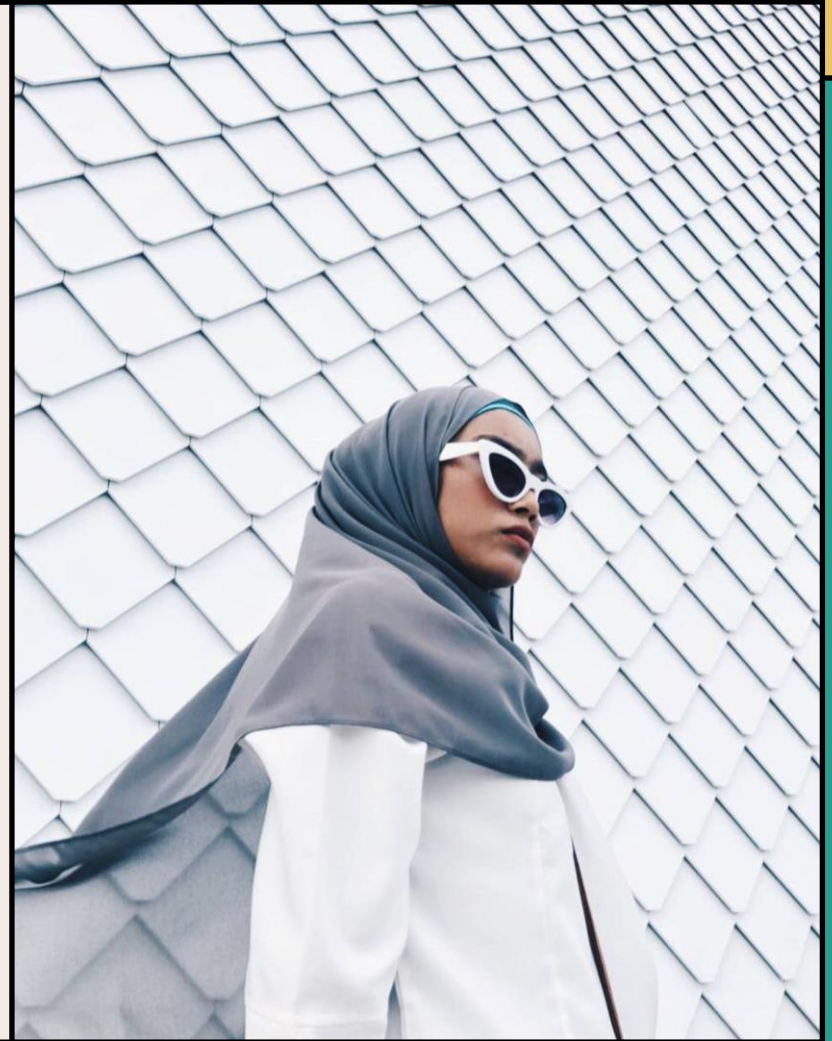


# Project Planning

FixIT: AI-Powered Vehicle Diagnostics



# Introduction

Goal: Develop a user-friendly, AI-based vehicle diagnostic system using an OBD-II reader and ESP32

Key features:

Compatibility with all OBD-II compliant vehicles (post-1996)

AI-based interpretation of DTC codes using online resources

Real-time diagnostics and user notifications

Vehicle health monitoring and preventative maintenance alerts



# How we get there

## System Architecture and Standards

### Hardware Components

- Wi-Fi OBD-II dongle
- ESP32 microcontroller
- Portable design powered by OBD-II port or internal battery

## Software and Communication

### Software and Communication

- ESP32 communicates with user's phone hotspot via Wi-Fi (IEEE 802.11)
- Secure data transmission between ESP32 and cloud (IEEE 11073)
- Cloud-based AI processing of DTC codes (SAE J1979, SAE J1850)

## End result

- Software and hardware requirements smoothly incorporated



## Project Milestones and Budget

- Requirements gathering and system design - Month 1-2
- Hardware prototyping and testing - Month 3-4
- Cloud infrastructure setup and AI model training - Month 4-5
- App development and integration - Month 5-7
- User testing and final refinements - Month 8
  
- Budget Constraints
  - Total hardware cost not to exceed \$100 per unit
  - Cost-effective solution for DIY users and casual car owners

# Risks and Mitigation Strategies

Compatibility issues with certain vehicle models

- Mitigation: Extensive testing across a wide range of OBD-II compliant vehicles

Unstable Wi-Fi connection affecting data transmission

- Mitigation: Robust error handling and data buffering in ESP32 firmware

Inaccurate AI interpretations of DTC codes

- Mitigation: Continuous model refinement using user feedback and expert input

# Thank you

Benjamin Muslic

Jonathan Duron

Mohamed Elaagip

William Griner

