



**ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**  
**Facultad de Ingeniería en Electricidad y Computación**

“DISEÑO DE UNA RED INALÁMBRICA DE SENSORES PARA  
ALERTAR ACCIDENTES AUTOMOVILÍSTICOS EN UNA  
AUTOPISTA”

**INFORME DE MATERIA INTEGRADORA**

Previa a la obtención del Título de:

**INGENIERO EN TELEMÁTICA**

CHRISTOPHER SERGEY BAIDAL MACIAS

DAVID STEVEN SALAZAR VILLEGAS

GUAYAQUIL – ECUADOR

AÑO: 2017

## AGRADECIMIENTOS

A Dios por ser el guía y soporte durante todos estos años.

A todas las personas que estuvieron conmigo en esta ardua jornada inyectándome optimismo y buenos consejos, de ellos también es el triunfo.

¡A mi familia y amigos, los amo!

¡Hic Sunt Dracones!

David Steven Salazar Villegas.

A Dios por ser el creador de nuestro mundo.

A mis padres por darme la oportunidad de salir adelante y ser un mejor humano.

A mis amigos quienes más saben lo que me ha costado.

A todos quienes formaron parte de mi proceso de educación en esta universidad.

Christopher Sergey Baidal Macias

## DEDICATORIA

A mi madre, nadie se lo merece más que ella.

David Steven Salazar Villegas.

A quienes en mi vida han estado a mi lado en todo mi camino, paso a paso, sin dejarme vencer de las adversidades, mis padres.

Christopher Sergey Baidal Macias

## TRIBUNAL DE EVALUACIÓN



Ph.D. Rebeca Estrada

PROFESOR EVALUADOR



Mg. Nestor Arreaga

PROFESOR EVALUADOR

## DECLARACIÓN EXPRESA

"La responsabilidad y la autoría del contenido de este Trabajo de Titulación, nos corresponde exclusivamente; y 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"



Christopher Sergey Baidal Macias



David Steven Salazar Villegas

## RESUMEN

Los siniestros viales en América Latina generan cada año cifras alarmantes. Según La Federación Internacional del Automóvil (FIA) [1] Ecuador es uno de los países que lidera los primeros puestos dentro de Latinoamérica pues en el año 2014 fue el país con la tasa de siniestralidad más preocupante con 17 personas fallecidas cada 100.000 habitantes. La falta de controles y la mala conducta al conducir están entre los principales problemas de seguridad vial que se enfrentan, detectándose con más frecuencia en las provincias de Pichincha, Guayas y Azuay [2].

El proyecto presentado a continuación busca integrar varias tecnologías con el fin de crear una red inalámbrica de sensores que permita detectar e informar en tiempo real siniestros vehiculares ocurridos en carreteras, ya que dentro de las mismas existen áreas con cobertura celular nula, la cual es un impedimento a la hora de solicitar a entidades de socorro la ayuda pertinente.

Para la implementación del prototipo se utilizaron herramientas basadas en el paradigma open source, como el microcontrolador Arduino nano, sensores de distancia (HC-SR04) y acelerómetro (MMA7361), sistema embebido Raspberry PI 3 modelo B en conjunto con módulos XBee PRO S2 que actúan como nodos de comunicación dentro de una red Zigbee que a su vez se comunica a una WLAN para lograr conectividad con el servidor.

Para el almacenamiento, procesamiento y visualización de los datos se implementó un servidor de base de datos MySQL y una aplicación web desarrollada en NODE.JS la cual permite realizar un monitoreo en tiempo real de los nodos y ejecutar consultas a la base de datos.

Para poder detectar de manera eficiente los choques se realizó un análisis de los datos obtenidos por medio de los sensores, para el sensor acelerómetro el análisis se centró en cada una de las 3 dimensiones mientras que para el sensor de distancia se estableció un disparador lógico al detectar distancias menores e iguales a los 10 cm.

Finalmente, se realizó un análisis de resultados del prototipo funcionando dentro de un ambiente real, para la consecución del mismo se diseñaron y ejecutaron pruebas de cobertura según distancia y pruebas de latencia, se obtuvieron valores de coberturas radiales de hasta 400 metros por nodo y un valor promedio de 3 segundos de latencia en la comunicación inalámbrica a medida que el prototipo se alejaba de la zona de cobertura.

