Smart Vehicle Monitoring and Maintenance System
Team ID - CU_CP_Team_4171

Wagh Harshal Vikram

Ranjeet Bapurao Waghmode
Hrishikesh Rajendra Garje
Gavand Abhishek Rohidas
Prasad Ganesh Dongare

Project Objectives

- Problem Statement
- Project Overview – Introduction
- Key objectives include
- End Users
- Wow Factor in Project
- Modelling/Block Diagram/Flow of Project
- Result/outcomes
- Conclusion
- Future Perspective

Problem Statement

a. **Challenge:**
   1. Managing fleet operations, ensuring vehicle health, and predicting maintenance needs.
   2. Lack of an integrated system leading to high maintenance costs, downtimes, and safety issues
b. **Solution:**
   1. A unified platform to monitor, predict, and manage vehicle health and operations.
   2. SAP integration for fleet management and predictive maintenance.

The increasing complexity of managing and maintaining a fleet of vehicles in various industries, along with the growing need for predictive maintenance, operational efficiency, and real-time safety monitoring, presents a significant challenge. Organizations face difficulties in tracking vehicle health, fuel consumption, maintenance schedules, and driver behavior effectively. The lack of a unified system that integrates vehicle health data and predicts failures before they happen leads to increased downtime, elevated maintenance costs, and higher safety risks.

This project aims to implement an IoT-based solution that integrates real-time vehicle monitoring, predictive maintenance, fleet management, and inventory management into a single cohesive platform, leveraging SAP technologies for streamlined operations. The system will utilize data from sensors installed in vehicles, such as temperature, fuel levels, and motion detection (via PIR sensors), to provide predictive insights and automate maintenance scheduling. Additionally, the system will monitor driver behavior, track fuel consumption, and optimize route efficiency.

Key objectives include:

- To integrate sensor data for real-time monitoring of vehicle components (temperature, fuel levels, etc.).
- Building a predictive maintenance model to foresee potential failures and schedule proactive maintenance.
- Optimizing fleet management by providing insights into vehicle utilization, route efficiency, and driver behavior.
- Creating a real-time dashboard that monitors vehicle status, predictive insights, and safety metrics, while also integrating features like music, navigation, and rear camera functionalities.
- The system will also improve driver safety by detecting motion around the vehicle during parking using PIR sensors, providing warning alerts through lights and a buzzer, and monitoring environmental factors like gas levels and temperature. By combining predictive maintenance with real-time data and SAP technology, this system will reduce operational costs, improve safety, and enhance decision-making, providing significant value to fleet owners.
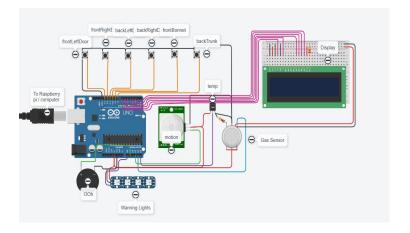
Project overview - Introduction

- Addressing the complexities x fleet management, maintenance, and safety.
- Leveraging IoT sensors, machine learning, and SAP technologies.
- Real-time monitoring, predictive maintenance, and operational efficiency.
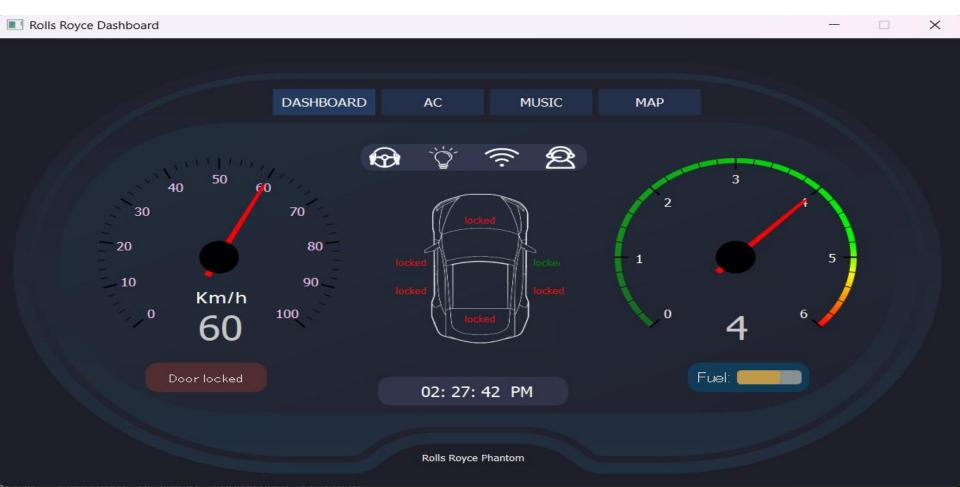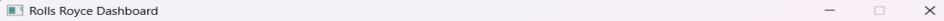
App Demonstration - Dashboard
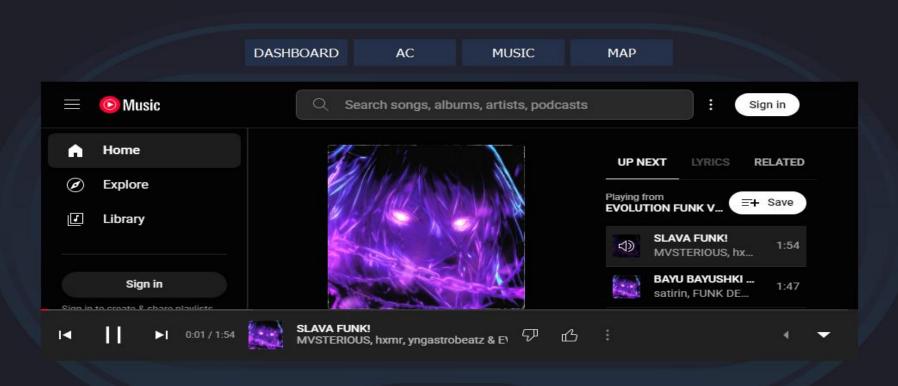
App Demonstration- AC

