R25 – 034

AUTONOMOUS DRIVING

PREDICTING DRIVER BEHAVIOR AND VEHICLE MAINTENANCE USING SIMPLE DATA

OUR TEAM

Mr. H. M. Samadhi Rathnayake

Supervisor

Ms. Aparana

Co - Supervisor

Munasinghe M.M.A.D.

Rashmika K.M.G.K

Wijesingha W.M.R

Moraes V.J

INTRODUCTION

Our research focuses on using simple on-board data to predict driver behavior and vehicle maintenance needs.

We identified four key components:

1. analyzing driving styles (Driver Behavior)
2. predicting vehicle maintenance
3. studying how weather impacts driving
4. improving fuel efficiency

This work aims to create affordable, scalable solutions to enhance driving safety and efficiency.

LETS GET STARTED

RESEARCH QUESTIONS

01 Component 01

How can predict driver behavior and detect distractions in autonomous driving?

02 Component 02

Can we predict engine condition using basic sensor data like RPM, oil pressure, and temperature?

03 Component 03

How do weather, road, and traffic conditions affect driving patterns, and how can machine learning be used to detect abnormal driving behavior based on these factors?

04 Component 04

How can we predict a vehicle's fuel efficiency using its basic specifications through regression models?

RESEARCH OBJECTIVES

Sub Objective 01 - Driver Behavior

Predicting Driver Behavior and Detecting Distractions: Develop machine learning models to detect distracted states using real-time telemetry and behavioral data.

Sub Objective 02 - predicting vehicle maintenance

Predicting Engine Condition Using Basic Sensor Data: Analyze patterns in sensor data (e.g., RPM, oil pressure) to identify early signs of engine malfunctions.

Sub Objective 03 - studying how weather impacts driving

Analyzing Weather, Road, and Traffic Conditions on Driving Patterns: Train anomaly detection algorithms to flag abnormal driving patterns caused by adverse conditions.

Sub Objective 04 - improving fuel efficiency

Predicting a Vehicle's Fuel Efficiency Using Specifications: Fuel Efficiency: Use regression models to establish relationships between basic vehicle specifications and fuel consumption.

IT21269820 MUNASINGHE M.M.A.D.

DRIVER BEHAVIOR ANALYSIS

Many road crashes happen because drivers get distracted. A driver may text, talk on the phone, drink, or turn to reach something. Even a few seconds of distraction can be dangerous.

We need a system that can watch the driver and give quick feedback. If we can detect the behavior early, we can prevent mistakes before they turn into accidents.

This system is a solid foundation for future driver safety apps, enabling real-time alerts, distraction logging, and integration with Advanced Driver Assistance Systems.

METHODOLOGY

I have trained my model using the State Farm driver photos. Ten types of behavior classes, like safe driving, texting, and drinking. the photos resized each to 224 by 224 and did simple cleanup and small flips.

I tried a few models in PyTorch, like DenseNet, EfficientNet, MobileNet V3, and a Vision Transformer. DenseNet worked best for good accuracy and fast speed, so why I picked it.

I make small demo app using Kotlin to show my analysis output this app uses the phone camera to capture frames in real time. Each frame goes to our FastAPI service, which runs the PyTorch DenseNet model and sends back the label and confidence. The screen shows the live video and the current behavior,

Dataset

102,152 images, 10 Classes

Safe driving, texting, phone use, drinking, etc

Framework and Model

PyTorch , DenseNet Model

Input

Live video from the phone camera. sends them to the FastAPI server through a WebSocket.

Output

The demo app shows the live video with the current behavior, updating instantly

SYSTEM DEMO DIAGRAM

High Level Flow

Camera

(Smartphone/Webcam)

Frames

Mobile App (Android)

CameraX capture

ImageAnalyzer (base64)

WebSocket client

base64 image

Network

WebSocket

ws://<IP>:8000/ws/{client_id}

FastAPI Backend (server.py)

PyTorch DenseNet

Preprocess: resize, normalize

Output: class, behavior, confidence

image bytes

Behavior Model (behavior_model.py)

PyTorch DenseNet

Preprocess: resize, normalize

Output: class, behavior, confidence

Results

Real-time UI Display

Live feed (top)

Prediction & analyze result (left)

frame rate (fps)

DEMOSTRATION

CAPABILITIES

Present & Future

System provides a foundation for future driver safety apps, ready to add real-time alerts, event logging, and ADAS integration to reduce distractions and crashes.

What It Can Do (Today)

- Detect 10 driver behaviors from the camera.
- Show label and confidence on screen.
- Simple warning for risky actions.
- Works on common hardware.

Applications (Tomorrow)

- Driver alerts to reduce distraction.
- Event logs for safety reviews.
- Fleet dashboards and reports.
- ADAS integration for smarter assistance. give new points

IT21389924 RASHMIKA K.M.G.K.

