

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

Implementación de una plataforma de conectividad para el monitoreo en tiempo real del diagnóstico de automóviles para el mantenimiento preventivo

PROYECTO INTEGRADOR

Previo la obtención del Título de:

Ingeniero en Telemática

Presentado por:

Andrea Sheyla Cárdenas Sumba

Mayken Yusselin Salavarría Tutivén

GUAYAQUIL - ECUADOR

Año: 2018

DEDICATORIA

Dedico este proyecto a Dios, a mis padres Verónica y Pedro, quiénes son el pilar fundamental en mi vida, por darme sus consejos, apoyo, ánimos para seguir adelante y creer en mí. A mis hermanos Alisson y Jordy por motivarme cada día y no dejarme vencer. A mis ángeles Douglas, Elena y Mercedes por estar conmigo y cuidarme.

A Antony por su apoyo incondicional, acompañarme en los momentos difíciles y trazarme retos.

A mis amigos, mi familia, por su confianza y ánimos, y a todas personas que compartieron sus conocimientos para lograr la finalización del proyecto.

Mayken Y. Salavarría Tutivén

DEDICATORIA

El presente proyecto se lo dedico a Dios, por permitirme culminar una etapa más de mi vida. Por respaldarme en mis éxitos y en mis fracasos y guiarme en cada paso.

A mis padres, Marta y Marco, por creer siempre en mí y apoyarme en todo momento. Su ejemplo de constancia, dedicación y humildad me motivaron a salir adelante. A mis hermanos, Cristina y Marcos, por darme la confianza de que puedo alcanzar grandes cosas cuando me las propongo.

A mis amigos y familia por estar pendientes de mí y su disposición a ayudarme cada vez que acudía a ellos.

Andrea S. Cárdenas Sumba

AGRADECIMIENTOS

Nuestro más sincero agradecimiento a la Escuela Superior Politécnica del Litoral por habernos formado como profesionales autodidactas. A nuestro tutor Ph.D José Córdova por su disponibilidad, apoyo y facilidad con los recursos necesarios para la realización del proyecto. A la Ing. María Fernanda Aguilar por guiarnos y escucharnos ante cualquier adversidad. Al Ph.D Gabriel Astudillo por su colaboración y respuesta inmediata a cualquier inquietud. A ConduEspol por su interés y disposición en el desarrollo del proyecto. A los expertos que con su amplia experiencia nos dieron consejos para la elaboración del proyecto. Y a los padres de Mayken y Sheyla por su apoyo económico.

DECLARACIÓN EXPRESA

“Los derechos de titularidad y explotación, nos corresponde conforme al reglamento de propiedad intelectual de la institución; *Andrea Sheyla Cárdenas Sumba* y *Mayken Yusselin Salavarría Tutivén* damos nuestro consentimiento para que la ESPOl realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual”

Andrea S. Cárdenas
Sumba

Mayken Y. Salavarría
Tutivén

EVALUADORES

Ph.D. Rebeca Estrada

PROFESOR DE LA MATERIA

Ph.D. José Córdova

PROFESOR TUTOR

RESUMEN

Las empresas que poseen flotas de vehículos no llevan una planificación adecuada de mantenimientos correctivos y preventivos. Al no tener un control constante sobre el estado actual de las unidades, éstas pueden presentar fallas inesperadas que alteran el desarrollo normal de las actividades. Si el problema es grave se pone en riesgo la vida del conductor y de los acompañantes o en la mayoría de los casos, representa un alto costo por reparación. El presente proyecto tiene como objetivo diseñar e implementar una plataforma de conectividad mediante la cual se pueda monitorear en tiempo real distintos parámetros de estado del vehículo obtenidos desde la unidad de control del motor a través del puerto OBD-II (disponible a partir de los modelos del año 2000).

Para su desarrollo se utilizó un dispositivo lector de códigos ELM327 de bajo costo, para la adquisición de datos del vehículo, una Raspberry pi 3 como plataforma de conectividad, y módulos GSM y LoRaWAN para la transmisión de datos de obtenidos a una plataforma destinada para el cliente.

Se obtuvo como resultado que la diferencia de tiempos promedio de recepción de datos en el servidor para las tecnologías LoRaWAN y GSM es de 2 segundos, siendo GSM de menor tiempo. Pero LoRaWAN se considera mejor ya que no existen costos fijos mensuales, los datos viajan encriptados y se tiene un monitoreo de los dispositivos implementados.

Palabras Clave: OBD-II, Parámetros de Estado, Tiempo Real, GSM, LoRaWAN.

ABSTRACT

Companies that own vehicle fleets do not make efficient planning of corrective and preventive maintenance. Without continuous monitoring of their status, unexpected vehicle failures alter normal operations of the fleet. Moreover, severe failures can be a risk for the driver and passengers or in most cases it represents a high cost of repair. The goal of the present project is the design and implementation of a connectivity platform that monitoring diagnosis of vehicles in real time obtained from the engine control unit through OBD-II port (available since 2000 years models).

For its development a low cost ELM327 code reader device was used for the acquisition of vehicle data, a Raspberry Pi 3 as a connectivity platform, and GSM and LoRaWAN modules for the transmission of obtained data to a platform destined to the client.

The results were that the average time difference of receiving data on the server for the technologies LoRaWAN and GSM is 2 seconds, being GSM of shorter time. But LoRaWAN is considered better since there are no fixed costs per month, data travels encrypted and lets monitoring of the implemented devices.

Keywords: OBD-II, diagnostics Parameter, Real Time, GSM, LoRaWAN.

ABREVIATURAS

BLE	Bluetooth Low Energy
DLC	Diagnostic Link Connector
DTC	Códigos de error/falla o Código de diagnóstico
ECM	Módulo de Control Electrónico
ECU	Engine Control Unit
ELM	Enterprise Learning Management
EPA	Agencia de Protección del Medio Ambiente.
GSM	Sistema Global para las Comunicaciones Móviles
JSON	JavaScript Object Notation
LoRa	Low Range
M2M	Machine to machine
MIL	Malfunction Indicator Light
OBD	On-Board Diagnostics.
OBD-II	On-Board Diagnostics Second Generation
SAE	Sociedad de Ingenieros Automotrices.
SD	Secure Digital.
TTN	The Things Network

SIMBOLOGÍA

bps	Bits por Segundo.
cm	Centímetros.
°C	Centígrados.
G	Gigabyte.
Hz	Hercio.
Kbps	Kilobits por segundo.
KHz	Kilohercios.
Km	Kilómetros.
m	Metros.
mA	Miliamperios.
ms	Milisegundos.
mV	Milivoltios.
V	Voltios.

ÍNDICE DE FIGURAS

Figura 1.1 Conexión entre el puerto OBD II, la plataforma de conectividad y la plataforma del cliente.....	6
Figura 2.1 Conector DLC J1962 Tipo A (izquierda) y Tipo B (derecha)	8
Figura 2.2. Terminales del conector DLC ODB-II	9
Figura 2.3. Ubicación DLC Tipo A	10
Figura 2.4. Ubicación DLC Tipo A	10
Figura 2.5. Diferentes formas en las que se presenta la MIL	14
Figura 3.1. Diagrama de bloques del sistema propuesto	22
Figura 3.2. Localización del conector DLC OBD-II	23
Figura 3.3. Interfaz de Diagnóstico ELM327	24
Figura 3.4. Módulo GSM Sim800L	27
Figura 3.5 Topología de la red GSM del Sistema de monitoreo de vehículos.....	28
Figura 3.6 Arduino-UN-915 (Arduino LoRaWAN)	29
Figura 3.7 Módulo Gateway RG1xx	30
Figura 3.8. Diagrama de Flujo -Servidor de Red	31
Figura 3.9 Topología de la red GSM del Sistema de monitoreo de vehículos.....	32
Figura 3.10 Modelo entidad-relación de la base de datos “Flota”	33
Figura 3.11 Página Home de la plataforma del cliente	34
Figura 3.12 Página de registro de vehículos en la Plataforma del Cliente	35
Figura 3.13 Página de Contacto de la Plataforma del Cliente.....	36
Figura 3.14 Formulario de contacto en la Plataforma del Cliente.....	36
Figura 3.15 Conexión del módulo GSM Sim800L y <i>Raspberry pi 3</i>	37
Figura 3.16. Representación de conexión Raspberry Pi 3 y Arduino LoRaWAN	37
Figura 4.1 Ubicación del Gateway Edificio 15A	39
Figura 4.2. Extensión de la ruta recorrida para la prueba de cobertura de LoRaWAN....	39
Figura 4.3. Distancia máxima de cobertura en LoRaWAN.....	40
Figura 4.4 Tiempo entre paquetes recibidos LoRaWAN.....	40
Figura 4.5 Tiempo entre paquetes recibidos GSM	41
Figura 4.6 Página de monitoreo de la Plataforma del Cliente	42
Figura 4.7 Página de monitoreo de la Plataforma del Cliente	42
Figura 4.8 Monitoreo de vehículos parámetro RPM Servidor Ubuntu	43
Figura 4.9 Monitoreo de vehículos parámetro RPM Servidor Raspbian	43
Figura 4.10. Representación gráfica de proyección de costos de LoRaWAN	45
Figura 4.11. Representación gráfica de proyección de costos de GSM.....	46

ÍNDICE DE TABLAS

Tabla 1.1 Análisis comparativo entre las diferentes soluciones propuestas.....	4
Tabla 2.1 Funcionalidades asociadas a los pines del conector DLC	9
Tabla 2.2 Características de los protocolos de comunicación para el sistema OBD-II....	11
Tabla 3.1 Parámetros y comandos de medición.....	25
Tabla 4.1 Costo total de implementación usando LoRaWAN	44
Tabla 4.2 Costo total de implementación usando GSM	44
Tabla 4.4 Proyección de costos LoRaWAN (6 meses)	45
Tabla 4.4. Proyección de costos GSM (6 meses)	45

ÍNDICE GENERAL

RESUMEN	I
ABSTRACT	II
ABREVIATURAS	III
SIMBOLOGÍA	IV
ÍNDICE DE FIGURAS	V
ÍNDICE DE TABLAS	VI
1. INTRODUCCIÓN	1
1.1 Descripción del problema	1
1.2 Delimitación del problema	2
1.3 Posibles soluciones	3
1.4 Solución Propuesta	3
1.5 Justificación	4
1.6 Objetivos	5
1.6.1 Objetivo General	5
1.6.2 Objetivos Específicos	5
1.7 Metodología	6
CAPÍTULO 2	8
2. Marco Teórico	8
2.1 Estándar Obd	8
2.1.1 OBD-I	8
2.1.2 OBD-II	8
2.2 Sistema OBD-II	8
2.2.1 Conector DLC	8
2.2.2 Protocolos de Comunicación	11
2.2.3 Modos de Operación	12
2.2.4 Check Engine Light	14
2.3 Plataforma de Conectividad	14
2.4 Raspberry Pi	15
2.5 Tecnologías de Comunicación	15
2.5.1 LoRaWAN	16
2.5.2 GSM	17
2.5.3 Bluetooth	18
2.6 Formatos de Intercambio de Datos	18

2.6.1	JSON.....	18
2.7	Aplicativo Web	19
2.7.1	CSS	19
2.7.2	Bootstrap	19
2.7.3	Javascript	19
2.7.4	HyperText Markup Language 5.....	19
2.7.5	PHP	20
2.7.6	Servidor Web	20
2.7.7	Base de datos	20
2.7.8	MySQL.....	20
2.7.9	Node Red	20
CAPÍTULO 3	21
3.	DESCRIPCIÓN DE LA SOLUCIÓN	21
3.1	Descripción General.....	21
3.2	Método de la solución	22
3.2.1	Adquisición de datos.....	22
3.2.2	Conectividad.....	24
3.2.2.1	Codificación de Datos.....	24
3.2.2.2	Transmisión y Decodificación de Datos	25
3.2.3	Recepción y Presentación de datos	32
3.2.3.1	Almacenamiento de datos	32
3.2.3.2	Plataforma del Cliente	34
3.3	ESQUEMÁTICO DE LA SOLUCIÓN	37
CAPÍTULO 4	38
4.	PRUEBAS Y RESULTADOS	38
4.1	Análisis LoRaWAN	38
4.2	Análisis GSM.....	41
4.3	Análisis de datos	41
4.4	Análisis Financiero	44
4.5	Comparación técnica.....	46
CONCLUSIONES Y RECOMENDACIONES	48
Conclusiones	48
Recomendaciones	49
BIBLIOGRAFÍA	50
ANEXOS	53

ANEXO A:	53
LISTA DE COMANDOS SOPORTADOS POR LIBRERÍA PY-OBd EN EL MODO 1	53
ANEXO B:	57
FLOTA DE VEHÍCULOS DE CONDUESPOL	57
ANEXO C:	58
PLAN DE PRUEBAS.....	58
ANEXO D:	60
VALIDACIÓN DEL CLIENTE CONDUESPOL.....	60

CAPÍTULO 1

1. INTRODUCCIÓN

A partir del año 1980, las compañías automotrices debían comercializar los vehículos con un sistema de diagnóstico a bordo integrado, OBD, (On-Board Diagnostics) que cumplieran con las normas de emisión de gases contaminantes determinadas por la Agencia de Protección del Medio Ambiente, EPA [1].

Con el tiempo, la Sociedad de Ingenieros Automotrices (SAE) requería que todos los vehículos fabricados a partir de 1996 posean un conector estandarizado por lo que este sistema evolucionó a una segunda generación (OBD II), que aparte de controlar las emisiones de contaminantes, permita detectar mediante códigos de fallas el problema que presenta el sistema. [2]

El uso de este conector proporcionaría la identificación del sistema o componente relacionado con el daño y a la vez permitiría llevar un monitoreo y control del vehículo asociado a este puerto lo cual ayudaría al parque automotor de las empresas a realizar una planificación de mantenimientos sean tipo correctivo o preventivo.

A continuación, se presentará la finalidad, beneficios y soluciones propuestas que se pueden llevar a cabo para mejorar el sistema de mantenimientos de vehículos.

1.1 Descripción del problema

La escuela de conductores profesionales de ESPOL E.P, también conocida como CONDUESPOL, posee una flota de 39 vehículos conformada por:

- 10 Automóviles CHEVROLET SEDAN SAIL 1.5 4X2 TM
- 22 Automóviles CHEVROLET AVEO ACTIVO 1.6 L
- 2 Camioneta CHEVROLET D- MAX 4X2 3.0 L DIESEL
- 4 Camiones CHEVROLET NQR 75L 5.2 4X2 TIPO FURGON

- 1 Camioneta GREAT WALL DOBLE CABINA 4X2 TM 2.8 1 TON

Sobre la cual se realizan mantenimientos preventivos y correctivos. Los mantenimientos preventivos se centran en cambios de aceite, filtros y refrigerantes y se realizan 4 veces al año. Dos por cada curso de obtención de licencia, el cual tiene una duración de 6 meses. Al ser vehículos que realizan recorridos cortos, aproximadamente menos de 10.000 km al año el cambio de aceite no se realiza en base al kilometraje como comúnmente ocurre, sino que se lleva a cabo cada 3 meses. Por otro lado, los mantenimientos correctivos se realizan únicamente cuando el carro experimenta algún tipo de falla y esta es reportada.

Cuando un vehículo es llevado al taller para ser reparado, al finalizar el procedimiento el mecánico se encarga de registrar el mantenimiento realizado en una plataforma web, por lo que la empresa tiene registros de lo que ha ocurrido en cada uno de los vehículos de su flota. La forma en que se ejecuta este procedimiento no permite obtener información del estado de la flota de manera constante e incluso la información obtenida no es precisa. A pesar de que se lleven a cabo los mantenimientos, no se puede asegurar que los vehículos no experimenten algún tipo de falla.

Si una unidad presenta una falla mecánica durante el desarrollo de una práctica de conducción, se altera el recorrido establecido y el tiempo definido para la misma, generando malestar en el estudiante. En caso de que el daño no pueda ser solucionado de manera inmediata otro vehículo reemplaza las funciones a cumplir por el vehículo averiado o se sustituye el componente defectuoso. Si la falla experimentada es grave, se pone en riesgo incluso la vida de las personas que viajan dentro del vehículo.

1.2 Delimitación del problema

La plataforma de conectividad propuesta para este proyecto será desarrollada durante el segundo término académico 2018 y evaluada en un vehículo perteneciente a la escuela de conducción de ESPOL E.P de marca CHEVROLET modelo AVEO ACTIVO (Ver Anexo B). El recorrido para las pruebas a realizar será dentro de ESPOL (Campus Prosperina), ya que la institución lleva a cabo las prácticas de conducción en el campus. Se usarán las tecnologías de comunicación

LoRaWAN y GSM, debido a que la universidad cuenta con una estación base LoRaWAN, instalada en el laboratorio Pedro Carló perteneciente a la carrera de Ingeniería en Telemática. Este lugar fue seleccionado porque en la actualidad la ciudad de Guayaquil no cuenta con suficientes bases LoRaWAN instaladas. Las pruebas de transmisión y recepción de datos se llevarán a cabo dentro de un radio de aproximadamente 448 m correspondiente al área de cobertura de la estación.