## ÍNDICE GENERAL

AGRADECIMIENTOS.....	ii
DEDICATORIA.....	iii
TRIBUNAL DE EVALUACIÓN.....	iv
DECLARACIÓN EXPRESA.....	v
RESUMEN.....	vi
ÍNDICE GENERAL.....	viii
INDICE DE TABLAS.....	x
CAPÍTULO 1.....	1
1. MARCO GENERAL.....	1
1.2. Descripción del problema.....	1
1.3. Justificación.....	2
1.4. Objetivos.....	4
1.4.1. Objetivo general.....	4
1.4.2. Objetivos específicos.....	4
1.5. Resultados esperados.....	4
1.6. Alcance.....	5
CAPÍTULO 2.....	6
2. ESTADO DEL ARTE.....	6
2.1. Sistemas de asistencia vial.....	6
2.2. Red de sensores inalámbricos (WSN).....	7
2.3. Tarjeta de desarrollo.....	8
2.3.1. Microcontrolador Arduino.....	9
2.3.2. Sistema embebido Raspberry PI.....	11
2.4. El sensor.....	12
2.4.1. Tipos de sensores.....	12
2.4.2. Sensor ultrasónico HC-SR04.....	13
2.4.3. Sensor acelerómetro MMA 7361.....	13
2.4.4. Sistema de posicionamiento global.....	14
2.5. Comunicación.....	15

2.5.1.	LAN inalámbrica o WLAN .....	15
2.5.2.	ZigBee basado en IEEE 802.15.....	16
2.6.	Base de datos .....	17
2.7.	Aplicación web .....	18
CAPÍTULO 3.....		19
3.	DESARROLLO.....	19
3.1.	Descripción del escenario .....	19
3.2.	Arquitectura general de la red inalámbrica de sensores.....	20
3.3.	Diseño del dispositivo sensor .....	20
3.4.	Diseño del concentrador .....	23
3.5.	Diseño de red de comunicación estructurada.....	24
3.5.1.	Comunicación sensor-concentrador.....	24
3.5.2.	Comunicación concentrador-servidor.....	25
3.6.	Diseño del Sistema de información .....	26
3.6.1.	Modelo entidad relación de la base de datos.....	27
3.6.2.	Aplicación web .....	30
3.7.	Arquitectura de la red inalámbrica robusta .....	33
CAPÍTULO 4.....		37
4.	ANÁLISIS DE RESULTADOS.....	37
4.1.	Visión general de la plataforma de experimentos .....	37
4.2.	Resultados empíricos de los sensores .....	38
4.3.	Análisis de cobertura vs distancia. ....	40
CONCLUSIONES Y RECOMENDACIONES .....		43
BIBLIOGRAFÍA.....		45
ANEXOS .....		48

## INDICE DE TABLAS

Tabla 1: Tipos de sensores de acuerdo al fenómeno a medir.....	12
Tabla 2: Asignación de PINES del sensor.....	22
Tabla 3: Diccionario de datos de la tabla nodo. ....	28
Tabla 4: Diccionario de datos de la tabla conductor.....	28
Tabla 5: Diccionario de datos de la tabla vehículo. ....	29
Tabla 6: Diccionario de datos de la tabla accidente. ....	29
Tabla 7: Mejoras del prototipo para una red robusta.....	34
Tabla 8: Datos recogidos por los sensores según la distancia.....	39
Tabla 9: Prueba de cobertura según distancia.....	41

# CAPÍTULO 1

## 1. MARCO GENERAL

### 1.1. Antecedentes

En el presente capítulo se expondrá la problemática que existe al producirse un accidente de tránsito en carreteras, en la mayoría de las ocasiones el tiempo de espera desde que se suscita el siniestro hasta que alguna unidad de socorro asista al lugar de los hechos suele ser un tiempo muy prolongado provocando que las lesiones en las víctimas se vean agravadas al no ser atendidas a tiempo, además se harán uso de estadísticas proporcionadas en el sitio web de la Agencia Nacional de Tránsito que permitirán justificar la validez del proyecto formulando de esta manera sus requerimientos y alcance.

### 1.2. Descripción del problema

De acuerdo a las estadísticas proporcionadas en el sitio web de la Agencia Nacional de Tránsito (ANT), en el año 2016 se registraron un total de 30.269 accidentes de tránsito a nivel nacional. Desde el mes de enero hasta el mes de marzo del año 2017 se han registrado a nivel nacional un total de 7.123 siniestros automovilísticos. De ese total 3.687 corresponden a accidentes en los cuales se vio involucrado algún tipo de choque (choque lateral, estrellamiento, choque posterior, rozamiento, choque frontal, colisión) (ver figura 1.1).



**Figura 1.1: Serie cronológica de siniestros de tránsito enero 2016 – marzo 2017 [3].**

Muchos de estos siniestros resultaron en muerte, la cual pudo ser evitada con una atención médica oportuna. Por lo general este tipo de accidentes en carreteras son alertados a las autoridades mediante una llamada telefónica de algún individuo ubicado en la escena del accidente, pero en muchas ocasiones debido a nervios o a desorientación la persona testigo del accidente puede dar información errónea de la situación, una de ellas es la ubicación exacta del accidente, número de personas involucradas, etc.