VEHICLE MAINTENANCE PREDICTION

- Efficiency
- Methods
- Analyzing these indicators

INTRODUCTION BACKGROUND

Vehicle maintenance is essential for ensuring safety, efficiency, and longer vehicle life. However, traditional reactive maintenance methods often result in costly repairs and unexpected breakdowns. This research proposes a predictive maintenance model that uses simple and cost-effective data sources such as engine RPM, coolant pressure, lubricant temperature, coolant temperature, and temperature difference. By analyzing these indicators, the system can predict failures in advance, reduce downtime, and improve road safety..

METHODOLOGY

Requirement Analysis – Reviewed literature, conducted field visits, and identified key parameters (RPM, coolant pressure, lubricant temp, coolant temp, temp difference).

Data Acquisition – Collected real-time data via OBD-II (Bluetooth) and historical maintenance records

Data Preprocessing – Cleaned, normalized, and integrated datasets for analysis.

Model Development – Trained predictive models using machine learning to forecast failures.

System Implementation – Integrated model into a mobile app for real-time monitoring and preventive maintenance alerts.

RESEARCH PROBLEM

Traditional vehicle maintenance is mostly reactive, resulting in unexpected breakdowns, costly repairs, and safety risks. There is a lack of affordable systems that can collect and analyze real-time vehicle data such as engine RPM, coolant pressure, lubricant temperature, coolant temperature, and temperature difference to predict failures in advance. A predictive maintenance model is required to enable timely preventive maintenance, improve vehicle reliability, and reduce operational costs

MAIN OBJECTIVE

To develop a predictive maintenance model that uses real-time vehicle parameters—such as engine RPM, coolant pressure, lubricant temperature, coolant temperature, and temperature difference—to anticipate failures, enable preventive maintenance, and improve vehicle safety, reliability, and efficiency.

SUB OBJECTIVES

Data Collection – Acquire real-time vehicle parameters using an OBD-II device and collect historical maintenance records..

Data Preprocessing – Clean, normalize, and integrate the collected data for analysis.

Model Development – Train predictive models using machine learning techniques to forecast component failures

System Implementation – Integrate the predictive model into a mobile application for real-time monitoring and maintenance alerts

Validation & Evaluation – Test the model's accuracy and reliability in predicting failures using historical and live data.

SYSTEM DEMO DIAGRAM

High Level Flow

Bluetooth

Connecting

Vehicle ECU
(ELM327)

CAN / K-Line Protocol

Mobile App Data Collection Preprocessing Prediction

Mobile App

Predictive Maintenance Alerts & UI

Send

Final Dashboard

DEMOSTRATION

IT21251382 WIJESINGHA W.M.R

IMPACT OF WEATHER ON DRIVING PATTERNS

- Studies how weather and environmental factors influence human driving, offering personalized insights

- Combines weather APIs with driving data from GPS and public datasets

- Provide adaptive driving recommendations based on evolving weather conditions.

MAIN OBJECTIVES

- To design and implement a mobile-based system that integrates vehicle internal data from OBD-II and environmental data from a weather API for real-time monitoring
- To develop and train an AI-driven model that analyzes combined vehicle and environmental parameters in order to detect abnormal driving behaviors and risky conditions.
- To enhance driver safety and situational awareness by providing real-time insights, alerts, and recommendations through a user-friendly mobile application interface.