1.3 Posibles soluciones

Para el desarrollo de este proyecto se analizaron las siguientes propuestas:

1. Sistema de mantenimiento preventivo usando interfaz OBD II que toma datos automáticamente del vehículo, esto se lo realizará mediante el diseño de una plataforma de conectividad encargada de comunicarse con el puerto mencionado para extraer la información del estado del vehículo y enviarla de manera autónoma a un aplicativo desarrollado para el cliente.
2. Actualizar los parámetros medidos de cada revisión para generar una retroalimentación del estado del camión a través del ajuste de la frecuencia de muestreo sobre la plataforma de conectividad a desarrollar.
3. Monitorear los camiones en los puntos de revisión, ubicados en lugares estratégicos. De esta manera, el personal técnico registrará el estado del vehículo en la base de datos de la empresa a través de un aplicativo web.

1.4 Solución Propuesta

Se considera el uso del estándar OBD II como la solución más factible, debido a que en la actualidad la mayoría de los vehículos poseen este sistema integrado. El estándar OBD II permite conocer el estado de los sistemas funcionales del vehículo mediante el uso de un conector de diagnóstico OBD II. Con la finalidad de que el cliente pueda tener un control de lo que ocurre en los vehículos de su flota, se desarrollará un aplicativo web en el que se muestre ciertos parámetros medidos desde el vehículo y de la misma manera se diseñará una plataforma de conectividad que se encargue de extraer esta información desde el puerto OBD-II y enviarla a la plataforma del cliente.

1.5 Justificación

De acuerdo a las soluciones planteadas se procedió a realizar un análisis comparativo para determinar los beneficios y dificultades que estas conllevaban.

Tabla 1.1 Análisis comparativo entre las diferentes soluciones propuestas

	Ventajas	Desventajas
Solución 1	<ul style="list-style-type: none">• Monitoreo del estado del vehículo en tiempo real.• Detección de fallas de algún componente de manera precisa.• Extracción de parámetros de estado del vehículo de manera automatizada.	<ul style="list-style-type: none">• Saturación de la plataforma del cliente por el envío excesivo de datos.• Fallas de fábrica del dispositivo OBD II.• Sistema disponible sólo para vehículos del año 2000 en adelante.• Errores en la transmisión de datos entre la plataforma de conectividad y el dispositivo debido a interferencias en el medio de comunicación.
Solución 2	<ul style="list-style-type: none">• Mediciones de los parámetros del vehículo en diferentes periodos de tiempo.• Capacidad de reconfiguración del periodo de medición de datos en la plataforma de conectividad.	<ul style="list-style-type: none">• Tiempo de espera de actualización de la plataforma de conectividad.
Solución 3	<ul style="list-style-type: none">• Información organizada del estado de los vehículos.• Actualizaciones del estado del vehículo en diferentes puntos de su recorrido.	<ul style="list-style-type: none">• Capacitación al personal técnico sobre el uso del aplicativo Web.• Mayor gasto en instalación, mantenimiento de equipos y localización de la oficina.• Registro erróneo de datos.

De acuerdo con el análisis comparativo que se observa en la Tabla 1, se determinó que la solución 1 es la que más se adecúa a los requerimientos del cliente. La

implementación de una plataforma de conectividad utilizando el Sistema de Diagnóstico a Bordo, OBD II (On Board Diagnostic System Information) permitirá conocer en tiempo real el comportamiento de los sistemas funcionales y fallas de un vehículo. Llevar a cabo un monitoreo continuo del estado del vehículo permitirá realizar la planificación de mantenimientos preventivos, evitar daños mecánicos prematuros, sustituir partes defectuosas a tiempo y prevenir los accidentes de tránsito.

Al tener acceso a información actualizada del parque automotor, la empresa conocería con más certeza las unidades que se encuentran aptas para llevar a cabo los recorridos de las prácticas de conducción.

1.6 Objetivos

1.6.1 Objetivo General

- Implementar una plataforma de conectividad y monitoreo de diagnóstico de vehículos en tiempo real para la planificación de mantenimientos mediante datos obtenidos del estándar OBD II.

1.6.2 Objetivos Específicos

- Diseñar una plataforma de conectividad para el registro y monitoreo automatizado del estado del vehículo.
- Identificar los parámetros de estado del vehículo del estándar OBD II para su conversión en el formato requerido para la plataforma del cliente.
- Desarrollar una plataforma web que permita al cliente la visualización de los parámetros de estado de los vehículos de su flota para la supervisión y control de cada uno de estos.
- Evaluar el desempeño de varias tecnologías de comunicación realizando un análisis de la integridad de los datos y costos.

1.7 Metodología

En base a los requerimientos del cliente se elaborará el diseño de una plataforma de conectividad usando herramientas de diagramación de esquemáticos para determinar la conexión de los componentes que intervendrán en la solución propuesta. Entre ellos: conector OBD II, Raspberry Pi y los módulos de comunicación de las tecnologías LoRAWAN y GSM.

Para la identificación de la funcionalidad del estándar OBD-II se llevará a cabo pruebas en una unidad de la flota de vehículos del cliente, a través de un dispositivo que se conecta directamente al puerto y que se comunica vía bluetooth a una aplicación instalada en un dispositivo móvil.

Los parámetros leídos por el dispositivo OBD II serán enviados mediante bluetooth hacia la plataforma de conectividad, en este una Raspberry Pi tendrá definidas librerías para la recepción e interpretación de las tramas de datos. Toda esta información será almacenada en una base de datos para lo cual se establecerán modelos relacionales.

Una vez acordado con el cliente los parámetros de los cuales se desea tener información actualizada, se procederá a procesarlos y convertirlos en formato JSON (requerido por la plataforma del cliente).

Se realizará una comparación entre las tecnologías de comunicación LoRaWAN y GSM realizando pruebas de integridad en el sistema propuesto. La Figura 1.1 presenta un esquema de la solución propuesta para el proyecto.

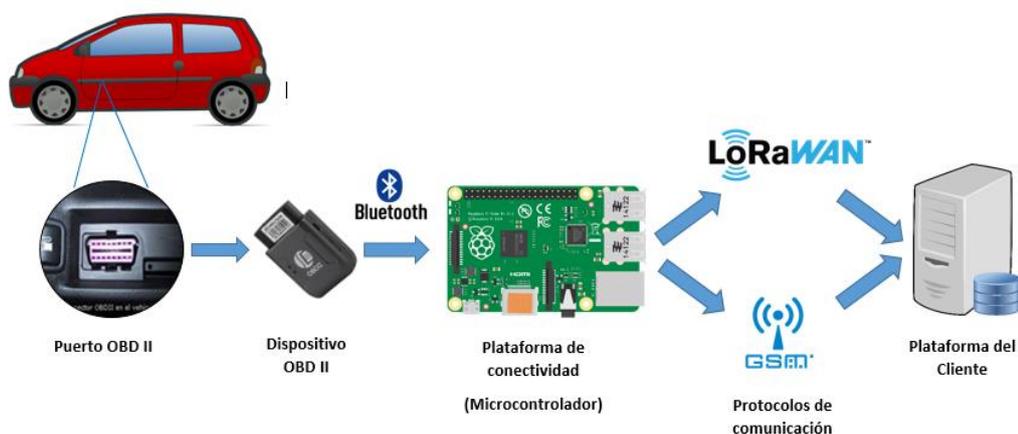


Figura 1.1 Conexión entre el puerto OBD II, la plataforma de conectividad y la plataforma del cliente

CAPÍTULO 2

2. MARCO TEÓRICO

En esta sección se describirá cada una de las tecnologías usadas para el desarrollo de este proyecto.

2.1 Estándar Obd

De acuerdo con la página web de la Agencia de Protección del Medio Ambiente, Diagnóstico a bordo, “On Board Diagnostics” (OBD), es un sistema informático integrado en todos los vehículos y camiones livianos del 1996 y posteriores, como requerimiento por parte de Enmiendas de la Ley de Aire Limpio (EPA) de 1990. Los sistemas OBD son están diseñados para monitorear el desempeño de algunos de los componentes principales del motor incluyendo aquellos que son responsables del control de emisiones [3].

El sistema también guarda otra información que facilita a los mecánicos o a los propietarios de los vehículos a encontrar y reparar problemas. A continuación, se especifican las diferencias entre la primera y segunda versión del protocolo OBD.

2.1.1 OBD-I

OBD-I es considerado la primera versión del sistema integrado OBD que fue implementado en 1991. Su enfoque principal era monitorear únicamente los componentes encargados de controlar las emisiones producidas por el vehículo.

Sin embargo, presentaban una falta de consistencia en la comunicación y la interfaz lo cual originaba que vehículos de diferentes marcas tuvieran interpretaciones diferentes

2.1.2 OBD-II

OBD-II (On Board Diagnostic Second Generation) es la segunda y más reciente versión del sistema OBD, surge como una nueva reforma del EPA en 1996 para automóviles y camiones livianos poniendo a disponibilidad un conector de 16 pines

estándar a nivel mundial. Su objetivo principal es la detección de fallos químicos, mecánicos y eléctricos presentes en el vehículo y la disminución del índice de emisiones de manera automática. Permite la autocorrección de cualquiera anomalía en la mezcla de combustible o en el encendido del vehículo.

2.2 Sistema OBD-II

Todos los sistemas OBD-II poseen un conector de diagnóstico (DLC), códigos de fallas comunes (DTC) y tienen el lenguaje común propio de la unidad de control electrónico (ECM) es decir un programa por el cual se puede acceder a toda la información del estado de los sensores de la ECU.

2.2.1 Conector DLC

El sistema OBD-II implementa un conector de enlace de diagnóstico estándar o DLC (Diagnostic Link Connector) de 16 pines (2x8) y forma trapezoidal definido por la norma SAE-J1962. Sirve como interfaz de acceso y recuperación de datos desde la Unidad de Control de Motor (ECU) hacia un equipo de diagnóstico. De acuerdo a la normativa el conector puede ser de tipo A, para vehículos de 12 [V] o tipo B para vehículos de 24 [V], ambas hembras, diferenciándose en la forma de la pestaña de alineación. La Figura 2 muestra los conectores mencionados. [4] [5]

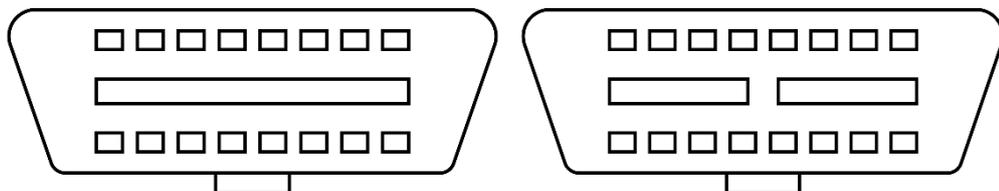


Figura 2.1 Conector DLC J1962 Tipo A (izquierda) y Tipo B (derecha)

- **Descripción de los pines**

La funcionalidad de cada uno de los pines del conector DLC está descrita por los diferentes fabricantes de vehículos, sin embargo, existe cierta distribución estándar para las conexiones a tierra (pines 4 y 5) y para suministro de energía a través de la batería del auto (pin 16). El resto de pines corresponden a los protocolos de comunicación existentes para la interfaz del OBD-II, esto se explica mejor en la siguiente sección. La Figura 2.2 presenta el esquema general del conector OBD-II con la distribución de los terminales del conector DLC. [6]

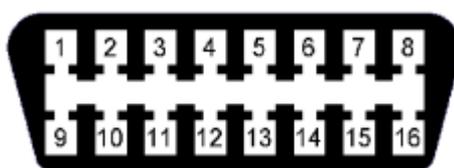


Figura 2.2. Terminales del conector DLC ODB-II

La tabla 2.1 muestra una descripción de la funcionalidad asociada cada pin en el conector DLC.

Tabla 2.1 Funcionalidades asociadas a los pines del conector DLC [7]

PIN	DESCRIPCIÓN
1	Reservado para el fabricante
2	SAE J1850 Bus +
3	Reservado para el fabricante
4	Tierra (vehículo)
5	Tierra (señal)
6	CAN Alto
7	Línea K, ISO 9141-2 y ISO/DIS 14230-4
8	Reservado para el fabricante
9	Reservado para el fabricante
10	SAE J1850 Bus -
11	Reservado para el fabricante
12	Reservado para el fabricante
13	Reservado para el fabricante
14	CAN Bajo
15	Línea L, ISO 9141-2 y ISO/DIS 14230-4
16	Alimentación de batería

- **Ubicación del conector**

De acuerdo con la norma SAE J1962 [8] , el DLC tipo A "debe estar ubicado en el compartimiento del pasajero o del conductor en el área delimitada por el extremo del conductor del panel de instrumentos a 300 mm (1 pie) más allá de la línea central del vehículo, unido al panel de instrumentos y fácil para acceder desde el asiento del conductor. La ubicación preferida es entre la columna de dirección y la línea central del vehículo". El DLC tipo B "se ubicará en el compartimiento del

pasajero o del conductor en el área delimitada por el extremo del conductor del panel de instrumentos, incluyendo el lado exterior, y una línea imaginada 750 mm (2.5 pies) más allá de la línea central del vehículo. Será conectado al panel de instrumentos y de fácil acceso desde el asiento del conductor o desde el asiento del copiloto o desde el exterior. El conector del vehículo debe montarse para facilitar el acoplamiento y el desacoplamiento". Las Figuras 2.3 y 2.4 muestran las ubicaciones de los dos tipos de conectores.

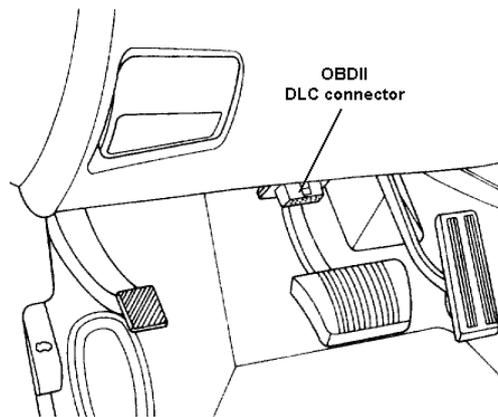


Figura 2.3. Ubicación DLC Tipo A [9]

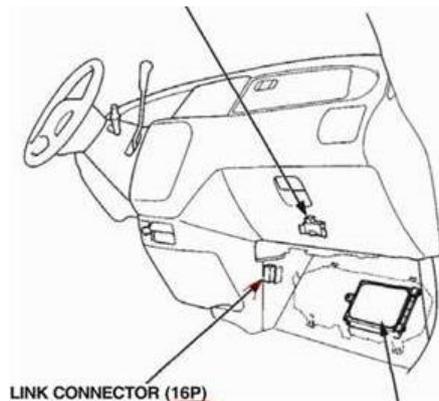


Figura 2.4. Ubicación DLC Tipo A [10]

- **Lector de Códigos**

Son dispositivos de bajo costo que se usan principalmente para escanear códigos de error (DTC) y leer parámetros de estado desde la computadora a bordo del vehículo cuando se conectan al conector DLC. Debido a que no cuentan con una interfaz gráfica, usualmente poseen interfaces bluetooth y wifi, para poder conectarse a dispositivos móviles (celulares o tabletas) a través de aplicaciones

desarrolladas para el estándar OBD-II. Es usado mayormente en los talleres mecánicos o tecnicentros para identificar que sensor o actuador ha dejado de funcionar correctamente, ya que la lámpara indicadora de mal funcionamiento (MIL) se enciende al detectar un código de falla. Luego de esto se verificará el daño revisando los sensores, actuadores, líneas o conectores para proceder a hacer la reparación y finalmente limpiar el código de error de la memoria del vehículo de manera que la luz MIL se apague. Se pueden encontrar en diferentes tiendas de repuestos de vehículos o incluso en tiendas de hardware o implementos para el hogar. [11] [12]

2.2.2 Protocolos de Comunicación

La forma en que opera el sistema OBD-II en un vehículo depende del protocolo de comunicación especificado por el fabricante. La diferencia entre protocolos es mínima y se basa principalmente en la descripción de los pines usados en el conector DLC. Actualmente existen 5 protocolos de comunicación a nivel de capa física y de enlace, la tabla 2.2 indica sus principales características.

Tabla 2.2 Características de los protocolos de comunicación para el sistema OBD-II [13] [14]

PROTOCOLO	FABRICANTE	TÉCNICA DE COMUNICACIÓN	TASA DE TRANFERENCIA	VOLTAJE	DESCRIPCIÓN PINES
SAE J1850	Ford Motors Mercury Lincoln	PWM (Pulse Width Modulation)	41,6 Kbaud/s	0-5 [V]	5 (Tierra) 16 (Alimentación) 2 y 10 (Datos)
SAE J1850	General Motors	VPW (Variable Pulse Width)	10,4 - 41,6 Kbaud/s	0-7 [V]	5 (Tierra) 16 (Alimentación) 2 (Datos)
ISO9141-2	Chrysler Jeep Dodge Fabricantes de Europa y Asia	Parecido al RS232	10,4 Kbaud/s	0-12 [V]	5 (Tierra) 16 (Alimentación) 7 (Datos)
ISO14230-4 KWP2000 (Keyword Protocol 2000)	Fabricantes de Europa y Asia	Parecido al RS232	1,2 - 10,4 Kbaud/s	0-12 [V]	5 (Tierra) 16 (Alimentación) 7 y 15 (Datos)

ISO 15765	Bosch	CAN (Contorller Area Network)	250 - 500 Kbaud/s	CANH: 3.5V (min/max 2.75 a 4.50) CANL: 1.5V (min/max 0.5 a 2.25)	5 (Tierra) 16 (Alimentación) 6 (Datos High) 14 (Datos Low)
-----------	-------	-------------------------------------	-------------------	---	--

2.2.3 Modos de Operación

El módulo de control electrónico (ECM) provee varios servicios de diagnóstico denominados modos de operación, los cuales están definidos por la normativa SAE J1979. Los fabricantes de vehículos pueden no contar con todos los modos y tienen la posibilidad de incluir nuevos en caso de requerirlos. Cada modo tiene un identificador único que sigue la forma "\$XX", donde las XX representan un valor numérico en formato hexadecimal. A continuación, se describe cada uno de los modos, indicando su nombre y funcionalidad e identificador [15].