Dificultando así el envío de unidades de rescate, nuestro proyecto busca aliviar principalmente ese problema, ofreciendo información de primera mano como fecha, hora, coordenadas exactas del siniestro, información del vehículo y su conductor a las respectivas unidades de socorro con el fin de facilitar su trabajo. Todo esto bajo una plataforma que funcione en tiempo real.

### 1.3. Justificación

Las redes de sensores son un campo de estudio actualmente en auge, el cual evoluciona rápidamente debido al gran interés suscitado en los últimos años. Son un concepto relativamente nuevo dentro de la adquisición y tratamiento de datos

el cual está estrechamente relacionado con el paradigma de Internet of things (IoT) [4], con muy diversas aplicaciones como la automatización de las instalaciones industriales, comerciales, agrícolas, residenciales, transporte, etc. Así como su uso en Smart Cities para poder medir diversas variables ambientales proveyendo así un ambiente amigable y un crecimiento económico sustentable a la ciudad.

Esta tecnología de sensores ha evolucionado siguiendo la misma tendencia de los sistemas embebidos los cuales cada vez son más pequeños, de menor consumo, con mayor potencia de procesamiento y mayor capacidad de almacenamiento, llegando así al punto de proporcionar dispositivos que difícilmente pueden diferenciarse de los nodos computacionales, incorporando capacidades cognitivas y de comunicación con las que llegan a establecer verdaderas redes de información.

Una de las aplicaciones de redes de sensores se encuentra relacionada con el entorno vehicular y los sistemas de transporte, se ha ido concentrando muchos esfuerzos durante los últimos años para intentar lograr nuevos servicios que reduzcan el número de accidentes y muertes, anticipen a los conductores ante posibles problemas en la ruta y/o provean de dicha información durante la marcha.

Muchos de estos servicios hacen uso de sensores y redes de sensores incorporadas a los vehículos e infraestructuras a lo largo de la carretera, para captar la información necesaria y hacer uso de la misma al tomar una decisión ya sea ante algún siniestro o para mejorar la calidad de vida de las personas que viven en los alrededores.

Actualmente no existe algún tipo de protocolo a seguir cuando se suscita un accidente, las personas por lo general llaman a emergencia sin tener información de primera mano cómo ubicación exacta, hora en la cual ocurrió el accidente, el tipo de vía (autopista, vía rápida, urbana, etc.), el tipo de vehículo, número de placa, número de identidad de la persona propietaria del mismo, edad, etc.

Toda esa información es útil para una rápida acción y evaluación de siniestros viales según la OMS (Organización Mundial de la Salud) en su manual de seguridad vial para decisores y profesionales.

## **1.4. Objetivos**

### **1.4.1. Objetivo general**

Diseñar una red de sensores que permita detectar una colisión entre dos o más vehículos, la misma que será capaz de comunicarse a un servidor con el fin de que las unidades de socorro pertinentes cuenten con información en tiempo real cuando un siniestro se suscite.

### **1.4.2. Objetivos específicos**

- Validar los estándares y tecnologías apropiadas para la implementación de redes de sensores inalámbricas.
- Analizar e implementar una topología de red para la adecuada comunicación entre los nodos y servidor.
- Diseñar, crear un prototipo y verificar el funcionamiento del sistema.

## **1.5. Resultados esperados**

Como se ha mencionado en apartados anteriores, actualmente en el Ecuador no existe un protocolo a seguir en términos de comunicación eficiente o herramientas colaborativas que permitan reportar siniestros automovilísticos, además un suceso de este tipo debe ser atendido de la manera más eficaz y rápida posible. El tiempo que demora el atender lesiones, así como el prestar asistencia a las víctimas de un siniestro solo aumenta la gravedad de las lesiones. Existen plazos críticos en la atención a brindar tras un accidente de tránsito, unos minutos de demora pueden marcar una gran diferencia entre la vida y la muerte.

Este proyecto propone realizar un prototipo basado en redes de sensores inalámbricos que permita detectar una colisión entre dos automóviles, se registre una bitácora de los sucesos y a la vez permita mediante una aplicación web

informar a las autoridades pertinentes sobre el accidente, brindando información antes mencionada logrando así una rápida respuesta que permita salvar vidas.

A partir del prototipo se analizarán los cambios que deberá tener el diseño del mismo para que sea implementado en un tramo dentro de una carretera del país, inclinándose por encontrar la solución más robusta, fiable, de bajo coste y accesible en el mercado, dando conclusiones y recomendaciones del sistema a gran escala, dejando así la puerta abierta para que personas que comparten el mismo interés puedan innovar la solución planteada.

