openOBD

Automotive Telematics Unit

Requirements - v3.1

Authors:

- Nicholas Mulvenna
- Kaibo Ma
- Ehsan Ahmadi
- Isaiah Thiessen

Table of Contents

History of Revisions

Version	Authors	Date	Changes
1	Isaiah Thiessen Kaibo Ma Nicholas Mulvenna Ehsan Ahmadi	2016-11-28	Initial Draft
2	Nicholas Mulvenna	2017-02-07	- Removes original NS06 (encryption) and replaces it with original NS07 - Reworded C07
3	Isaiah Thiessen	2017-04-05	- Grammar, wording, and formatting
3.1	Nicholas Mulvenna	2017-04-17	- Wording and length

List of Tables and Figures

List of Technical Terms

Term	**Definition**
Internet of Things	The network of internet connecting everyday devices and appliances, enabling benefits such as home automation.
Connected Car	A vehicle enhanced with “Internet of Things” technology. [1]
Dependency Inversion Principle	A software design principle in which classes depend on abstractions. [2]
Single Responsibility Principle	A software design principle in which each class has only one responsibility which is its only reason to change. [3]
Onboard Diagnostics Port (OBD)	A standard port located under the steering wheel in North American vehicles, used to communicate with the vehicle’s internals for diagnostic purposes.

1 - Introduction

This document states the purpose, scope, and requirements for the Automotive Telematics Unit project.

1.1 - How to Use this Document

Most of the content presented here will be high level. A conceptual understanding of software and hardware design will be helpful to the reader for the last 3 sections on specifications and constraints.

Less technical readers may be directed to use the following notes:

- For an overview of the purpose of this project, see section 2: *Context & Background* on the next page.
- For coverage of the scope and goals, see sections 3: *Domain* and 4: *Goal,* starting from page 5.

Otherwise,

- Section 5: *Functional Specifications* describes the capabilities and services that must be delivered by the product (page 8).
- Section 6: *Non-Functional Specifications* discusses key attributes the product must exhibit (page 9).
- Section 7: *Constraints* lists cost, safety, and other compliance requirements (page 10).

This is a living document and changes must be recorded in the table on the previous page.

2 - Context & Background

Mojio provides a cloud software service that interfaces with connected car applications. They currently supply an automotive telematics unit that reads from the OBD-2 (On-Board Diagnostics) port of vehicles and on board sensors to feed behavioural and diagnostics data to the platform, typically done through a constant connection to a cellular network. Currently the production and design of the device is outsourced to a third-party vendor.

The primary objective given for this project is to develop a prototype telematics unit in order to give the company valuable insights into what is involved in developing an OBD device “in-house”. This device is a vital source of data for the organisation’s software platform and associated applications. Transitioning the design in-house could enable extended customisation of features to meet more customer needs, advanced modularity of design, and the potential for cost reduction for Mojio and consumers.

3 - Domain

In terms of market sectors, the Automotive Telematics Unit falls into the consumer electronics domain.

The product aims to follow Mojio's mission statement "Driven Towards a more connected world", in that enables customers to know the health and status of their vehicles at all times. [4]

This caters to a growing market of consumers who wish to be able to integrate their car into the "internet of things". Our sensor hub dongle is in a small venture for automotive technologies as there are not many competitors. This sector is expected to see major growth as consumers seek to have more and more connected devices all on a single platform. [5]

4 - Goal

In order to take advantage of and grow the “internet of things” market offerings, this product is aimed at a newer niche for connected car enthusiasts. The automotive industry has traditionally been relatively slow to adopt advances in computer technology. Many new tech companies or startups are looking into the industry as a result, and need to cater to legacy models of vehicles. This product aims to bridge the gap between legacy technology and today’s connected world, but also facilitate learning for Mojio.

Our client and partner, Mojio, seeks a prototype able to connect a large variety of cars to their cloud platform, allowing vehicles to be monitored remotely through a computer or smartphone. For example, a car owner could monitor their car’s location, speed, temperature, and health of the vehicle from their mobile devices no matter how far away they are from their car.