- **MODO 1: Acceso a datos de Diagnóstico Actuales**

Identificador (\$01)

Permite el acceso a los valores de datos actuales relacionados con las emisiones, incluyendo las entradas y salidas analógicas, las entradas y salidas digitales y la información de estado del sistema.

- **MODO 2: Acceso a Cuadro de Datos Congelados**

Identificador (\$02)

Permite el acceso a un conjunto de parámetros del motor y del vehículo almacenados en la memoria no volátil, que son generados cada vez que se presenta un código de error.

- **MODO 3: Acceso a Códigos de Error de Diagnóstico**

Identificador (\$03)

Permite la extracción de los códigos de error almacenados en la ECU.

- **MODO 4: Borrar /Restablecer Información de Diagnóstico relacionada las Emisiones**

Identificador (\$04)

Proporciona un medio para que el equipo de prueba externo ordene a la ECU borrar toda la información de diagnóstico relacionada con las emisiones.

- **MODO 5: Acceso a los Resultados de la Prueba de monitorio del Sensor de Oxígeno**

Identificador (\$05)

Proporciona los resultados del autodiagnóstico realizado en los sensores de oxígeno para determinar el funcionamiento de los mismos y la eficiencia del convertidor catalítico. Se aplica principalmente solo a vehículos de gasolina.

- **MODO 6: Acceso a Resultados de las Pruebas de monitoreo a bordo para Sistemas Específicos**

Identificador (\$06)

Permite el acceso a los resultados para las pruebas de monitoreo de diagnóstico a bordo de componentes/sistemas específicos que son monitoreados continuamente.

- **MODO 7: Acceso a Códigos de Problemas Pendientes**

Identificador (\$07)

Permite al equipo de prueba externo obtener códigos de diagnóstico de diagnóstico (DTC) "pendientes" detectados durante el ciclo de conducción actual o el último completado para componentes/sistemas relacionados con las emisiones.

- **MODO 8: Solicitud de Control del Sistema, Prueba o Componente a Bordo**

Identificador (\$08)

Permite al equipo de prueba externo controlar la operación de un sistema, prueba o componente a bordo. Con la ayuda de este servicio, el técnico de servicio puede activar un modo de prueba a bordo.

- **MODO 9: Acceso a la Información del Vehículo**

Identificador (\$09)

Permite al equipo de prueba externo solicitar información específica del vehículo, como: Número de identificación del vehículo (VIN), Número de Calibración del Módulo (CALID), Número de verificación de calibración (CVN) y Valores de la relación de rendimiento en uso (IUPR).

- **MODO 10: Solicitar Códigos de Problemas Permanentes**

Identificador (\$0A)

Permite al equipo de prueba externo obtener todos los códigos de diagnóstico de problemas con el estado "permanente". Estos son los DTC que están "confirmados" y se almacenan en la memoria no volátil de la ECU.

2.2.4 Check Engine Light

Conocida también como lámpara indicadora de mal funcionamiento, MIL, (Malfunction Indicator Light) permite que el sistema de control de propulsión (PCM) alerte al conductor que la computadora del vehículo ha detectado una falla mediante el encendido de una luz denominada "check engine". Esta luz se encuentra en el panel del tablero, puede ser de color amarillo o rojo. Debido, a que la forma o el etiquetado de esta luz no se encuentra estandarizado, los fabricantes de vehículos usan el ícono de un motor, las palabras "check engine" o "Service Engine Soon", o incluyen la palabra "check" dentro del ícono del motor. La Figura 2.5 muestra las diferentes formas que MIL puede presentarse en el tablero de los vehículos.

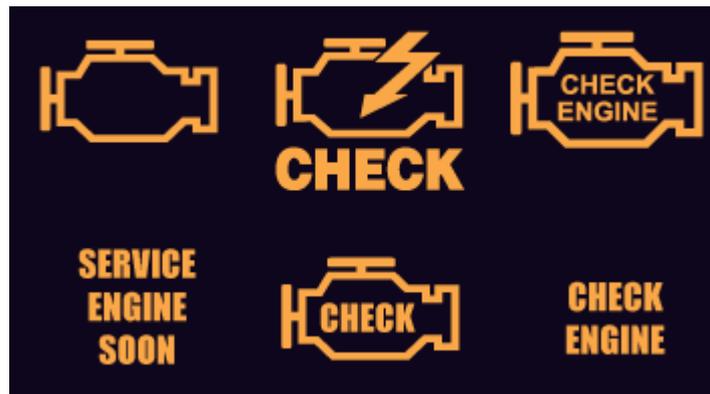


Figura 2.5. Diferentes formas en las que se presenta la MIL [16]

2.3 Plataforma de Conectividad

Una plataforma de conectividad también llamada M2M (Machine to machine) está formada por dispositivos conectados a través de redes de telecomunicaciones (por ejemplo, tarjetas SIM) que intercambian información de manera autónoma. Suelen confundirse con las plataformas IoT pero a diferencia de estas, basan sus funciones

en procesamiento y enriquecimiento de los diferentes conjuntos de datos de los sensores.

2.4 Raspberry Pi

Es una computadora de bajo costo y de tamaño reducido, la cual posee un microprocesador, unidad de procesamiento gráfico, memoria RAM, tarjeta de memoria SD y puertos periféricos. Funciona con las diferentes distribuciones de Linux.

Modelos

- Raspberry Pi 1 Modelo A
- Raspberry Pi 1 Modelo B y B
- Raspberry Pi 2 Modelo B
- Raspberry Pi 3 Modelo B
- Raspberry Pi 3 Modelo B+
- Raspberry Pi 3 Modelo A+

Raspberry Pi 3 Modelo B

Es la tercera generación de Raspberry Pi. Entre las principales mejoras están:

- Un procesador de cuatro núcleos
- Conectividad Wifi 802.11n y Bluetooth 4.
- Conectividad Bluetooth Low Energy (BLE)

2.5 Tecnologías de Comunicación

En la actualidad existen diferentes tecnologías de comunicación que permiten la transmisión y recepción de datos. Dado que para la implementación de un sistema de monitoreo en tiempo real se requiere de un protocolo de comunicación que permita la comunicación a largas distancias, un menor consumo de energía y el mayor envío de datos en un tiempo determinado. En el presente estudio, se procederá a describir los protocolos LoRaWAN y GSM. Debido a que LoRaWAN es nuevo en el mercado ecuatoriano y GSM lleva años en operación.

2.5.1 LoRaWAN

El protocolo de red de baja frecuencia y área amplia está diseñado para definir una red inalámbrica con dispositivos y aplicaciones de bajo consumo a una tasa de bits baja.

Esta arquitectura implementa una topología estrella-estrella en donde se establece un intercambio de mensajes entre dispositivos finales, gateways y un servidor de red central. [17]

Este protocolo utiliza tecnología LoRa (Low Range), que hace referencia al tipo de modulación en radiofrecuencia. Se encuentra en la capa física del modelo OSI, emplea la técnica del espectro ensanchado, donde se hace uso de un ancho de banda adicional al necesario, para que se puedan recibir señales de diferentes velocidades. La transmisión y recepción de datos se establece a distancias superiores a 20 Km.

LoRaWAN permite ajustar y administrar parámetros de conexión como lo son canal, ancho de banda, cifrado de información, entre otros. Trabaja en la banda de ultra alta frecuencia (UHF) operando en frecuencias de 433 MHz, 868 MHz y 915 MHz pertenecientes a Asia, Europa, América respectivamente. [18]

Existen diferentes servidores de red LoRaWAN que a la vez ofrecen su aplicación para la visualización y manipulación de datos entre ellos se encuentran Lorient, Senet, Strem y The Things Network (TTN).

Entre las características principales del protocolo LoRaWAN se encuentran:

- Largo alcance y bajo consumo
- Topología Estrella
- Protocolo bidireccional
- Un alcance aproximadamente de 10 a 15 Km en línea de vista
- Posee encriptación AES 128
- Baja transferencia de datos

Plataforma The Things Network

También denominada TTN, es una plataforma que tiene toda la programación y estructura de una red determinada, da soporte a los dispositivos registrados. La plataforma TTN puede determinar la duplicidad de mensajes, gestión de integración con otras plataformas, registro de dispositivos entre otras. Ofrece diferentes interfaces de lenguaje de programación entre ellas: Java, Node-Red, Node.js, mediante esto se puede construir una aplicación de nodos, gateways y servidores TTN también una plataforma desarrollada

2.5.2 GSM

Sistema Global para las Comunicaciones Móviles (GSM) es un estándar mundial para teléfonos móviles digitales, el cual fue creado por la CEPT. Una red GSM está compuesta por:

- Sistema de Conmutación
- Sistema de Estaciones Base
- Dispositivos móviles del usuario

El sistema de conmutación se conecta a la red telefónica pública conmutada (PSTN), identifica al suscriptor al cuál se desea enviar información, ubicándolo en su zona de cobertura.

En sistema de estaciones base, se encuentran los transmisores los cuales son capaces de verificar y corregir los errores que se presenten en las señales recibidas, también permiten encriptar la información. Cada estación base es el centro de una celda y emite señales de radio a los móviles que se estén cerca. [19]

SMS

Short Message Service es un servicio que permite el intercambio de mensajes de texto mediante teléfonos móviles. El tamaño máximo de un mensaje de texto es de 160 caracteres de 7 bits. La encapsulación incluye algunos parámetros como por ejemplo fecha de envío, hora, número de teléfono destinatario y remitente y el número de SMSC que originó el mensaje. El Centro de Servicio de mensajes cortos (SMSC) procesa, almacena los mensajes hasta que se envíen, también permite la conexión con los elementos que pertenecen a la red GSM.

GSM en Ecuador

De acuerdo con las operadoras de telecomunicaciones que se encuentran actualmente en el Ecuador, se encontró las frecuencias a las cuales trabajan cada una de ellas:

- CONECEL S.A, opera en el rango de frecuencias comprendido entre los siguientes rangos: 850MHz–1900MHz GSM.
- OTECEL S.A., opera en el rango frecuencias comprendido entre los siguientes rangos 850MHz–1900MHz GSM.
- CNT E.P, opera en la frecuencia 1900 MHz GSM.

2.5.3 Bluetooth

Es un estándar de comunicación inalámbrico, utiliza transmisiones de radio de onda corta y consume poca energía. Permite la conexión de dispositivos que se encuentren dentro del radio de alcance. El alcance depende del medio ya que la comunicación se puede ver afectada por la estructura física del lugar. En esta tecnología se puede realizar la transmisión de voz, datos, vídeo.

2.6 Formatos de Intercambio de Datos

Los estándares de intercambio de información surgieron para que las computadoras y dispositivos puedan comunicarse independiente del sistema operativo o el lenguaje de programación utilizado. Debido a que se requiere enviar los parámetros de estados hacia el servidor, para luego decodificarlos se procedió a seleccionar el siguiente formato de intercambio de datos.

2.6.1 JSON

JavaScript Object Notation, JSON, es un formato libre, ligero, fácil de entender que permite el intercambio de datos. Los objetos JSON se utilizan cuando se requiere transmitir datos a través de la red. Está construido sobre dos formas:

- Una colección de pares de nombre de la propiedad, valor de la propiedad
- Una lista, vector o secuencia ordenada de valores

2.7 Aplicativo Web

Una aplicación web es un programa que permite la interacción entre los usuarios y los datos, estos datos son procesados el desarrollo de una tarea o varias tareas específicas en la web.

Para el desarrollo de la plataforma web se utilizan las herramientas que se describen a continuación:

2.7.1 CSS

Es un lenguaje que permite definir la presentación de archivos realizados con HTML. Permite determinar el aspecto de la página web como lo es color, tamaño y tipo de letra a utilizar, estilo de los botones, tablas, cuadros de texto, contenedores, separaciones, ubicación de los elementos entre otros.

2.7.2 Bootstrap

Es un framework CSS que permite dar forma y estilo a un determinado sitio web, posee librerías CSS que facilitan la creación de interfaces y adaptación a pantallas de diferentes dispositivos. Posee un sistema GRID, en el cual los elementos se encuentran posicionados en columnas determinadas.

Bootstrap es compatible con los navegadores como: Opera, Google Chrome, Safari, Mozilla FireFox e Internet Explore.

2.7.3 Javascript

Es un lenguaje de programación basado en acciones, utilizado para crear efectos, donde los usuarios puedan interactuar con el sistema Web. Los programas creados se denominan scripts.

2.7.4 HyperText Markup Language 5

HyperText Markup Language, HTML5, es la última versión de HTML, es un lenguaje de etiquetas que permite el desarrollo de páginas webs. En esta versión se puede realizar una mejor interacción entre el contenido multimedia y la página web.

2.7.5 PHP

Hypertext Preprocessor, PHP, es un lenguaje de código abierto que permite el desarrollo de páginas web dinámicas. Es utilizado para la interpretación de los datos en el servidor web, puede implementarse dentro del código HTML5.

2.7.6 Servidor Web

Un servidor es un ordenador que provee servicios a otras computadoras denominadas clientes, los cuales almacenarán a cualquier tipo información. También se encarga de enviar la información solicitada. En el servidor se almacenan documentos de diferentes tipos.

2.7.7 Base de datos

Es el conjunto de datos que poseen una relación entre sí. Una base de datos está compuesta por: Hardware, Software y datos. El hardware está formado por dispositivos de almacenamiento, el software está ligado al gestor de administración de base de datos y los datos que la información que será procesada, almacenada y solicitada.

2.7.8 MySQL

Es un sistema de gestión de base de datos relacional de licencia pública general. Entre las características que posee están:

- Tiene un sistema de contraseñas flexible y seguro.
- Contiene y soporta grandes bases de datos, permite hasta 64 índices por tabla.
- Puede contener de 1 a 16 columnas cada tabla. El máximo ancho de límite de cada columna es de 1000 bytes.
- Reserva de memoria muy rápido.
- Funciona en diferentes plataformas.

2.7.9 Node Red

Es una herramienta que permite conectar hardware, servicios en línea e interfaces de diferentes aplicaciones mediante la programación visual. Los elementos o módulos para conectar se denominan nodos.

CAPÍTULO 3

3. DESCRIPCIÓN DE LA SOLUCIÓN

En este capítulo se describirá la solución propuesta para la adquisición de estados de parámetros del vehículo, la implementación de la plataforma de conectividad y la plataforma del cliente.

3.1 Descripción General

Se propone diseñar el sistema mostrado en la Figura 3.1, estructurado por los siguientes módulos:

- **Adquisición de datos:** Conexión directa de un escáner de diagnóstico ELM327 con interfaz bluetooth al puerto estándar OBD-II del vehículo.
- **Codificación de datos:** Integración de los datos obtenidos del escáner bluetooth en una plataforma Raspberry Pi 3 mediante el uso de librería Python OBD. Esta será la Plataforma de conectividad.
- **Transmisión de datos:** Uso de módulos de comunicación: GSM (Sim 800L) y LoRaWAN (nodo Arduino) ambos mediante interfaz UART de los pines GPIO de la Raspberry.
- **Decodificación de datos:** Transferencia de datos se realizará a través de la red GSM (Claro) y la red LoRaWAN (radio base de ESPOL).
- **Recepción y Presentación de datos:** Configuración de un servicio web para la recepción y presentación de los datos en una plataforma para el cliente.

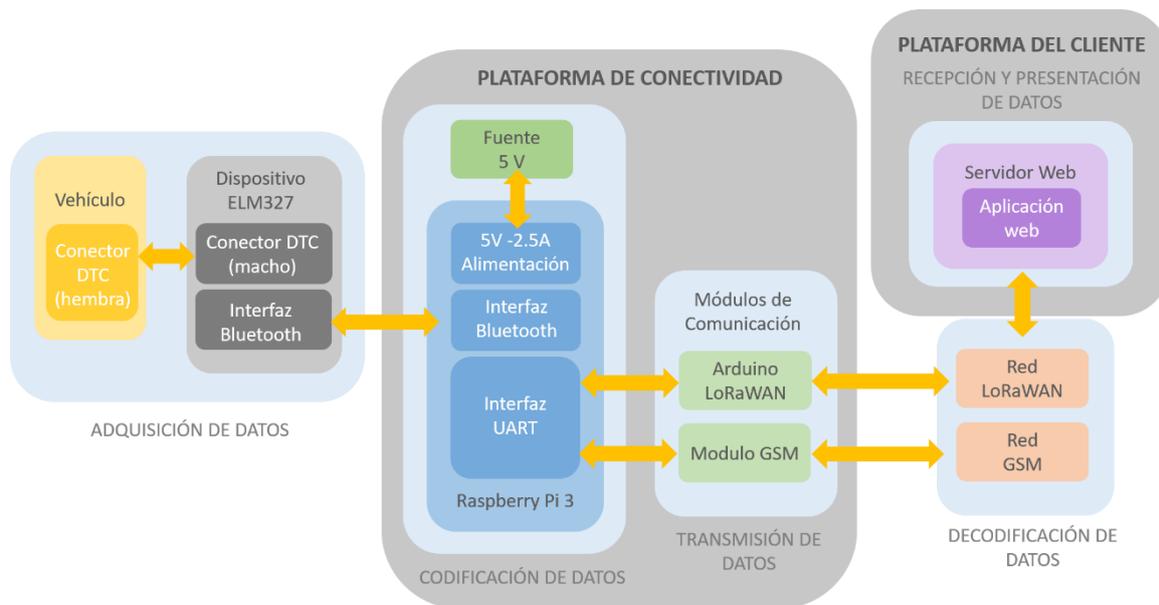


Figura 3.1. Diagrama de bloques del sistema propuesto

3.2 Método de la solución

Se detallará la funcionalidad de cada uno de los módulos a implementar en el Sistema de monitoreo de diagnóstico de vehículos descritos en la sección anterior.