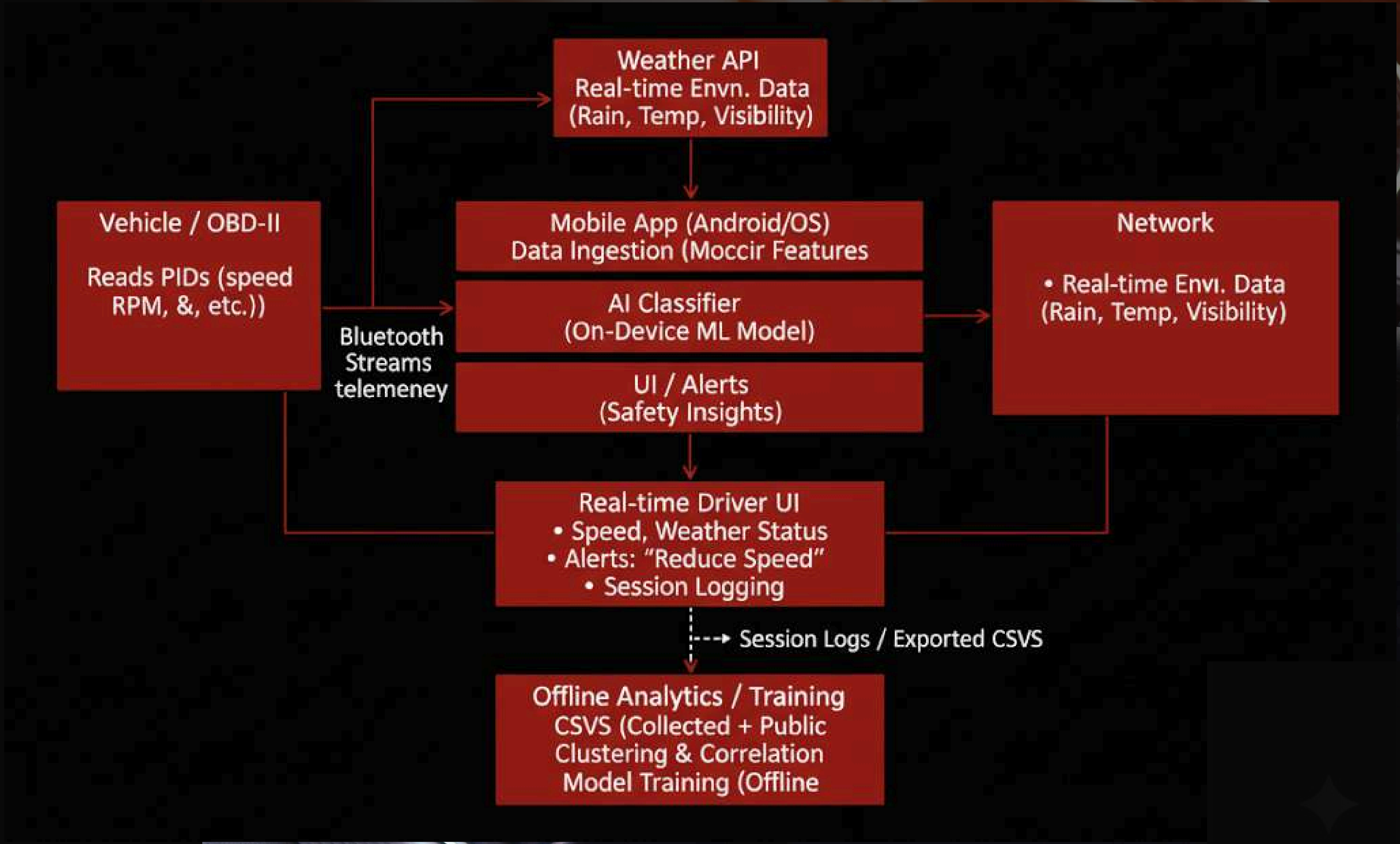
METHODOLOGY

Frameworks/Tools: Scikit-learn (Raandom Forest Classifier), Flutter (Mobile App)

Data Sources: OBD-II vehicle data (via Bluetooth), Weather API data

- Data Acquisition – Gather real-time vehicle parameters from OBD-II and environmental conditions from a weather API.
- Preprocessing – Clean and format the collected data for use in the machine learning model.
- Model Training – Train a Random Forest Classifier using combined vehicle and environmental data to classify driving behavior as normal or abnormal.
- System Integration – Integrate the trained model with the mobile app, connecting OBD-II and weather API streams.
- Application Development – Develop the mobile app using Flutter, providing a user-friendly interface to display real-time data and safety alerts.
- Testing & Validation – Test the system with real-world driving scenarios, validate model outputs, and evaluate app performance and user feedback.

SYSTEM DIAGRAM



DEMOSTRATION

IT21274534 V.J MORAES

FUEL EFFICIENCY PREDICTION AND ECONOMY

Many vehicles waste fuel due to inefficient driving behaviors like rapid acceleration, harsh braking, and idling. Small changes in driving habits can significantly reduce fuel use and emissions.

By monitoring engine RPM, vehicle speed, throttle position, coolant temperature, and fuel level, the system provides real-time feedback and actionable recommendations.