<u>The project’s goal is the following:</u>

> The company will have an inhouse solution and knowledge to maintain their product for future development. It will be more customizable and modular in order to further meet customer needs while remaining cost effective and efficient.

Aligned with Mojio’s purpose and mission statement, the project will ultimately improve on connected car technology, which promotes safety through data on the status and health of vehicles.

5 - Functional Specifications

With the project's goals in mind, the product is subject to meeting the following functional specifications to fulfill Mojio's user needs:

ID#	Description
FS01	The system is capable of reading and requesting data from the OBD port of a vehicle.
FS02	The system is capable of reading on-board sensors including (but not limited to): ▸ Accelerometer ▸ GPS ▸ Gyroscope ▸ Temperature
FS03	The system can output the data to a terminal window, or similar human-readable interface.
FS04	The system includes an API to facilitate interfacing with the device.
FS05	The system can return or log data in a format which is compatible with the Mojio platform.
FS06	The system has the ability to interface with the Mojio cloud platform.
FS07	The system can be powered from a vehicle's OBD port.

Table 5.0 - Functional Specifications List

The Functional Specification ID#'s used here may be referenced in other documents and are subject to the details recorded in this document only.

A feature-based approach was chosen rather than a use-case model because the project is focused on a product that will be used in further development by developers rather than being marketed directly to consumers.

6 - Non-Functional Specifications

In order to allow Mojio to take advantage of a more customizable device, the product must satisfy the following non-functional specifications including software quality attributes such as security, modularity, and portability.

ID#	Description
NS01	Software must be portable across different types of single-board computers.
NS02	Software classes must follow the dependency inversion principle.
NS03	Software classes must follow the single responsibility principle.
NS04	Software classes that are not directly related to interacting with the hardware modules (eg. sensors and GPS) must be testable without requiring the hardware modules as part of the test.
NS05	Software architecture will be able to maintain interoperability and be able to interface with legacy technology
NS06	Software and Hardware will be documented such that Mojio developers can reproduce and modify the product to further fit their needs if necessary.

Table 6.0 - Non-Functional Specifications List

The Non-Functional Specification ID#'s used here may be referenced in other documents and are subject to the details recorded in this document only.

7 - Constraints

Our product must satisfy the constraints in the table below, which help define the scope of the project in that they affect our software and hardware implementation and influence the overall design.

ID#	Description
C01	The design cannot incorporate off-the-shelf OBD protocol readers that abstract away the protocols and their complexities (e.g. no black box components).
C02	The project is constrained to UBC's provided budget ($650)
C03	The design must not violate any OBD standards in how it communicates with the protocols
C04	The designed system must be safe to use and not present any health or other risks to the user.
C05	The final prototype device must have a form factor which fits conveniently in the OBD port in a significant number of vehicles.
C06	The product, when connected to a vehicle, must not damage any of the vehicle's electronics or computer systems.
C07	The system must be convenient to adapt to hardware changes (eg new sensors or vehicle protocols) with minimal software changes.

Table 7.0: List of Constraints

The Constraint ID#'s used here may be referenced in other documents and are subject to the details recorded in this document only.

8 - References

[1]"The Open Connected Car Platform", Mojio, 2016. [Online]. Available: https://www.moj.io/connected-car-platform/. [Accessed: 27- Nov- 2016].

[2]"Dependency Inversion Principle | Object Oriented Design", Oodesign.com, 2016. [Online]. Available: http://www.oodesign.com/dependency-inversion-principle.html. [Accessed: 27- Nov- 2016].

[3]"Dissection of the Single Responsibility Principle - dylanwilson.net", Dylanwilson.net, 2016. [Online]. Available: http://www.dylanwilson.net/dissection-of-the-single-responsibility-principle. [Accessed: 27- Nov- 2016].

[4]"About Mojio, the Open Connected Car Platform", Mojio, 2016. [Online]. Available: https://www.moj.io/our-company. [Accessed: 27- Nov- 2016].

[5]J. Greenough, "How the 'Internet of Things' will impact consumers, businesses, and governments in 2016 and beyond", Business Insider, 2016. [Online]. Available: http://www.businessinsider.com/how-the-internet-of-things-market-will-grow-2014-10. [Accessed: 27- Nov- 2016].