3.2.1 Adquisición de datos

En esta sección se describe el proceso de adquisición de parámetros de estado del vehículo desde la ECU.

- **Localización del puerto OBD-II en el vehículo**

De acuerdo con la norma SAE J1962 [8], se determinó que el conector DLC es de tipo B y de acuerdo a la marca CHEVROLET modelo AVEO ACTIVO, el puerto se encuentra localizado en la parte inferior del volante como se muestra a continuación en la Figura 3.2.



Figura 3.2. Localización del conector DLC OBD-II

- **Interfaz de Diagnóstico OBD-II ELM327 Bluetooth**

La empresa Enterprise Learning Management (ELM), desarrolla microcontroladores y otros productos electrónicos. Uno de sus productos que más sobresale son los microcontroladores ELM para diagnóstico automotriz como el OBD II. Este microcontrolador procesa los datos transmitidos por la unidad de control electrónica, ECU (Engine Control Unit) de acuerdo con el protocolo que se utilice.

Este dispositivo tiene como elemento principal al microcontrolador ELM327 perteneciente a la empresa mencionada. Este circuito integrado, tiene como medio de comunicación los protocolos: bluetooth, WiFi o RS232, a una tasa de 38400 baudios, permite comunicar e interpretar los datos de diferentes protocolos. Se escogió el dispositivo, como se observa en la Figura 3.3, debido a que cuenta con tecnología bluetooth, es de bajo costo y asequible. El cual permitirá la adquisición de los parámetros de estado enviados por la computadora del vehículo.



Figura 3.3. Interfaz de Diagnóstico ELM327 [20]

3.2.2 Conectividad

Para la implementación de la plataforma de conectividad se hará uso de una Raspberry pi 3 modelo B, la cual se encargará de la recepción y procesamiento de los parámetros del estado del vehículo enviados por dispositivo de análisis OBD II.

A continuación, se describirá los diferentes protocolos y módulos de comunicación a utilizar.

3.2.2.1 Codificación de Datos

Para establecer una comunicación entre el dispositivo lector y plataforma de conectividad (Raspberry Pi 3) se hará uso del protocolo bluetooth, disponible en ambas partes y de la librería Python-OBD. Esta librería posee funciones que permiten la conexión con el dispositivo de diagnóstico ELM327 y la extracción de datos del vehículo en tiempo real desde el puerto OBD-II mediante la especificación de los comandos. Las principales funciones a utilizar se describen a continuación [21]:

- OBD()
Crea un nuevo objeto de conexión OBD y escanea los puertos seriales para Bluetooth y USB, luego escoge la primera conexión disponible de la Plataforma de Conectividad.

- `scan_serial()`
Escanea y retorna una lista de los puertos USB o RF válidos, es decir, aquellos que están conectados.
- `query()`
Envía un comando OBD al vehículo y retorna un objeto de tipo OBD respuesta. Esta función se bloquea hasta recibir una respuesta del vehículo, también verifica que el comando sea soportado por este. En caso de que no sea soportado, no se envía y se retornará una respuesta OBD vacía. La lista de todos los comandos disponibles se adjunta en el anexo B.

En la tabla 4 se indica los parámetros escogidos para obtener mediciones y sus respectivos comandos del modo 1 del estándar OBD-II en la librería PyOBD.

Tabla 3.1 Parámetros y comandos de medición

PARÁMETROS	COMANDO
Velocidad	SPEED
RPM	RPM
Posición del acelerador	THROTTLE_POS
Nivel de Combustible	FUEL_STATUS
Temperatura del refrigerante	COOLANT_TEMP
Carga del motor	ENGINE_LOAD
Temperatura del aire	AMBIANT_AIR_TEMP

Estos parámetros fueron seleccionados para precautelar el estado de vida del motor. La plataforma de conectividad esperará que la trama de los parámetros de estado del vehículo leídos de objeto OBD esté completa para el envío por los puertos mini UART o UART.

3.2.2.2 Transmisión y Decodificación de Datos

Para establecer la comunicación entre la plataforma de conectividad y plataforma del cliente se realizará un análisis entre las tecnologías GSM y LoRaWAN en donde se presentarán las configuraciones respectivas.

Una vez extraídos y codificados los parámetros de estado del vehículo, descritos en la sección anterior, se procederá a explicar el proceso transmisión y decodificación de datos.

Comunicación GSM

En esta sección se describe el esquema de la solución propuesta utilizando la tecnología de GSM. Para los nodos de transmisión y recepción se utilizará una Raspberry pi 3 para el procesamiento de los datos.

Conexión Raspberry pi a una red GSM

En la Raspberry se deberán realizar los siguientes pasos para activar los puertos seriales mini UART:

- Actualizar el repositorio, los paquetes instalados y el firmware.
- Habilitar la interfaz UART en el encabezado de GPIO del archivo `/boot/config.txt`
- Deshabilitar la consola en el archivo `/boot/cmdline.txt`
- Se deberá agregar en el archivo `/lib/systemd/system/hciuart.service` el puerto a utilizar. En este caso `ttyS0`.

Para establecer la red se utilizará el módulo GSM de la serie SIM800L, el cual permitirá la transmisión y recepción de parámetros del estado del vehículo. Este módulo fue seleccionado de acuerdo a las características que posee, además que es una versión más nueva en módulos y posee un regulador de voltaje.

Módulos GSM

Los siguientes módulos se utilizarán para la transmisión y recepción de datos.

- GSM Sim800L V2.0
- GSM Sim800L V2.2
- Módulo de Identidad de Suscriptor

GSM Sim800L

Este módulo presenta las siguientes características:

- Voltaje de operación: 3.4~ 4.4[V] DC

- Frecuencias de trabajo: Quad-Band 850/900/1800/1900 MHZ
- Consumo de Corriente: 500mA (max), 0.7mA (sleep)
- Temperatura de operación: -40 a +85 [°C]
- Interfaz: Serial UART
- Datos GPRS:
 - Velocidad de Transmisión máxima: 85.6 Kbps
 - Chip con protocolo TCP/IP
 - Soporte a reloj en tiempo real (RTC)
 - Velocidad de transmisión serial: 1200 bps – 115200 bps



Figura 3.4. Módulo GSM Sim800L [22]

Módulo de Identidad de Suscriptor

También llamado Tarjeta SIM, permitirá identificar el suscriptor del número celular y ofrece el servicio móvil. Los módulos de serie SIM800L requieren de la tarjeta SIM nano que tiene las dimensiones de 1.2cm de largo x 0.8cm de ancho.

Topología GSM

En la Figura 3.5, se muestra el esquema general de comunicación para la tecnología GSM. Los nodos de transmisión se encargarán de transferir los parámetros de estado del vehículo extraídos mediante el dispositivo de diagnóstico ELM327. Estos nodos enviarán un mensaje de texto que contendrá los valores de estado leídos hacia un número determinado utilizando un módulo SIM800L. La Red GSM se encargará de conmutar el mensaje de texto hacia el número especificado, este número pertenecerá a la plataforma del cliente, donde estará conectado otro módulo SIM800L que se encargará de recibir el mensaje para su decodificación. Una vez que el nodo SIM800L receptor haya recibido un

mensaje de texto, se revisará que el destinatario pertenezca a un vehículo registrado. Si el destinatario existe, el módulo receptor leerá el mensaje, identificando los parámetros recibidos. Luego estos parámetros serán almacenados en una base de datos denominada “Flota” para luego ser mostrados en la aplicación web.

La aplicación web se encuentra en el nodo receptor en un Servidor Raspbian.

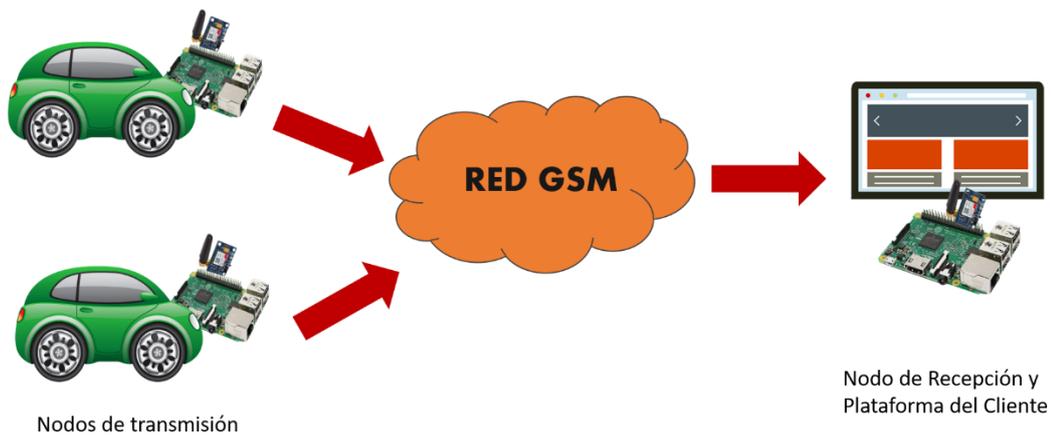


Figura 3.5 Topología de la red GSM del Sistema de monitoreo de vehículos

Comunicación LoRaWAN

En esta sección se describe el esquema de la solución propuesta utilizando la tecnología de LoRaWAN. Para los nodos de transmisión se utilizará una Raspberry pi 3 que permitirá la codificación y procesamiento de los datos junto con un Arduino TTN-UN-915 encargado de enviar los datos al gateway.

Módulos

Se describen los módulos a utilizar en el nodo transmisor y el nodo receptor:

Nodo Transmisor: TTN-UN-915

Este módulo posee las siguientes características:

- Alcance inalámbrico aproximadamente hasta 10 Km

- Biblioteca LoRaWAN incluida
- Funciona a una frecuencia de 915 MHz (América)
- Basado en estándares de código abierto
- Integración en la nube con plataformas IoT

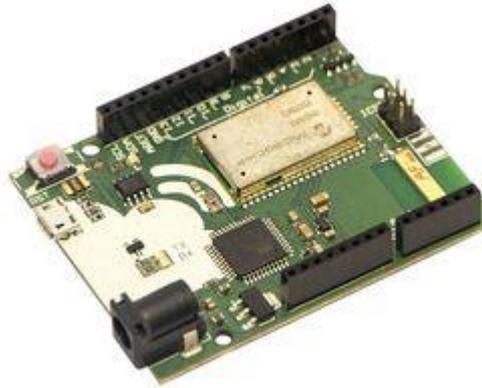


Figura 3.6 Arduino-UN-915 (Arduino LoRaWAN) [23]

Nodo Gateway: Sentries RG1xx

El dispositivo posee las siguientes características:

- Sistema operativo Linux, kernel 4.x
- Interfaces: LoRaWAN, 802.11 a/b/g/n, Bluetooth v4.0 y Ethernet
- Frecuencias:
LoRa 863- 870 MHz (Europa), 902-928 MHz (América)

Wi-Fi 2.4 y 5 G

- Temperatura: -30 a +70 [°C]
- Tiene un radio de cobertura de aproximadamente 16 Km
- Soporta los estándares de seguridad: WEP, WPA, WPA2
- Encriptación: WEP, TKIP



Figura 3.7 Módulo Gateway RG1xx [24]

Topología LoRaWAN

Los dispositivos se encuentran divididos de acuerdo con la Figura 3.9, se describirán las partes de la topología:

Nodos Finales: Se conectarán los dispositivos de hardware: una Raspberry pi 3 modelo B y un arduino LoRaWAN. La Raspberry Pi se encargará de remitir los datos procesados a través del puerto serial del arduino LoRaWAN utilizando un cable conector USB A macho - USB micro B macho (Ver Figura 3.16). El arduino será capaz de enviar la información de los parámetros de estado de los vehículos hacia el Gateway.

Gateway: Estará ubicado en un edificio con línea de vista. Este dispositivo puede comunicarse de forma bidireccional vía LoRaWan con los nodos finales. Se encargará de recibir los datos de los nodos finales para luego enviar la información hacia Internet en donde la plataforma The Things Network se encargará de receptor los datos.

Servidores de Red LoRaWAN: Se seleccionó la Plataforma The Things Network (TTN) ya que es Open Source, de fácil manejo y ofrece diferentes métodos de integración con aplicaciones. Para esto se procedió a registrar en la plataforma TTN el nombre la aplicación, la MAC e identificadores del nodo Arduino LoRaWAN y el gateway a utilizar. Una vez finalizado el registro, se generarán automáticamente las claves de acceso para la aplicación y dispositivos registrados. La plataforma se

encargará de procesar y decodificar los datos transmitidos por el Gateway, para luego enviarlos al servidor donde se alojará la aplicación web.

Servidores de Red: Se encargará del enrutamiento y reenvío de los mensajes de la plataforma The Things Network hacia la Aplicación Web. En esta parte se utilizó Node-Red ya que permite integración de los módulos The Things Network, mediante una programación visual, para esto se realizó el proceso que se muestra en el diagrama de flujo de la Figura 3.8:

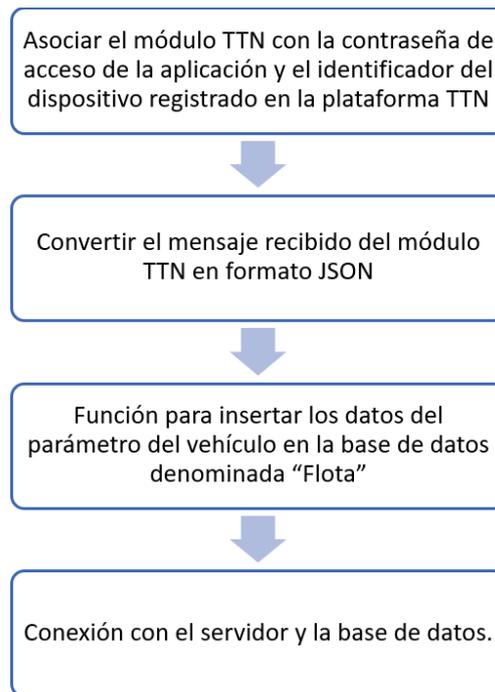


Figura 3.8. Diagrama de Flujo -Servidor de Red

Aplicación Web: Sistema que permite mostrar el monitoreo y registro de estado actual de vehículos. La aplicación se ejecuta dentro de un servidor Ubuntu 16.04 LTS.

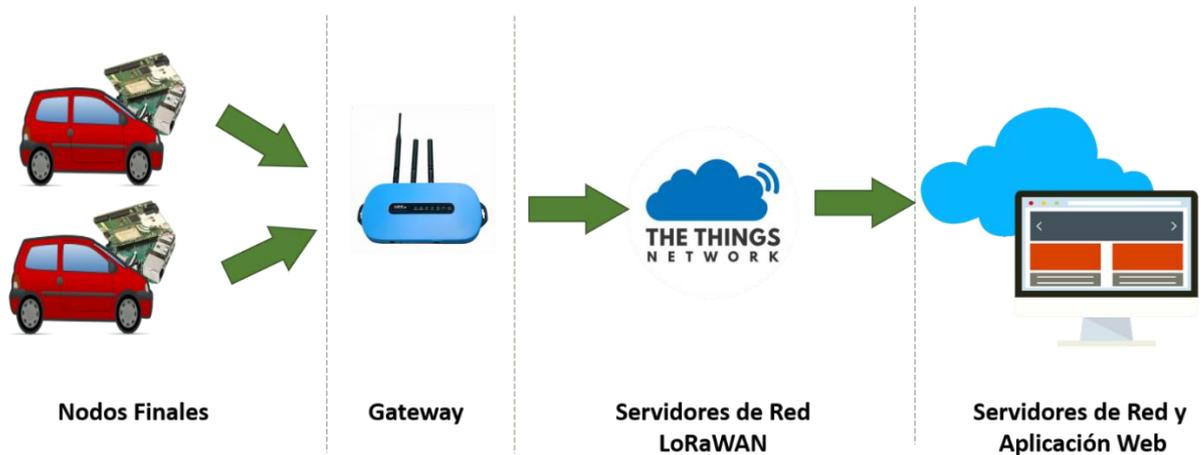


Figura 3.9 Topología de la red GSM del Sistema de monitoreo de vehículos

3.2.3 Recepción y Presentación de datos

Una vez que los datos fueron decodificados por el servidor de red, estos serán almacenados en una base de datos para ser visualizados en la aplicación web de monitoreo. A continuación, se describen los elementos a utilizar.