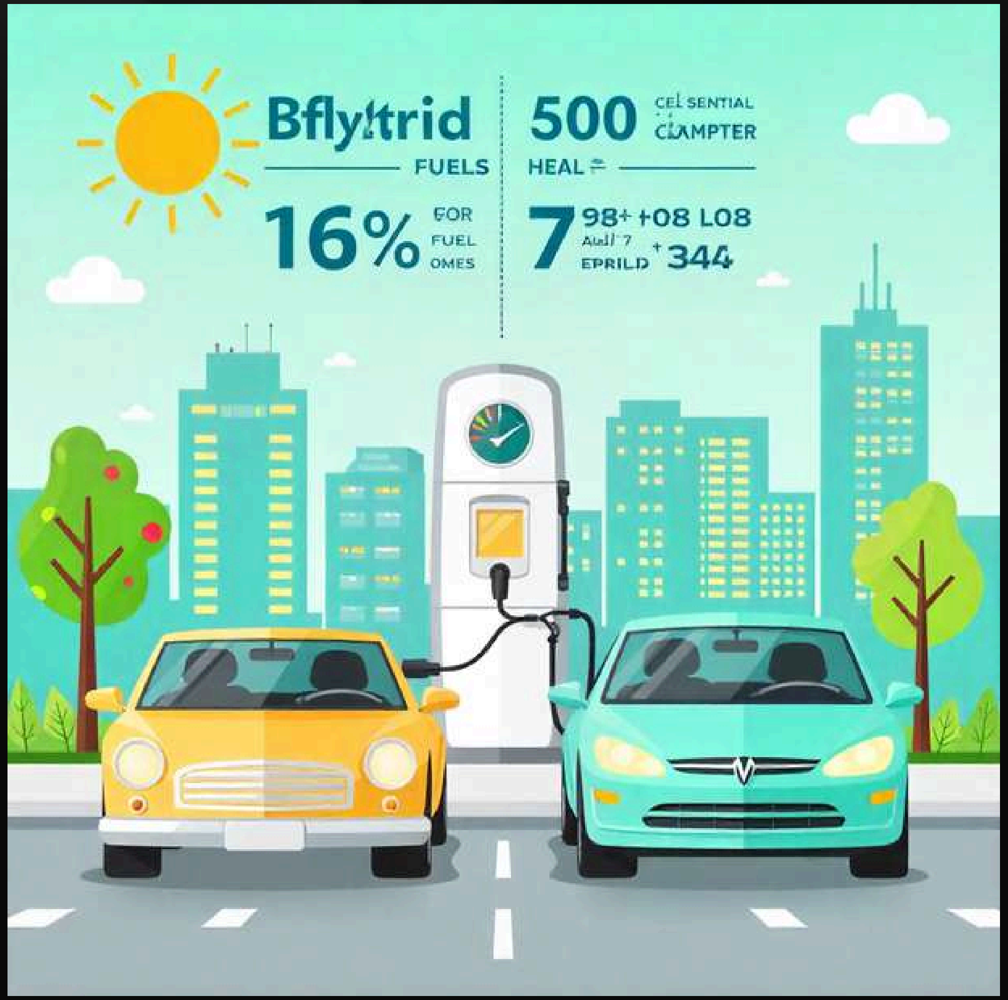
This module enables smarter driving, improving fuel efficiency and forming the basis for future eco-driving tools.

MAIN OBJECTIVES

Monitor real-time vehicle parameters: The system continuously collects data from the vehicle, such as engine RPM, speed, and throttle position, to understand driving behavior.

Predict fuel efficiency accurately: Using the collected data, the module estimates how efficiently the vehicle is consuming fuel under current conditions.

Provide actionable recommendations: Based on the predictions, the system guides drivers on how to adjust their driving to save fuel and reduce emissions.



SPECIFIC OBJECTIVES

- Collect real-time data via OBD-II and Bluetooth: This ensures accurate and up-to-date vehicle performance information.
- Analyze driving behavior and conditions: Factors like acceleration, braking, and road conditions are examined to estimate fuel usage.
- Develop prediction models: Machine learning models (regression, decision trees, random forest) are used to accurately estimate fuel consumption.

METHODOLOGY

DATA COLLECTION

- Regression and Random Forest models trained to predict fuel consumption (L/100 km).
- Features included RPM, speed, throttle %, acceleration, and coolant temperature.
- Random Forest gave the highest accuracy by handling non-linear patterns.

MODEL DEVELOPMENT

- Regression and Random Forest models trained to predict fuel consumption (L/100 km).
- Features included RPM, speed, throttle %, acceleration, and coolant temperature.
- Random Forest gave the highest accuracy by handling non-linear patterns.

MODEL DEVELOPMENT

- Predictions displayed in the app with efficiency values and actionable recommendations (e.g., “Avoid sudden acceleration,” “Maintain stable speed,” “Reduce idling”).
- Driving logs stored offline using SQLite for later analysis.

SYSTEM DEMO DIAGRAM

High Level Flow

vehicle ECU

The ECU collects engine/vehicle data (RPM, speed, throttle, coolant temp, fuel level).

OBD-II Bluetooth Adapter

OBD-II adapter extracts and streams it via Bluetooth.

Mobile App

The app receives this data in real time.

Prediction Models

Data is fed into ML models.
Regression = finds mathematical relationships.
Random Forest = handles complex, non-linear driving data.

Efficiency Estimation

predicted fuel consumption (e.g., 6.8 L/100 km).
Accuracy ~within 5% of real-world measurements.

Recommendation Engine

Detects inefficient driving patterns.
Generates actionable advice (e.g., “Avoid harsh braking”).

Driving Tips

Displays efficiency + tips in real time.

THANK YOU!

ANY QUESTIONS?