

# Project Report: OBD Data Collection and Web Serving on Raspberry Pi

## 1. Introduction

The project focuses on collecting On-Board Diagnostics (OBD) data from a vehicle using a Raspberry Pi and serving that data on a local webpage using the Apache framework. OBD data includes vital information about the vehicle's performance and health, such as fuel consumption, transmission temperature, coolant level, brake fluid level, oil pressure, and battery voltage.

## 2. Project Overview

The project involves the following key components:

- **Raspberry Pi:** The central processing unit responsible for data collection and web serving.
- **OBD Scanner:** Connected to the Raspberry Pi via Bluetooth, it provides real-time vehicle data.
- **Python Script:** The core script running on the Raspberry Pi to collect OBD data and store it in JSON format.
- **Apache Framework:** Used for serving a local webpage.
- **Webpage:** A locally hosted webpage displaying the collected OBD data in a user-friendly manner.

## 3. Hardware and Software Requirements

### Hardware:

- Raspberry Pi (with Bluetooth support)
- OBD-II Scanner
- Vehicle with OBD-II Port
- Internet Connection (for initial setup)

**Software:**

- Raspbian OS
- Python
- PyOBD Library
- Apache Web Server
- HTML/CSS/JavaScript (for webpage)
- JSON (for data storage)

## 4. Web Server Setup

- Apache web server is installed and configured on the Raspberry Pi to serve web content.
- The HTML/CSS/JavaScript files for the webpage are stored in the server's root directory.

## 5. Serving OBD Data on the Web

- The locally hosted webpage connects to the Raspberry Pi's Apache server.
- JavaScript is used to periodically request the `obd\_data.json` file.
- The retrieved data is displayed on the webpage in a user-friendly format.

## 6. Conclusion

This project successfully demonstrates the integration of OBD data collection with web serving on a Raspberry Pi. Users can access real-time vehicle information conveniently through a locally hosted webpage.

## 7. Future Enhancements

Potential future enhancements for this project include:

- Real-time data visualization (e.g., charts and graphs) on the webpage.
- Data logging for historical analysis.
- Security measures for data protection.
- Integration with external databases for data storage and retrieval.

By implementing these enhancements, the project can provide an even richer and more comprehensive user experience for vehicle data monitoring and analysis.