3.2.3.1 Almacenamiento de datos

Se almacenarán los parámetros de estado del vehículo en una base de datos en MySQL denominada "Flota", el modelo entidad-relación se muestra en la Figura 3.10:

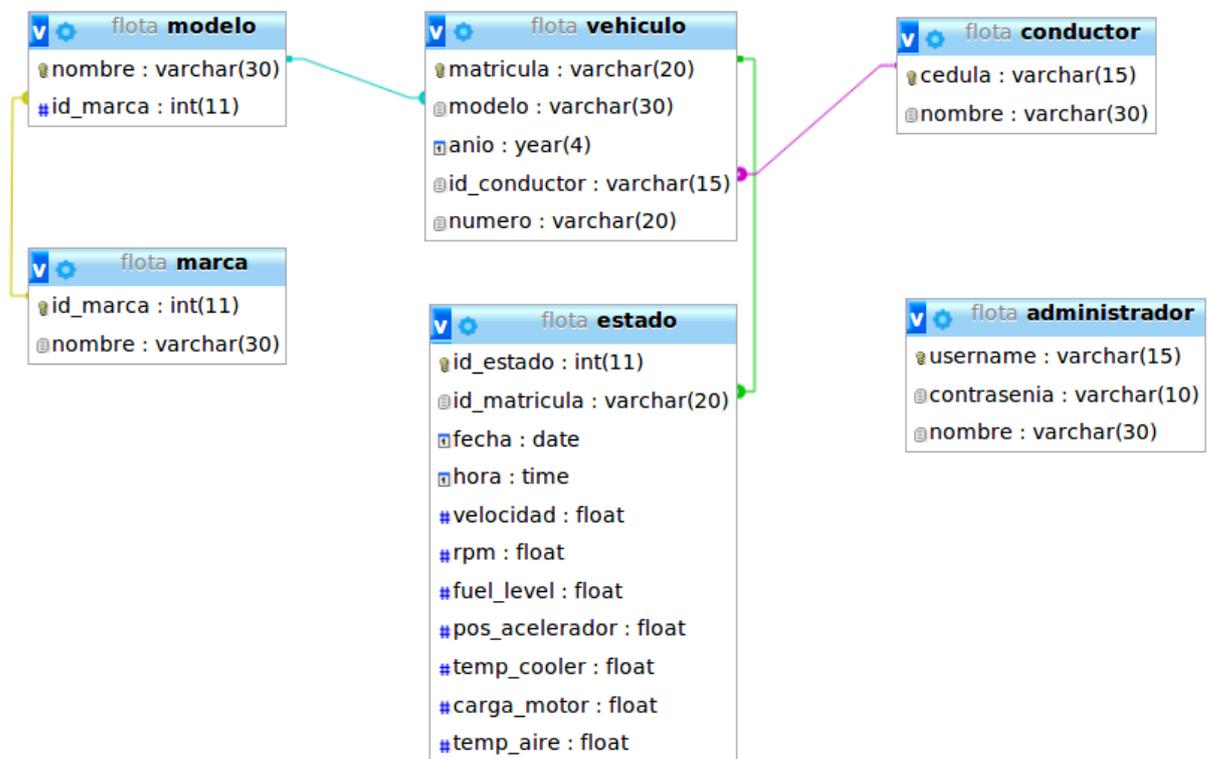


Figura 3.10 Modelo entidad-relación de la base de datos “Flota”

Tabla marca: Contiene las marcas de vehículos existentes en el parque automotor, asociadas a un identificador respectivo.

Tabla modelo: Información de los modelos de vehículos asociados a la marca.

Tabla vehículo: Contiene información de un vehículo de parque automotor como: matrícula, modelo al que pertenece, año de fabricación, número de celular, identificación del conductor.

Tabla conductor: Contiene información del conductor encargado del vehículo.

Tabla estado: Contiene información de la matrícula del vehículo, fecha, hora y parámetros de estado como: velocidad, RPM, nivel de combustible, posición del acelerador, temperatura del aire, carga del motor y temperatura del refrigerante.

Tabla administrador: Contiene información de los técnicos que realizarán el monitoreo de parámetros de estado del vehículo y tendrán acceso al aplicativo web.

3.2.3.2 Plataforma del Cliente

Para el diseño de la aplicación Web se utilizarán herramientas como: Bootstrap, HTML5, CSS, PHP y JavaScript las cuales permitirán la asignación de estilos, posicionamiento de elementos e interacción con la página web.

Se presenta la página principal de la plataforma del cliente (ver Figura 3.11):

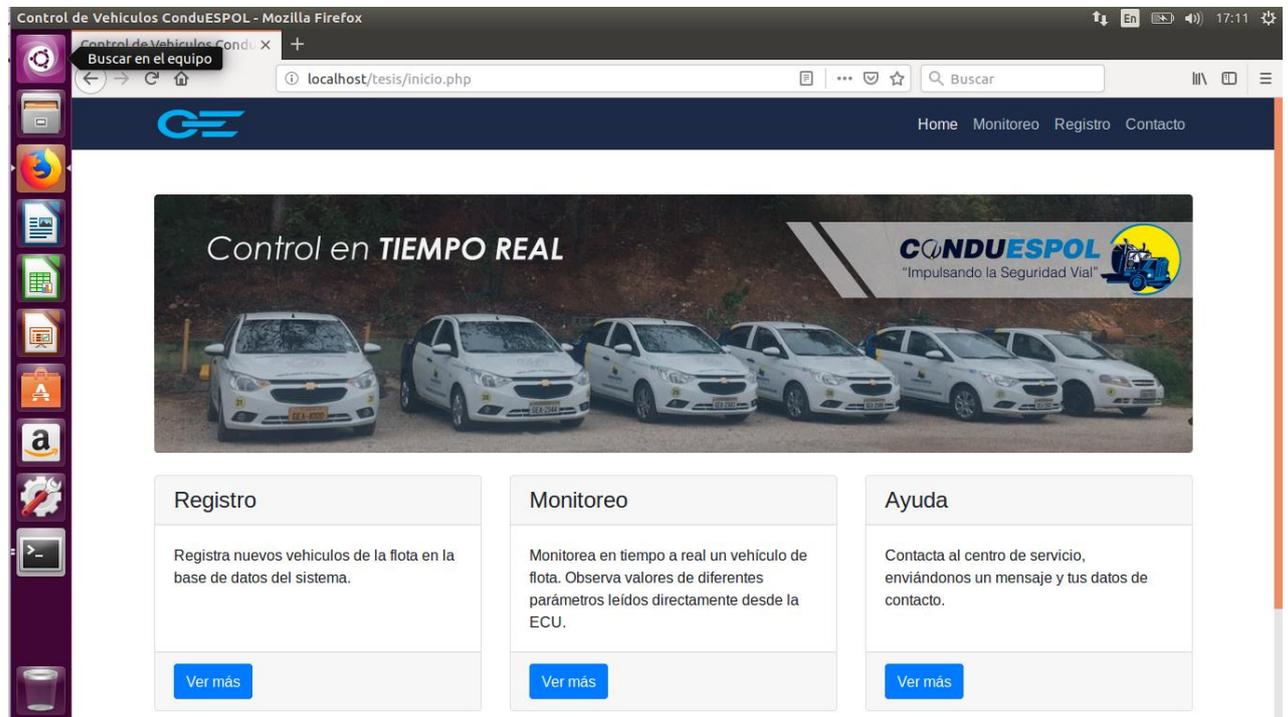


Figura 3.11 Página Home de la plataforma del cliente

En la Figura 3.12 se observa la página de registro de un nuevo vehículo, para lo cual se necesitará especificar la marca, el modelo, matrícula, año de fabricación, conductor encargado de la unidad, número de cédula y el número celular que se asociará al vehículo.

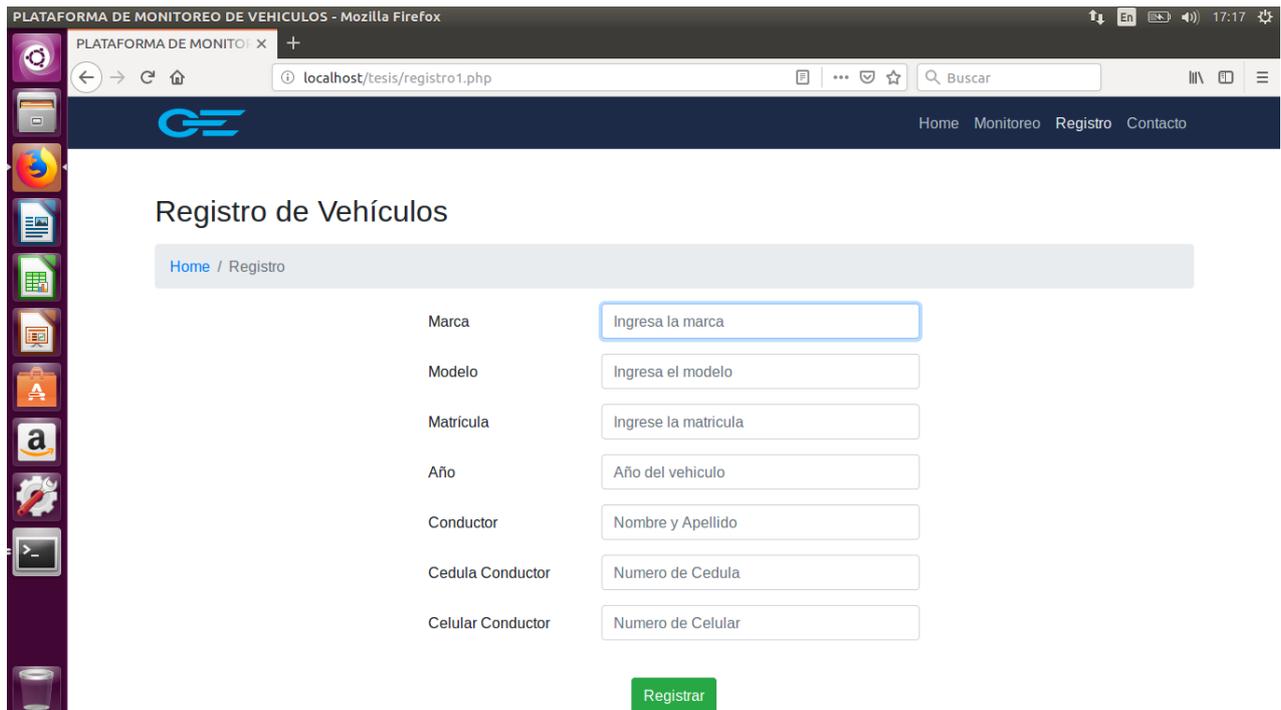


Figura 3.12 Página de registro de vehículos en la Plataforma del Cliente

La plataforma del cliente también tiene asociado una página de contacto, donde se muestra la ubicación de ConduEspol y los correos de las personas encargadas de la aplicación (Ver Figura 3.13). La página de contacto también tiene un formulario en caso de que se desee realizar sugerencias acerca de la aplicación de monitoreo de vehículos (Ver Figura 3.14) .

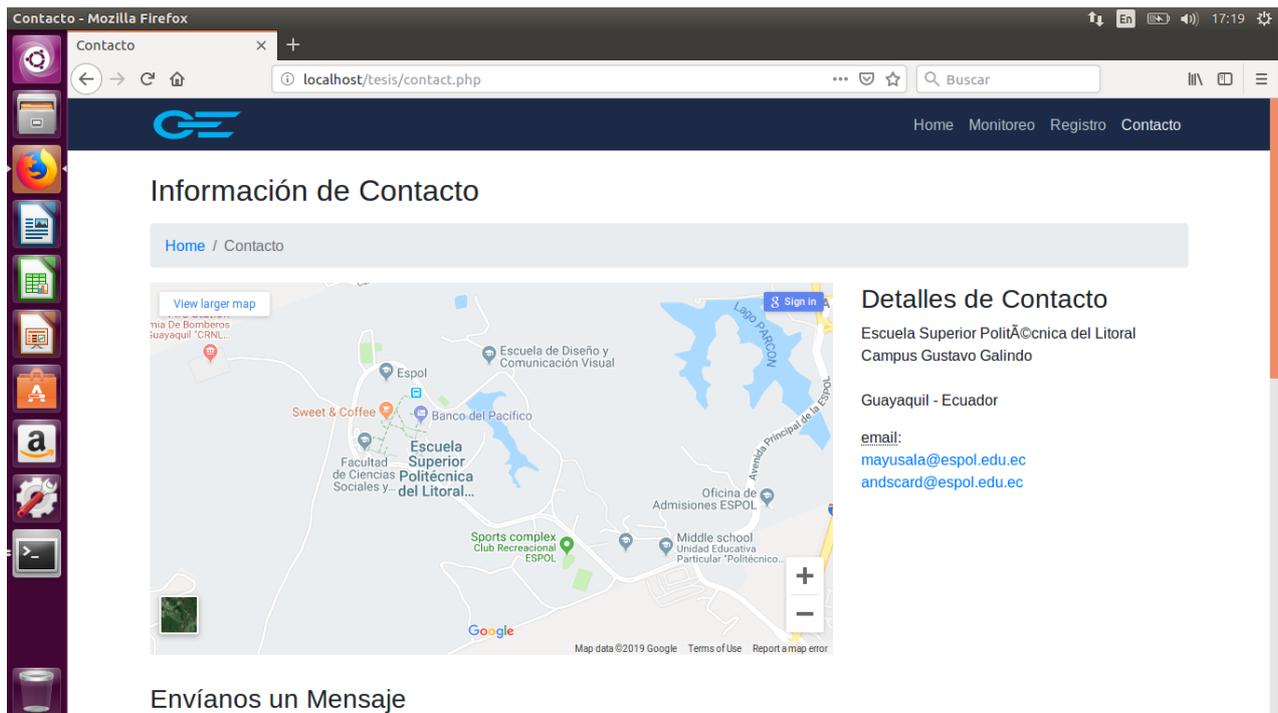


Figura 3.13 Página de Contacto de la Plataforma del Cliente

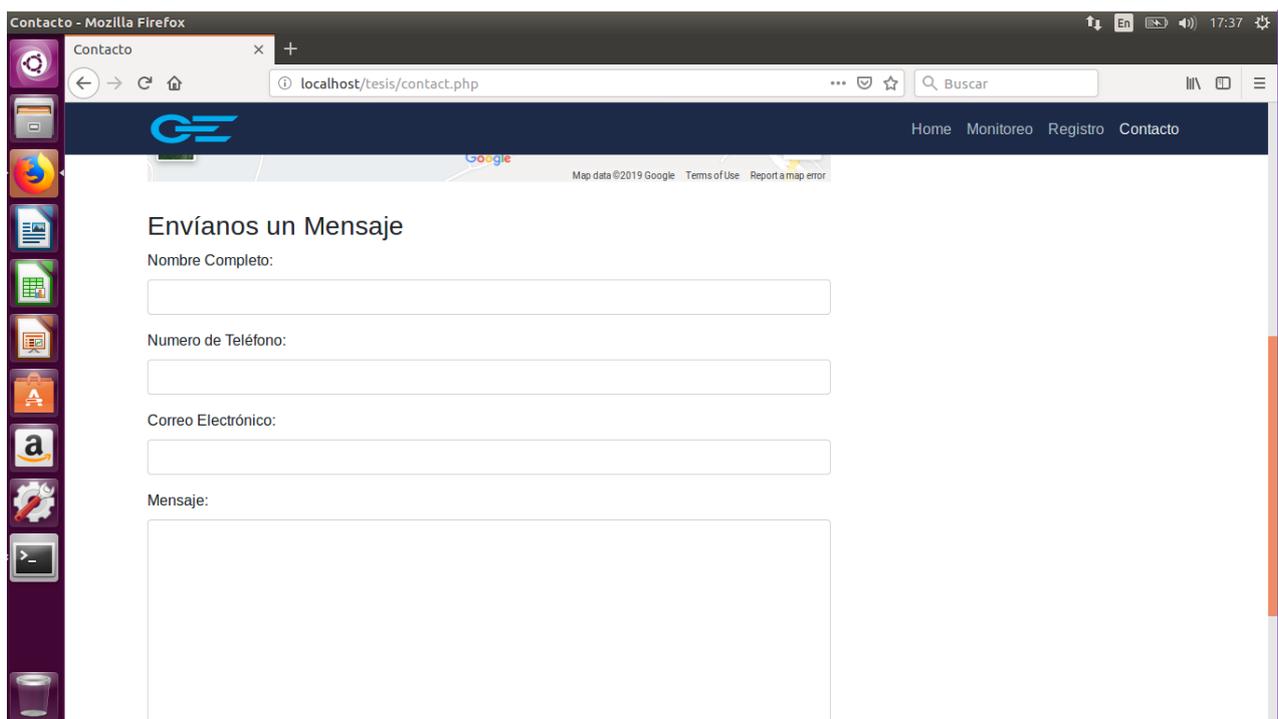


Figura 3.14 Formulario de contacto en la Plataforma del Cliente

3.3 ESQUEMÁTICO DE LA SOLUCIÓN

A continuación, en las Figura 3.15 y 3.16 se presentan diagramas esquemáticos de la solución propuesta para este proyecto, en el cual se muestran las conexiones entre los diferentes dispositivos utilizados.

