Fall Model – why did you chose ?)

3**. System Requirements**

- Hardware Requirements (Write short note)
    1. GPS devices installed in vehicles.
    2. Server with high availability for handling real-time data.
    3. User devices (PCs, tablets, smartphones) with internet access.

- Software Requirements (introduction of each component)
    1. Server-side: Node.js, Express.js
    2. Database: MongoDB or PostgreSQL
    3. Client-side: React.js
    4. Mapping service: Google Maps API or OpenStreetMap
    5. Communication: WebSockets for real-time updates

4. Functional Requirements

- User Authentication: Login and manage user accounts.
- Vehicle Management: Add, update, and remove vehicles from the system.

- Real-Time Tracking: Display real-time location of vehicles on a map along with the co-ordinates.
- History Tracking: Access historical movement data of vehicles.
- Alerts: Set up and receive alerts for predefined conditions (e.g., speed limits, geo-fencing).
- Reports: Generate reports on vehicle usage, travel routes, and other metrics.

5. Non-Functional Requirements

- Performance: System should handle high volumes of data and provide real-time updates with minimal latency.

- Scalability: Should be able to scale horizontally to accommodate more vehicles and users.

- Security: Ensure data encryption, secure authentication, and authorization mechanisms.

- Usability: User-friendly interface with responsive design for various devices.

- Reliability: High availability and fault tolerance to ensure continuous operation.

6. System Modules

- Authentication Module:
  - ✓ Handles user login,
    This module contains the login information of user which has already been registered.
  - ✓ registration,
    This module may contain the registration details of user like (name ,email , phone and other recovery and recognition details)
  - ✓ session management.
    This module has a special features that keeps the user information saved into the session and keeping the user logged in to the system
- Vehicle Tracking Module:
  - ✓ Processes real-time GPS data
    In this module , we takes the gps data in real time and process and check for every a killo meter.
  - ✓ Updates vehicle locations.
    In this Module , We keep the track of vehicle by real data and update the data at each location (location may have either name or latitude and longitude or both)
- History Module:
  - ✓ Stores and retrieves historical movement data.
    We keep track of total movement of data on a trip and show the history.
- Alert Module:

- ✓ Configures and sends alerts based on user-defined conditions.
  At last we send the vehicle number and location to the owner at each killo meter.

7. Implementation Details

Technologies Used:

- ✓ Backend:
  - I. Node.js
    Node.js® is a free, open-source, cross-platform JavaScript runtime environment that lets developers create servers, web apps, command line tools and scripts.
  - II. Express.js
    Express is a minimal and flexible Node.js web application framework that provides a robust set of features for web and mobile applications.

- ✓ Frontend:
  - I. React.js,
    React is a framework that employs Webpack to automatically compile React, JSX, and ES6 code while handling CSS file prefixes. React is a JavaScript-based UI development library. Although React is a library rather than a language, it is widely used in web development.
  - II. Redux

    Redux is a JS library for predictable and maintainable global state management.It helps you write applications that behave consistently, run in different environments (client, server, and native), and are easy to test. On top of that, it provides a great developer experience, such as live code editing combined with a time traveling debugger.

- ✓ Database:
  MongoDB
- ✓ Real-Time Communication: WebSockets
- ✓ Mapping: Google Maps API

CONCLUSION :

- ✓ Real time vehicle tracking system is mainly focus project on vehicle security , to keep track of location (if there is accident , it will definitely going to help on finding vehicle or people in it by current. Even if there no connection , the current location given through, will help to find.) and of course the business model . It definitely going to help people to have great time on tour and travel and increase the local business of vehicle rental system .