App Demonstration - Music
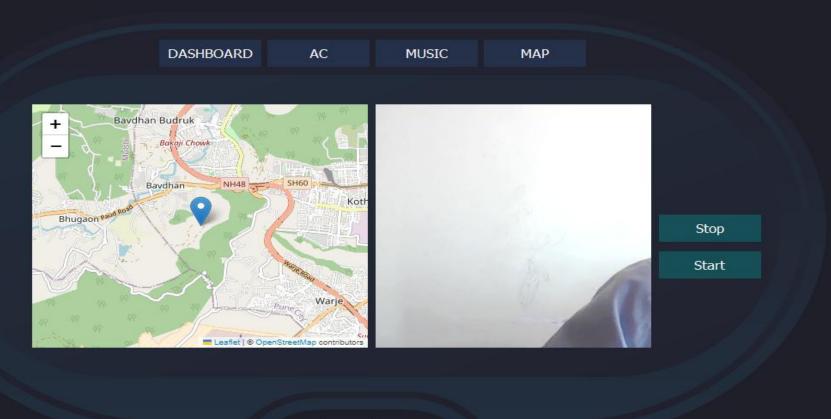
App Demonstration –Map & Rear camera
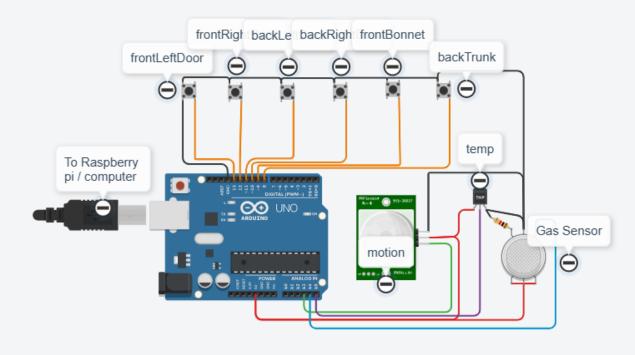
Version 2 : Prototyping and circuit diagram

edunet foundation

Proudly supported by

SAP

code unnati

TINKER CAD

arduino for car dashboard

All changes saved

Simulator time: 00:00:05.199

Code | Stop Simulation | Send To

frontRightDoor | backLeftDoor | backRightDoor | frontBonnet

frontLeftDoor

backTrunk

To Raspberry pi / computer

temp

Gas Sensor

motion

```
23   int lastButtonState[6] = {HIGH, HIGH, HIGH, HIGH, HIGH, HIGH};
24   int buttonState[6] = {HIGH, HIGH, HIGH, HIGH, HIGH, HIGH};
25
26   void setup() {
27     // Initialize serial communication
28     Serial.begin(9600);     // Serial to PC
29
30     // Initialize limit switch pins as inputs with internal pull-up
31     pinMode(frontLeftDoorPin, INPUT_PULLUP);
32     pinMode(frontRightDoorPin, INPUT_PULLUP);
33     pinMode(backLeftDoorPin, INPUT_PULLUP);
34     pinMode(backRightDoorPin, INPUT_PULLUP);
35     pinMode(frontBonnetPin, INPUT_PULLUP);
36     pinMode(backTrunkPin, INPUT_PULLUP);
37
38     // Initialize sensor pins
39     pinMode(tempPin, INPUT);   // Temperature sensor pin
40     pinMode(gasPin, INPUT);    // Gas sensor pin
41     pinMode(pirPin, INPUT);    // PIR sensor pin
42   }
43
44   void loop() {
45     // Read the temperature sensor (LM35 at A5)
46     int tempValue = analogRead(tempPin);
47     float temperatureC = (tempValue * 5.0 * 100.0) / 1024.0; // LM
48
49     // Read the gas sensor (analog reading from A4)
50     int gasValue = analogRead( gasPin);
51     bool gasDetected = (gasValue > 100);  // Threshold for detecti
52
53     // Read the PIR sensor (detect
54     bool motionDetected = digitalR                               motio
55
56     // Limit switch reading with d
57     unsigned long currentMillis =
58
59     // Read each limit switch and
60     buttonState[0] = digitalRead(f
61
```

Serial Monitor

0,0,0,0,0,0,74.71,0,0
0,0,0,0,0,0,74.71,0,0
0,0,0,0,0,0,74.71,0,0
0,0,0,0,0,0,74.71,0,0
0,0,0,0,0,0,74.71,0,0
0,0,0,0,

Send | Clear

Version 3 : Prototyping and circuit diagram