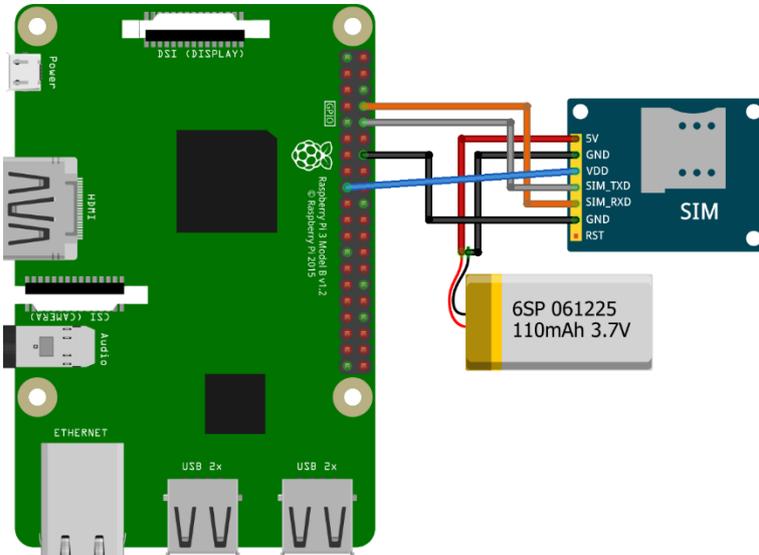


Figura 3.15 Conexión del módulo GSM Sim800L y *Raspberry pi 3*

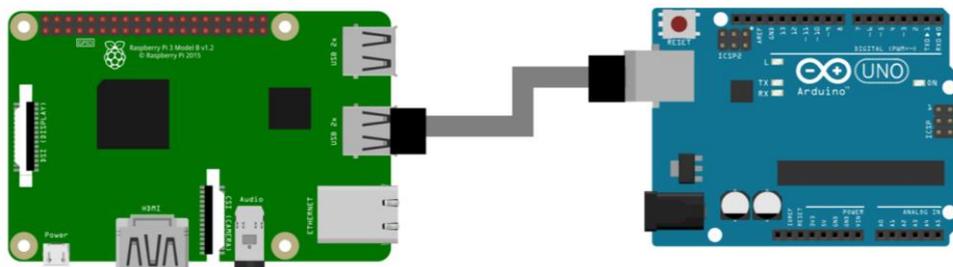


Figura 3.16. Representación de conexión Raspberry Pi 3 y Arduino LoRaWAN

CAPÍTULO 4

4. PRUEBAS Y RESULTADOS

A continuación, se detallan los resultados obtenidos en la implementación de la aplicación de monitoreo de estados de parámetros de vehículos en tiempo real utilizando las tecnologías LoRaWAN y GSM. Además, se incluye un análisis financiero de costos de implementación.

4.1 Análisis LoRaWAN

Se realizó un recorrido de 1.92 km en la avenida principal de la Espol y sus alrededores (ver Figura 4.2), buscando la distancia máxima de transmisión. Se tomó como punto de partida el parqueadero de profesores de la Facultad de Ingeniería en Electricidad y Computación. Ubicando el dispositivo Gateway en la terraza del edificio 15A a una altura de 5.50 m (Ver Figura 4.1) y el módulo arduino transmisor LoRaWAN en el vehículo el cuál se mantuvo en movimiento en todo el recorrido, enviando datos cada tres segundo. Se llevó un monitoreo en la plataforma The Things Network de los paquetes recibidos mediante un contador establecido, logrando transmitir hasta una distancia de aproximadamente 448 m (ver Figura 4.3).

Se determinó que existen zonas en donde la conexión es inestable dado que en la plataforma The Things Network se observaron que ciertos paquetes se perdieron, esto se debe a que no hay línea de vista entre el Gateway y el módulo arduino transmisor LoRaWAN, ya que la infraestructura física de los edificios que se encuentran entre ambos dispositivos impide que la señal se propague correctamente. Mientras que en otras zonas se perdía la conexión totalmente.

En las Figuras 4.2 y 4.3 las zonas con conexión estable están representadas con viñetas verdes, las inestables con viñetas amarillas y conexión nula con viñetas rojas. Adicionalmente se indica con una viñeta naranja la posición del Gateway.



Figura 4.1 Ubicación del Gateway Edificio 15A

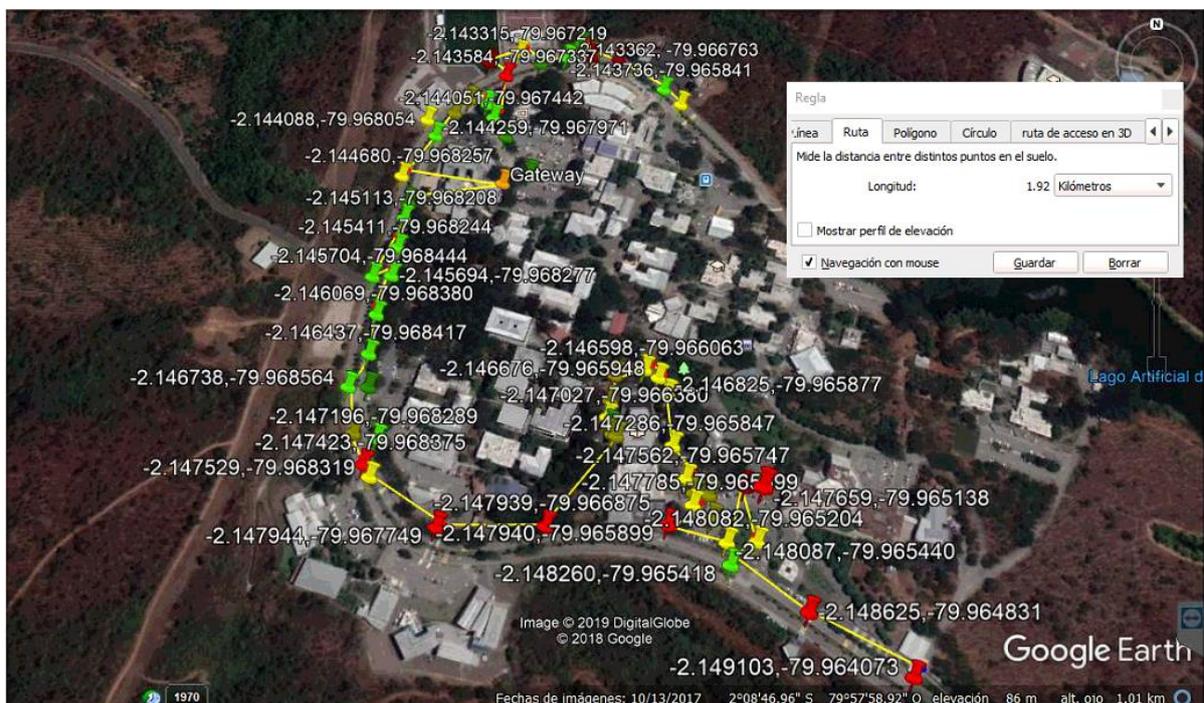


Figura 4.2. Extensión de la ruta recorrida para la prueba de cobertura de LoRaWAN

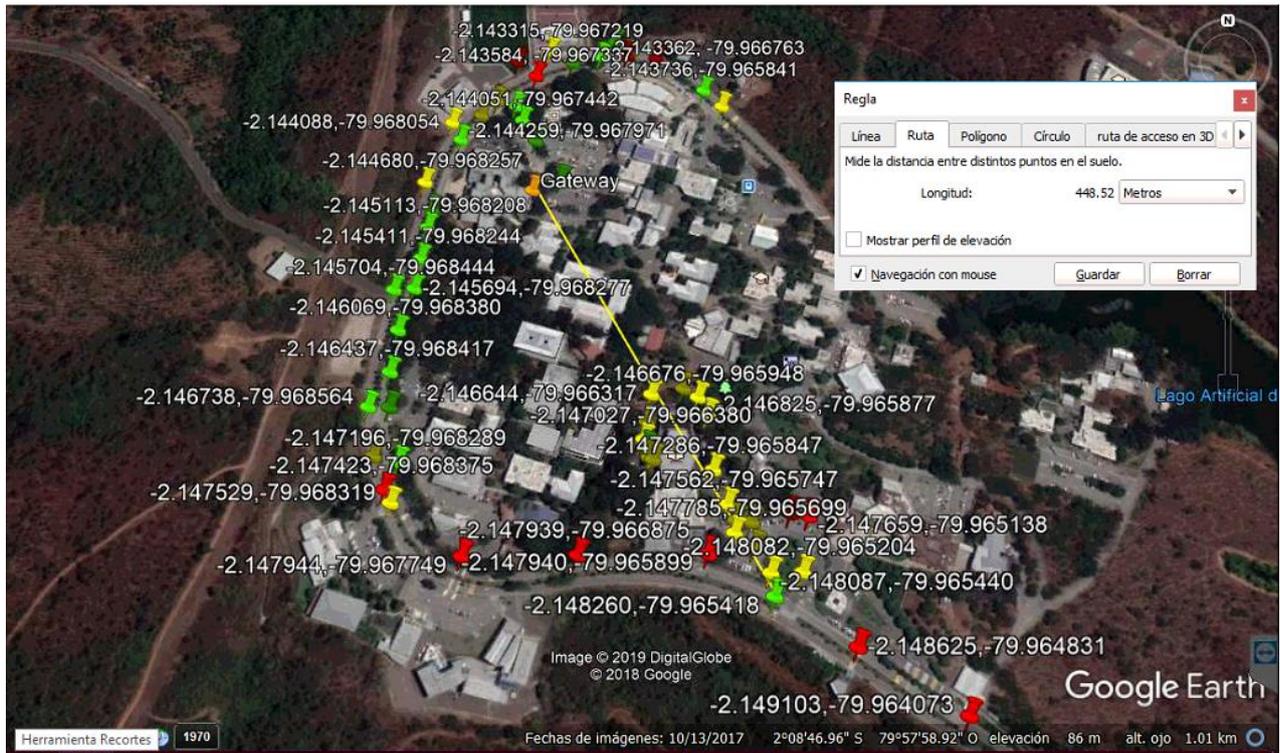


Figura 4.3. Distancia máxima de cobertura en LoRaWAN

Para analizar el tiempo entre los datos enviados se escogió una muestra de un total de 50 lecturas de parámetros de estado dando como resultado la gráfica que se detalla a continuación ver Figura 4.4. Existen aumentos de tiempo de espera de hasta un minuto esto se debe a que el vehículo se encontraba en movimiento. El tiempo promedio de paquetes que llegan al servidor es de 12 segundos.

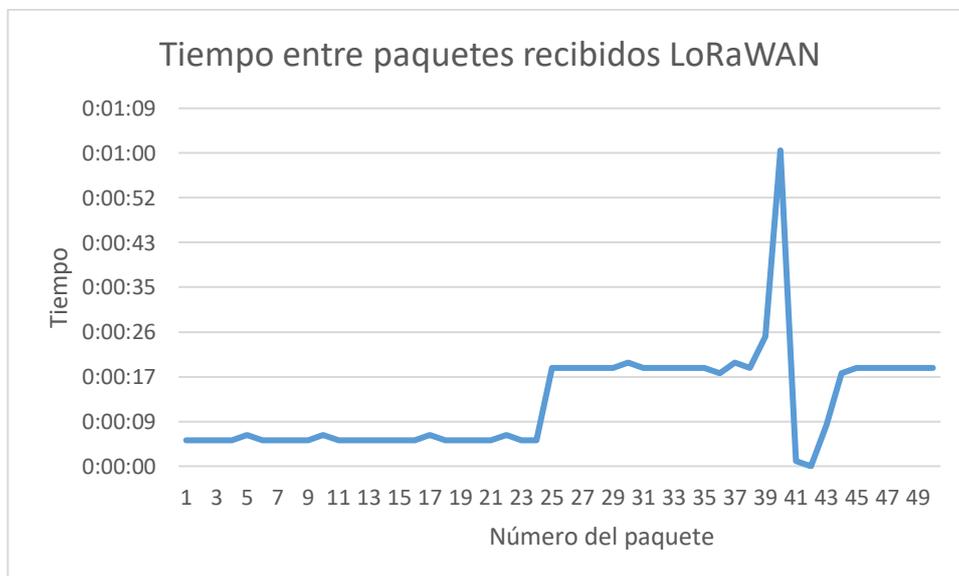


Figura 4.4 Tiempo entre paquetes recibidos LoRaWAN

4.2 Análisis GSM

Una vez conectados los módulos sim800L módulo transmisor y el módulo receptor en los nodos respectivamente se procedió a analizar el tiempo entre paquetes recibidos en el servidor. Dando como resultado que el tiempo promedio entre arribos de mensajes en el servidor es de aproximadamente 8 segundos. Los picos que se muestran en la Figura 4.5 denotan que el mensaje tardó más en llegar, esto se debe a problemas de alimentación propios del módulo sim800L por lo que deben reiniciarse para luego estabilizarse y reanudar la comunicación con el módulo transmisor. Durante la transmisión el módulo sim800L puede generar picos hasta 2A. Para la transmisión se debe conectar un paquete de mensajería ilimitada.

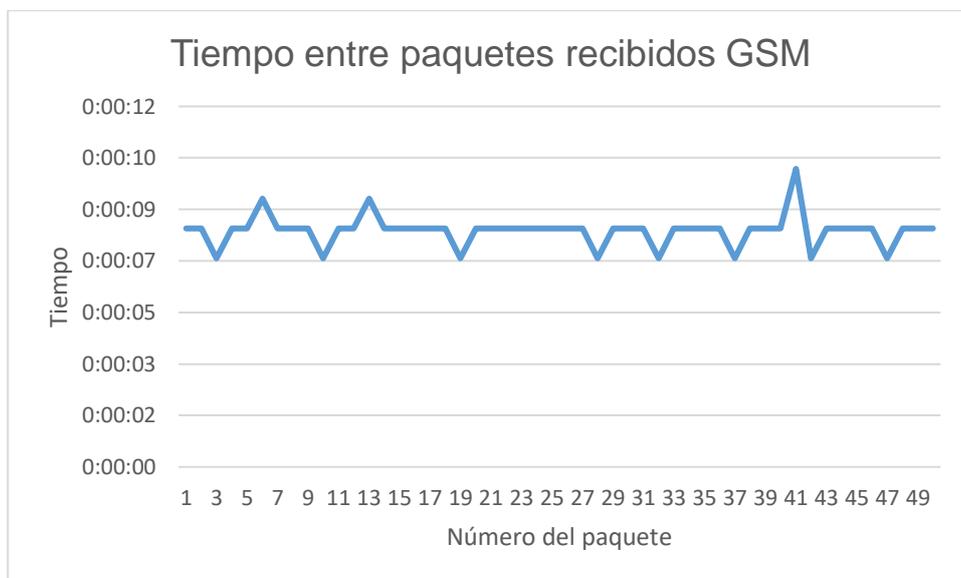


Figura 4.5 Tiempo entre paquetes recibidos GSM

4.3 Análisis de datos

Como se observa en la Figura 4.6 inicialmente los parámetros del estado del vehículo se encuentran en cero esto significa que el vehículo se encuentra apagado y por lo tanto el Sistema de diagnóstico OBD-II no está leyendo los parámetros de estado de la ECU.

MATRICULA	FECHA	HORA	VELOCIDAD	RPM	POS. ACELERADOR	TEMP. REFRIGERANTE	TEMP. AIRE	NIV. COMBUSTIBLE	CARGA MOTOR
GKN-900	2019-02-13	16:38:32	0	0	0	0	0	0	0
GKN-900	2019-02-13	16:38:37	0	0	0	0	0	0	0
GKN-900	2019-02-13	16:38:42	0	0	0	0	0	0	0
GKN-900	2019-02-13	16:38:47	0	0	0	0	0	0	0
GKN-900	2019-02-13	16:38:52	0	0	0	0	0	0	0
GKN-900	2019-02-13	16:38:58	0	0	0	0	0	0	0
GKN-900	2019-02-13	16:39:03	0	0	0	0	0	0	0
GKN-900	2019-02-13	16:39:08	0	0	0	0	0	0	0
GKN-900	2019-02-13	16:39:13	0	0	0	0	0	0	0
GKN-900	2019-02-13	16:39:18	0	0	0	0	0	0	0
GKN-900	2019-02-13	16:39:24	0	0	0	0	0	0	0
GKN-900	2019-02-13	16:39:29	0	0	0	0	0	0	0
GKN-900	2019-02-13	16:39:34	0	749	85	23.13	23.13	12.94	28

Figura 4.6 Página de monitoreo de la Plataforma del Cliente

En la Figura 4.7 se observa el monitoreo de los parámetros de estado del vehículo en tiempo real, como por ejemplo el parámetro velocidad se encuentra en cero ya que el vehículo estaba estacionado por lo tanto las revoluciones por minuto (RPM) deben oscilar en valor de 600 hasta 800.

MATRICULA	FECHA	HORA	VELOCIDAD	RPM	POS. ACELERADOR	TEMP. REFRIGERANTE	TEMP. AIRE	NIV. COMBUSTIBLE	CARGA MOTOR
GKN-900	2019-02-15	11:47:58	0.9	803	12.649	93	0	13.9	38.6471
GKN-900	2019-02-15	11:48:07	0.9	803	12.649	93	0	13.9	38.6471
GKN-900	2019-02-15	11:48:15	0.9	803	12.649	93	0	13.9	38.6471
GKN-900	2019-02-15	11:48:23	0.9	803	12.649	93	0	13.9	38.6471
GKN-900	2019-02-15	11:48:31	0.9	803	12.649	93	0	13.9	38.6471
GKN-900	2019-02-15	11:48:39	0.9	803	12.649	93	0	13.9	38.6471
GKN-900	2019-02-15	11:48:47	0.9	803	12.649	93	0	13.9	38.6471
GKN-900	2019-02-15	11:48:54	0.9	803	12.649	93	0	13.9	38.6471
GKN-900	2019-02-15	11:49:02	0.9	803	12.649	93	0	13.9	38.6471
GKN-900	2019-02-15	11:49:10	0.9	803	12.649	93	0	13.9	38.6471
GKN-900	2019-02-15	11:49:18	0.9	803	12.649	93	0	13.9	38.6471
GKN-900	2019-02-15	11:49:26	0.9	803	12.649	93	0	13.9	38.6471
GKN-900	2019-02-15	11:49:34	0.9	803	12.649	93	0	13.9	38.6471
GKN-900	2019-02-15	11:49:42	0.9	803	12.649	93	0	13.9	38.6471
GKN-900	2019-02-15	11:49:49	0.9	803	12.649	93	0	13.9	38.6471
GKN-900	2019-02-15	11:49:57	0.9	803	12.649	93	0	13.9	38.6471
GKN-900	2019-02-15	11:50:05	0.9	803	12.649	93	0	13.9	38.6471
GKN-900	2019-02-15	11:50:13	0.9	803	12.649	93	0	13.9	38.6471
GKN-900	2019-02-15	11:50:21	0.9	803	12.649	93	0	13.9	38.6471
GKN-900	2019-02-15	11:50:28	0.9	803	12.649	93	0	13.9	38.6471

Figura 4.7 Página de monitoreo de la Plataforma del Cliente

Para el almacenamiento de los parámetros de estado en tiempo real se utilizaron dos servidores ya que se necesitaban observar los datos recibidos y analizar el tiempo de recepción en el servidor para ambas tecnologías. El nodo receptor GSM tiene un módulo SIM800L que necesita de una red GSM, en cambio el servidor de la aplicación de monitoreo debe conectarse a Internet para acceder a la plataforma LoRaWAN y recibir los datos.

GKN-900	2019-02-15	10:58:06	799.2
GKN-900	2019-02-15	10:58:09	799.2
GKN-900	2019-02-15	10:58:17	799.2
GKN-900	2019-02-15	10:58:18	799.2
GKN-900	2019-02-15	10:58:24	799.2
GKN-900	2019-02-15	10:58:32	799.2
GKN-900	2019-02-15	10:58:34	825
GKN-900	2019-02-15	10:58:45	825
GKN-900	2019-02-15	10:58:45	825
GKN-900	2019-02-15	10:58:57	825
GKN-900	2019-02-15	10:58:58	825
GKN-900	2019-02-15	10:59:08	825
GKN-900	2019-02-15	10:59:09	795
GKN-900	2019-02-15	10:59:20	795
GKN-900	2019-02-15	10:59:31	795
GKN-900	2019-02-15	10:59:47	852
GKN-900	2019-02-15	11:00:06	852
GKN-900	2019-02-15	11:00:26	840.2
GKN-900	2019-02-15	11:00:45	840.2
GKN-900	2019-02-15	11:01:09	835
GKN-900	2019-02-15	11:01:24	821.7
GKN-900	2019-02-15	11:01:47	821.7
GKN-900	2019-02-15	11:02:04	821.7

Figura 4.8 Monitoreo de vehículos parámetro RPM Servidor Ubuntu

GKN-900	2019-02-13	22:48:25	803
GKN-900	2019-02-13	22:48:32	803
GKN-900	2019-02-13	22:48:40	803
GKN-900	2019-02-13	22:48:48	803
GKN-900	2019-02-13	22:48:57	803
GKN-900	2019-02-13	22:49:05	803
GKN-900	2019-02-13	22:49:13	803
GKN-900	2019-02-13	22:49:21	803
GKN-900	2019-02-13	22:49:29	803
GKN-900	2019-02-13	22:49:37	803
GKN-900	2019-02-13	22:49:44	803
GKN-900	2019-02-13	22:49:52	803
GKN-900	2019-02-13	22:50:00	803
GKN-900	2019-02-13	22:50:08	803
GKN-900	2019-02-13	22:50:16	803
GKN-900	2019-02-13	22:50:24	803
GKN-900	2019-02-13	22:50:32	803
GKN-900	2019-02-13	22:50:40	803
GKN-900	2019-02-13	22:50:48	803
GKN-900	2019-02-13	22:50:56	803

Figura 4.9 Monitoreo de vehículos parámetro RPM Servidor Raspbian

4.4 Análisis Financiero

El costo total de implementación del proyecto para cada tecnología de comunicación utilizada se detalla a continuación en las tablas 4.1 (LoRaWAN) y 4.2 (GSM). Para el análisis de costos por implementación en LoRaWAN no se considera el pago de un plan de Internet ya que el cliente actualmente cuenta con uno para actividades varias dentro la empresa.

Tabla 4.1 Costo total de implementación usando LoRaWAN

Descripción	Cantidad	Precio Unitario	Precio Total
Arduino LoRaWAN	39	\$77,00	\$3.003,00
GATEWAY LORAWAN	3	\$280,00	\$840,00
Escáner ELM327 OBD-II	39	\$20,00	\$780,00
Raspberry Pi 3	39	\$50,00	\$1.950,00
Fuente 5V-3A	40	\$11,00	\$440,00
	160	\$438,00	\$7.013,00

Tabla 4.2 Costo total de implementación usando GSM

Descripción	Cantidad	Precio Unitario	Precio Total
GSM SIM 800L	40	\$27,00	\$1.080,00
Chip CLARO	40	\$4,50	\$180,00
Plan Mensajería Ilimitada	40	\$12,00	\$480,00
Cables Jumpers (Tirasx40)	7	\$2,80	\$19,60
Fuente 5V-3A	40	\$11,00	\$440,00
Escáner ELM327 OBD-II	39	\$20,00	\$780,00
Raspberry Pi 3	40	\$50,00	\$2.000,00
	245	\$127,30	\$4.967,60

Para ambos casos se realizó una proyección de costos a seis meses, tomando en cuenta que existen algunos gastos que se hacen una única vez y otros que requieren realizarse mes a mes debido a contratos directos con las operadoras de telecomunicaciones, para que el sistema pueda seguir funcionando. Las tablas 4.1 y 4.2 detallan estas proyecciones para el uso de LoRaWAN y GSM respectivamente y adicionalmente se muestran gráficos del comportamiento de estas proyecciones en el tiempo representados en las Figuras 4.10 y 4.11.

Tabla 4.3 Proyección de costos LoRaWAN (6 meses)

Proyección LoRaWAN - 6 Meses					
1er Mes	2ndo Mes	3er Mes	4to Mes	5to Mes	6to Mes
\$7.013,00	\$7.013,00	\$7.013,00	\$7.013,00	\$7.013,00	\$7.013,00

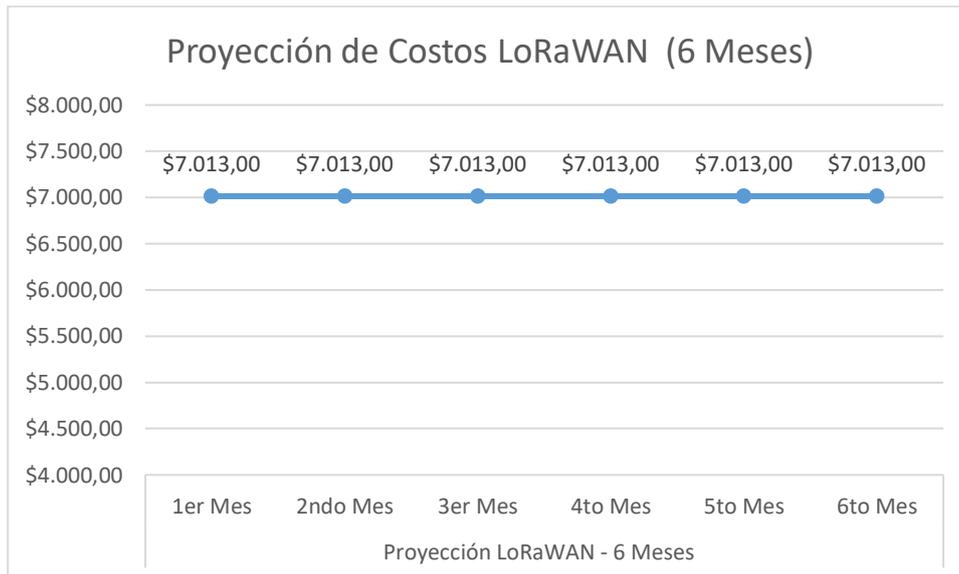


Figura 4.10. Representación gráfica de proyección de costos de LoRaWAN

Tabla 4.4. Proyección de costos GSM (6 meses)

Proyección GSM - 6 Meses					
1er Mes	2ndo Mes	3er Mes	4to Mes	5to Mes	6to Mes
\$4.967,60	\$5.447,60	\$5.927,60	\$6.407,60	\$6.887,60	\$7.367,60

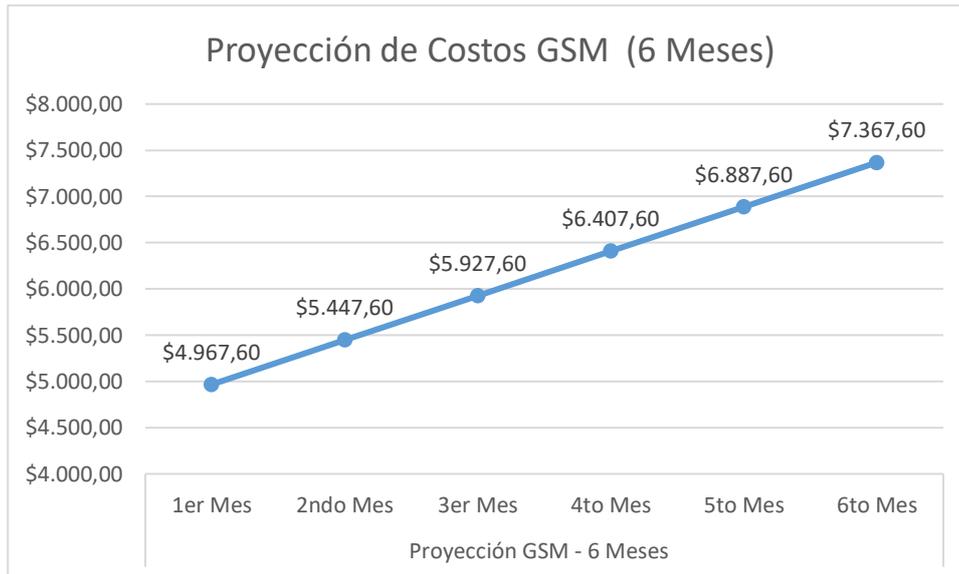


Figura 4.11. Representación gráfica de proyección de costos de GSM

4.5 Comparación técnica

- En la comunicación LoRaWAN los datos se transmiten encriptados por lo tanto es un protocolo seguro, además de que la plataforma The Things Network lleva un monitoreo de los dispositivos y aplicaciones que se encuentren registrados, mientras que en la tecnología GSM se debe establecer un acuerdo con la Empresa de Telecomunicaciones ya que se enviarán cada cierto tiempo una cantidad considerable de mensajes de texto, además por ser un protocolo antiguo puede sufrir ataques, como la clonación de tarjetas SIM.
- LoRaWAN necesita de Internet y GSM tener un servicio contratado con una Operadora Telefónica.
- De acuerdo a los análisis realizados en las secciones 4.3 y 4.2, el tiempo promedio de recepción de parámetros de estado del vehículo en tiempo real para las tecnologías implementadas tienen una diferencia de 2 segundos, siendo GSM de menor tiempo, esto se debe a que los paquetes se transfieren una red GSM y no necesitan de Internet mientras que en LoRaWAN los datos deben pasar a la plataforma TTN a través de Internet, encriptarse y luego llegar a la plataforma de monitoreo de vehículos. También la dependerá de la de conexión entre el Gateway e Internet.

- GSM tiene mejor cobertura, pero la transmisión y recepción dependerá de la calidad de fabricación del módulo o de la tarjeta SIM, conexiones, alimentación. Estos factores no se pueden controlar fácilmente.
- Los dispositivos LoRaWAN tienen un menor consumo de energía, los módulos GSM necesitan una alimentación adicional de 5V-2.5A. Además, se debe considerar que para el proceso de transmisión GSM los módulos deben estar estables, ya que se pueden generar picos de hasta 2A.

CONCLUSIONES Y RECOMENDACIONES

A continuación, se describen las conclusiones y recomendaciones del proyecto.

Conclusiones

- LoRaWAN y GSM son eficientes para la entrega de mensajes en tiempo real, ya que los datos son fiables y no sufren alteraciones en el proceso de transmisión y recepción.
- De acuerdo con el análisis financiero, el costo de implementación de la tecnología LoRaWAN es elevado, pero este pago se realiza una sola vez siendo esta la mejor opción para el sistema de monitoreo, ya que el cliente no tendría gastos fijos mensuales por concepto de mensajería ilimitada.
- LoRaWAN permite mayor control sobre la transmisión, ya que en las plataformas de red LoRaWAN se puede llevar un monitoreo remotamente del estado de los dispositivos utilizados y no se requiere de hacer una inspección al espacio físico donde se encuentren los módulos.
- Para la implementación de ambas tecnologías y lograr un mayor alcance de transmisión, los costos de cobertura son por parte del cliente.
- Para la implementación del Sistema de la plataforma de conectividad y monitoreo de diagnóstico de vehículos y de acuerdo a las necesidades del cliente se considera que LoRaWAN es mejor ya que no existen costos fijos mensuales, los datos viajan encriptados y se tiene un control de los dispositivos implementados de esta manera se podría conocer si un nodo está funcionando de manera correcta.

Recomendaciones

- Como mejora a futuro se podría incluir dentro de la plataforma del cliente una opción para leer códigos de error desde la ECU del vehículo y eliminarlos. Además de llevar a cabo un monitoreo de localización del vehículo.
- Las antenas de LoRaWan y GSM no están en igualdad de condiciones por lo que se recomienda instalar una antena LoRaWAN de mayor ganancia en el módulo transmisor además la cobertura dependerá de cuántas radios bases hayan sido desplegadas para que pueda transmitir a mayores distancias y de esta manera pueda ser comparable con GSM.
- Se debe tomar en cuenta que la implementación de la tecnología GSM puede presentar problemas de transmisión o recepción atribuibles a los módulos SIM800L, por lo que se requiere que se revise el estado físico de módulo o de la tarjeta SIM, conexiones, alimentación. Estos factores no se pueden controlar fácilmente.
- Se debe considerar que al implementar la tecnología GSM se debe utilizar un limitador de corriente, ya que en la transmisión al generarse picos de hasta 2A en ciertas ocasiones el módulo SIM800L puede sufrir averías porque puede requerir a la fuente de alimentación más corriente de la que puede soportar.

BIBLIOGRAFÍA

- [1] H.-J. Yun, S.-K. Lee y O.-C. Kwon, «IEEEEXPLORE,» 4 Octubre 2012. [En línea]. Available: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6316578&isnumber=6316561>.
- [2] D. Sosnowski y E. Gardetto, «United States Environmental Protection Agency,» Junio 2001. [En línea]. Available: http://lobby.la.psu.edu/_107th/093_OBD_Service_Info/Agency_Activities/EPA_OTAQ/Vehicle_inspection_and_maintenance.pdf.
- [3] U. S. E. P. Agency. [En línea]. Available: <https://www.epa.gov/>.
- [4] Cablematic.es, «Cablematic.es,» [En línea]. Available: <http://www.cablematic.es/OBD-OBDII-OBD2-EOBD/>.
- [5] W. Simbaña , J. Caiza, D. Chávez y G. López, «Revista Politécnica EPN,» Marzo 2016. [En línea]. Available: https://www.revistapolitecnica.epn.edu.ec/images/revista/volumen37/tomo1/Diseno_e_Implementacion_de_un_Sistema_de_Monitoreo_Remoto_del_Motor_de_un_Vehiculo_basado_en_Obd_II_y_la_plataforma_Arduino.pdf.
- [6] FreeASEStudyGuides.com, «FreeASEStudyGuides.com,» 2018. [En línea]. Available: <https://www.freeasestudyguides.com/electrical-troubleshooting-data-link-connector.html>.
- [7] E. A. JAVAZ, «equipoautomotrizjavaz.com,» [En línea]. Available: https://equipoautomotrizjavaz.com/datos_tecnicos/alimentacion-de-5-protocolos-de-obd2.pdf.
- [8] S. International, «SAE International,» [En línea]. Available: <https://www.sae.org/>.
- [9] R. Z., «Just Answer,» 24 08 2009. [En línea]. Available: <https://www.justanswer.com/dodge/2gfld-obd-located-2004-dodge-stratus.html>.
- [10] Fixya.com, «Fixya.com,» 7 08 2014. [En línea]. Available: http://www.fixya.com/cars/t24917579-cannot_locate_port_obd_ii_1997_isuzu.
- [11] T. Miller, «OBD Advisor,» 26 Diciembre 2018. [En línea]. Available: <https://www.obdadvisor.com/brief-intro-to-3-types-obd2-scanners/>.

- [12] C. Electronics, «CSS Electronics,» [En línea]. Available: <https://www.csselectronics.com/screen/page/simple-intro-obd2-explained/language/en>.
- [13] W. A. Simbaña Coyago, «Biblioteca Digital EPN,» Junio 2015. [En línea]. Available: <http://bibdigital.epn.edu.ec/bitstream/15000/10659/1/CD-6287.pdf>.
- [14] J. A. Ros Marín y O. Barrera Doblado, Sistemas electricos y de Seguridad y de Confortabilidad, Madrid: Paraninfo, 2011.
- [15] X-Engineer.org, «X-Engineer.org,» [En línea]. Available: <https://x-engineer.org/automotive-engineering/internal-combustion-engines/diagnostics/on-board-diagnostics-obd-modes-operation-diagnostic-services/>.
- [16] Danameyerautocare.com, «Danameyerautocare.com,» 12 01 2017. [En línea]. Available: <http://danameyerautocare.com/2017/01/12/pesky-dashboard-lights-mean-can-ignored/>.
- [17] L. A. TM, «Lora Alliance,» [En línea]. Available: <https://lora-alliance.org/about-lorawan>. [Último acceso: 19 12 2018].
- [18] A. corporativo, «Lorawan.es,» [En línea]. Available: <https://lorawan.es/index.html#msg-box8-4>. [Último acceso: 19 Diciembre 2018].
- [19] J. R. M. Fernández, «Sistemas Celulares. Estándar GSM,» Enero 2016. [En línea]. Available: https://www.researchgate.net/publication/308879041_Sistemas_Celulares_Estandar_GSM. [Último acceso: 19 Diciembre 2018].
- [20] DHGate.com, «DHGate.com,» [En línea]. Available: <https://www.dhgate.com/product/getProductImages.do?act=getProductImages&itemcode=414881687&pid=ff80808163d83acc0163f1dc77ba3fab>.
- [21] B. Whitfield, «Python-obd,» [En línea]. Available: <https://python-obd.readthedocs.io/en/latest/>.
- [22] Alibaba.com, «Alibaba.com,» [En línea]. Available: <https://spanish.alibaba.com/product-detail/sim800l-v2-0-5v-wireless-gsm-gprs-module-pcb-antenna-sim800l-module-60648299276.html>.
- [23] N. Element14. [En línea]. Available: <https://www.newark.com/the-things-network/ttn-un-915/silicon-manufacturer-microchip/dp/05AC1811>. [Último acceso: 2 Enero 2019].
- [24] L. Connectivity, «Laird Connectivity,» 2018. [En línea]. Available: <https://connectivity.lairdtech.com/wireless-modules/lorawan-solutions/sentrius-rg1xx-lora-enabled-gateway-wi-fi-bluetooth-ethernet>. [Último acceso: 10 Enero 2019].

- [25] S. Actron, «Manuals Directory,» 2004. [En línea]. Available: <http://www.manualsdir.com/manuals/35917/actron-9040-kal9040.html?page=3&original=1>.
- [26] K. McCord, «Diagnostic Trouble Codes,» de *Automotive Diagnostic Systems: Understanding OBD I and OBD II*, CarTech, 2011, pp. 16-17.
- [27] CaknowApp, «Medium.com,» 29 Enero 2017. [En línea].

ANEXOS

ANEXO A:

LISTA DE COMANDOS SOPORTADOS POR LIBRERÍA PY-OBd EN EL MODO 1

PID	Nombre	Descripción	Valor de Respuesta
0	PIDS_A	Supported PIDs [01-20]	bitarray
1	STATUS	Status since DTCs cleared	special
2	FREEZE_DTC	DTC that triggered the freeze frame	special
3	FUEL_STATUS	Fuel System Status	(string, string)
4	ENGINE_LOAD	Calculated Engine Load	Unit.percent
5	COOLANT_TEMP	Engine Coolant Temperature	Unit.celsius
6	SHORT_FUEL_TRIM_1	Short Term Fuel Trim - Bank 1	Unit.percent
7	LONG_FUEL_TRIM_1	Long Term Fuel Trim - Bank 1	Unit.percent
8	SHORT_FUEL_TRIM_2	Short Term Fuel Trim - Bank 2	Unit.percent
9	LONG_FUEL_TRIM_2	Long Term Fuel Trim - Bank 2	Unit.percent
0A	FUEL_PRESSURE	Fuel Pressure	Unit.kilopascal
0B	INTAKE_PRESSURE	Intake Manifold Pressure	Unit.kilopascal
0C	RPM	Engine RPM	Unit.rpm
0D	SPEED	Vehicle Speed	Unit.kph
0E	TIMING_ADVANCE	Timing Advance	Unit.degree
0F	INTAKE_TEMP	Intake Air Temp	Unit.celsius
10	MAF	Air Flow Rate (MAF)	Unit.grams_per_second
11	THROTTLE_POS	Throttle Position	Unit.percent
12	AIR_STATUS	Secondary Air Status	string
13	O2_SENSORS	O2 Sensors Present	special
14	O2_B1S1	O2: Bank 1 - Sensor 1 Voltage	Unit.volt
15	O2_B1S2	O2: Bank 1 - Sensor 2 Voltage	Unit.volt
16	O2_B1S3	O2: Bank 1 - Sensor 3 Voltage	Unit.volt
17	O2_B1S4	O2: Bank 1 - Sensor 4 Voltage	Unit.volt
18	O2_B2S1	O2: Bank 2 - Sensor 1 Voltage	Unit.volt
19	O2_B2S2	O2: Bank 2 - Sensor 2 Voltage	Unit.volt
1A	O2_B2S3	O2: Bank 2 - Sensor 3 Voltage	Unit.volt

1B	O2_B2S4	O2: Bank 2 - Sensor 4 Voltage	Unit.volt
1C	OBD_COMPLIANCE	OBD Standards Compliance	string
1D	O2_SENSORS_ALT	O2 Sensors Present (alternate)	special
1E	AUX_INPUT_STATUS	Auxiliary input status (power take off)	boolean
1F	RUN_TIME	Engine Run Time	Unit.second
20	PIDS_B	Supported PIDs [21-40]	bitarray
21	DISTANCE_W_MIL	Distance Traveled with MIL on	Unit.kilometer
22	FUEL_RAIL_PRESSURE_VAC	Fuel Rail Pressure (relative to vacuum)	Unit.kilopascal
23	FUEL_RAIL_PRESSURE_DIRECT	Fuel Rail Pressure (direct inject)	Unit.kilopascal
24	O2_S1_WR_VOLTAGE	O2 Sensor 1 WR Lambda Voltage	Unit.volt
25	O2_S2_WR_VOLTAGE	O2 Sensor 2 WR Lambda Voltage	Unit.volt
26	O2_S3_WR_VOLTAGE	O2 Sensor 3 WR Lambda Voltage	Unit.volt
27	O2_S4_WR_VOLTAGE	O2 Sensor 4 WR Lambda Voltage	Unit.volt
28	O2_S5_WR_VOLTAGE	O2 Sensor 5 WR Lambda Voltage	Unit.volt
29	O2_S6_WR_VOLTAGE	O2 Sensor 6 WR Lambda Voltage	Unit.volt
2A	O2_S7_WR_VOLTAGE	O2 Sensor 7 WR Lambda Voltage	Unit.volt
2B	O2_S8_WR_VOLTAGE	O2 Sensor 8 WR Lambda Voltage	Unit.volt
2C	COMMANDED_EGR	Commanded EGR	Unit.percent
2D	EGR_ERROR	EGR Error	Unit.percent
2E	EVAPORATIVE_PURGE	Commanded Evaporative Purge	Unit.percent
2F	FUEL_LEVEL	Fuel Level Input	Unit.percent
30	WARMUPS_SINCE_DTC_CLEAR	Number of warm-ups since codes cleared	Unit.count
31	DISTANCE_SINCE_DTC_CLEAR	Distance traveled since codes cleared	Unit.kilometer
32	EVAP_VAPOR_PRESSURE	Evaporative system vapor pressure	Unit.pascal
33	BAROMETRIC_PRESSURE	Barometric Pressure	Unit.kilopascal
34	O2_S1_WR_CURRENT	O2 Sensor 1 WR Lambda Current	Unit.milliampere
35	O2_S2_WR_CURRENT	O2 Sensor 2 WR Lambda Current	Unit.milliampere
36	O2_S3_WR_CURRENT	O2 Sensor 3 WR Lambda Current	Unit.milliampere

37	O2_S4_WR_CURRENT	O2 Sensor 4 WR Lambda Current	Unit.milliampere
38	O2_S5_WR_CURRENT	O2 Sensor 5 WR Lambda Current	Unit.milliampere
39	O2_S6_WR_CURRENT	O2 Sensor 6 WR Lambda Current	Unit.milliampere
3A	O2_S7_WR_CURRENT	O2 Sensor 7 WR Lambda Current	Unit.milliampere
3B	O2_S8_WR_CURRENT	O2 Sensor 8 WR Lambda Current	Unit.milliampere
3C	CATALYST_TEMP_B1S1	Catalyst Temperature: Bank 1 - Sensor 1	Unit.celsius
3D	CATALYST_TEMP_B2S1	Catalyst Temperature: Bank 2 - Sensor 1	Unit.celsius
3E	CATALYST_TEMP_B1S2	Catalyst Temperature: Bank 1 - Sensor 2	Unit.celsius
3F	CATALYST_TEMP_B2S2	Catalyst Temperature: Bank 2 - Sensor 2	Unit.celsius
40	PIDS_C	Supported PIDs [41-60]	bitarray
41	STATUS_DRIVE_CYCLE	Monitor status this drive cycle	special
42	CONTROL_MODULE_VOLTAGE	Control module voltage	Unit.volt
43	ABSOLUTE_LOAD	Absolute load value	Unit.percent
44	COMMANDED_EQUIV_RATIO	Commanded equivalence ratio	Unit.ratio
45	RELATIVE_THROTTLE_POS	Relative throttle position	Unit.percent
46	AMBIANT_AIR_TEMP	Ambient air temperature	Unit.celsius
47	THROTTLE_POS_B	Absolute throttle position B	Unit.percent
48	THROTTLE_POS_C	Absolute throttle position C	Unit.percent
49	ACCELERATOR_POS_D	Accelerator pedal position D	Unit.percent
4A	ACCELERATOR_POS_E	Accelerator pedal position E	Unit.percent
4B	ACCELERATOR_POS_F	Accelerator pedal position F	Unit.percent
4C	THROTTLE_ACTUATOR	Commanded throttle actuator	Unit.percent
4D	RUN_TIME_MIL	Time run with MIL on	Unit.minute
4E	TIME_SINCE_DTC_CLEARED	Time since trouble codes cleared	Unit.minute
4F	<i>unsupported</i>	<i>unsupported</i>	
50	MAX_MAF	Maximum value for mass air flow sensor	Unit.grams_per_second
51	FUEL_TYPE	Fuel Type	string
52	ETHANOL_PERCENT	Ethanol Fuel Percent	Unit.percent
53	EVAP_VAPOR_PRESSURE_ABS	Absolute Evap system Vapor Pressure	Unit.kilopascal
54	EVAP_VAPOR_PRESSURE_ALT	Evap system vapor pressure	Unit.pascal

55	SHORT_O2_TRIM_B1	Short term secondary O2 trim - Bank 1	Unit.percent
56	LONG_O2_TRIM_B1	Long term secondary O2 trim - Bank 1	Unit.percent
57	SHORT_O2_TRIM_B2	Short term secondary O2 trim - Bank 2	Unit.percent
58	LONG_O2_TRIM_B2	Long term secondary O2 trim - Bank 2	Unit.percent
59	FUEL_RAIL_PRESSURE_ABS	Fuel rail pressure (absolute)	Unit.kilopascal
5A	RELATIVE_ACCEL_POS	Relative accelerator pedal position	Unit.percent
5B	HYBRID_BATTERY_REMAINING	Hybrid battery pack remaining life	Unit.percent
5C	OIL_TEMP	Engine oil temperature	Unit.celsius
5D	FUEL_INJECT_TIMING	Fuel injection timing	Unit.degree
5E	FUEL_RATE	Engine fuel rate	Unit.liters_per_hour
5F	<i>unsupported</i>	<i>unsupported</i>	

ANEXO B:

FLOTA DE VEHÍCULOS DE CONDUESPOL



Figura Anexo B1. Flota de Vehículos de Conduespol



Figura Anexo B2. Chevrolet Aveo Activo de Conduespol

ANEXO C:

PLAN DE PRUEBAS

1. ¿Qué es lo que hay que probar?

Un Sistema

2. ¿Qué se va a probar de ese objeto?

HARDWARE

- Pruebas de conectividad dispositivo escáner OBD II con raspberry pi 3 a través de bluetooth.
- Revisar el sistema de alimentación de todos los componentes a utilizar.
- Sectorizar e indicar los lugares con mayor latencia en el campus Prosperina.
- Probar el radio de cobertura LoRaWAN en el campus.
- Pruebas de recepción de mensajes mediante GSM.

SOFTWARE

- Analizar la integridad de los datos recibidos del dispositivo escáner a la plataforma.
- Recepción de los datos en el servidor que se encuentra en la nube.
- Inserción y revisión de datos utilizando el Aplicativo Web.

3. ¿Cómo se lo va a probar?

- Pedir permiso a Conduespol para acceder los vehículos de su flota.
- Verificar la conectividad del escáner OBD II con la raspberry en uno de los vehículos, para esto se enviarán comandos al puerto serial asignado para observar su estado activo.
- Activar el bluetooth de diferentes dispositivos mientras se realiza la conexión bluetooth y comprobar si existe interferencia.
- Luego, comparar los datos del sistema con valores obtenidos mediante la conexión de escáner obd II con una aplicación disponible en la tienda.
- Configuración de los equipos: Servidor de Aplicación Web y de Base de Datos.

- Armar la red y revisar que los equipos conectados se encuentren en buen estado, ya sea midiendo el voltaje en los terminales para los módulos, observar el estado activo de color verde en la plataforma local (para LoRaWAN) o comandos “AT” en el terminal para comprobar su estado activo (para GSM) o enviando comando de ping para verificar conectividad con el servidor.
- Insertar datos en el Aplicación Web con usuarios registrados previamente.
- Filtrar datos del estado de vehículos en el Aplicativo Web.

4. *¿Cuándo se va a probar?*

El vehículo se encuentra a disposición: 2 días la semana (lunes y martes) en el horario de 2:00 - 4:30 pm.

5. *¿Cuáles son los criterios para comenzar con las pruebas?*

- Sistema debe estar completo para comparar ambas tecnologías.
- El sistema de base de datos debe estar completo.
- Disponibilidad de vehículos.
- Manual de usuario.

6. *¿Cuáles son los criterios para la finalización de las pruebas?*

- Sobre calentamiento del motor
- Si no establece la conexión GSM y LoRaWAN
- Ambientes con mucha interferencia
- Saturación en los puertos serial por el envío y recepción de datos
- Aplicativo web no disponible
- Recepción de datos erróneos

ANEXO D:

VALIDACIÓN DEL CLIENTE CONDUESPOL



gustavo alonso ortiz dirtiest GreSham <gustavortiz21@gmail.com>

Hoy, 17:32

Mayken Yusselin Salavarría Tutiven; Andrea Sheyla Cardenas Sumba

Responder a todos

Buenas Tardes

Proyecto presentado para la medición de rangos utilizando un dispositivo OBD II, muy bueno para nuestra área de trabajo por el cual podremos saber las mediciones en tiempo real de los vehículos a la vez saber las falencias que se puedan presentar

Observando las dos propuestas de que tecnología podremos utilizar mi opinion personal seria la de LORAWAN por ser mas factible y acaparar costos ya que solo seria un solo pago en las instalaciones y las redes solo se encargaría de recibir información de los vehículos y enviarlas al cpu principal que este seria el que se encuentre en el area de Logistica.

La otra opción el costo seria muy alto ya que tendremos que adquirir planes para cada vehiculo para el envio de mensajes de texto que seria receptado directamente al cpu en el area de logistica.

Una recomendación ademas de saber las mediciones, el manejo de vehiculo si acelera mucho, a cuantas revoluciones coloca un cambio, seria mucho mejor trabajar con un Dispositivo OBD con GPS para saber la ubicación real de las unidades al momento de salir

Debemos de analizar la propuesta y comparar en el mercado costos tiempo de garantía utilidades, ademas este proyecto esta enfocado dentro de los que seria toda la área de Campus Espol, en el caso de que los vehículos salgan del rango estipulado para captar la información y medición de parametros del vehiculo tendríamos que realizar otra inversión para obtener nuevas antenas ubicadas en lugares mas altos

Ing.Gustavo Ortiz