### **1.6. Alcance**

El alcance de este proyecto será para que pueda ser implementado en la Escuela Superior Politécnica del Litoral, dentro del tramo Coliseo - EDCOM dando pautas para futuras implementaciones en las principales carreteras de la provincia.

En base a esta investigación se podrá desarrollar un sistema de redes inalámbricas más fiables y con redundancia a fallos, dejando en claro la infraestructura y el encapsulamiento para el envío de paquetes a través de la red de sensores desplegada.

Esta investigación procura ser un referente para el análisis, diseño e implementación de redes de sensores inteligentes, fomentando así el desarrollo tecnológico en el país, el mismo que muestra un entorno urbano que demanda una mejor calidad de vida, calidad de los servicios y una administración de los recursos públicos de manera eficaz, incentivando a usar herramientas de tecnologías de la información y comunicación las cuales permitan impactar en el desarrollo económico del país

## CAPÍTULO 2

### 2. ESTADO DEL ARTE

#### 2.1. Sistemas de asistencia vial

Debido a que la principal causa de los accidentes de tránsito es el factor humano ya sea por algún fallo, distracción o mal comportamiento del conductor, tanto los gobiernos nacionales como los fabricantes de vehículos han optado por implementar sistemas que puedan reducir los accidentes ocasionados por negligencia humana, entre las opciones comúnmente usadas tenemos el control de velocidad, sistema de frenada de emergencia automática, asistente de mantenimiento en el carril, etc.

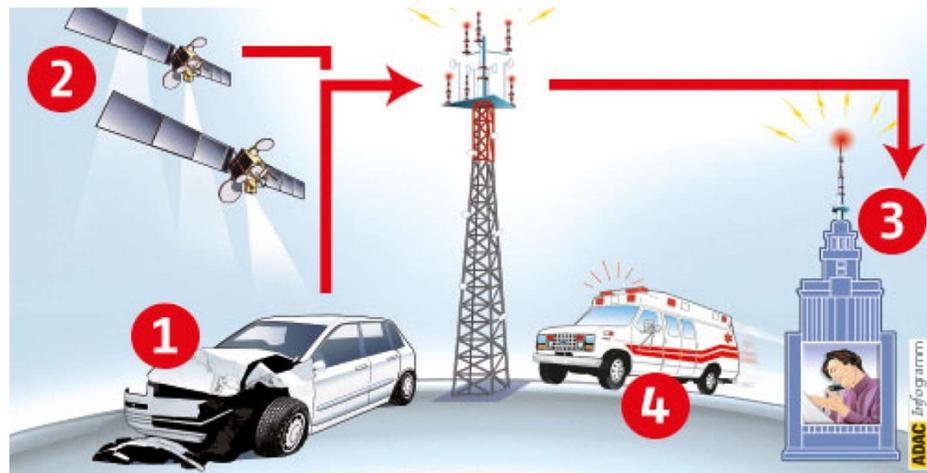
A continuación, se analiza el caso de un sistema integral de asistencia vial denominado eCall el mismo que se encuentra implementado en el continente europeo.

#### **ECall**

En el año 2015 el parlamento europeo votó en favor a la regulación eCall, la cual requiere que todos los vehículos que salgan a la venta se encuentren equipados con la tecnología eCall a partir del año 2018 [5].

El sistema eCall es un perfecto ejemplo de que la unión europea (EU) apoya proyectos que desarrollan soluciones tecnológicas las cuales permiten salvar vidas, en el caso de que ocurra un accidente vehicular, eCall reporta la ubicación exacta a los servicios de emergencia, además de la hora del indicio aun cuando el conductor se encuentre inconsciente para poder realizar una llamada.

El servicio de eCall también puede ser manejado de forma manual presionando un botón dentro del vehículo o por algún testigo del accidente en caso de haberlo, una vez que la alerta es disparada se realizará una llamada de forma automática a los números de socorro.



**Figura 2.1: Descripción de las partes correspondientes al servicio eCall. Fuente: [6]**

Como se puede apreciar en la figura 2.1, tenemos: 1). El accidente vehicular ocurre y una alerta es enviada. 2). La alerta es receptada satelitalmente y reenviada por los canales de comunicación existentes, esta puede ser retransmitida por distintas radios bases. 3). El operador del sistema de ayuda recibe la alerta y asigna una unidad de socorro al siniestro. 4). La unidad de socorro se encuentra en camino al área del accidente [6].

## 2.2. Red de sensores inalámbricos (WSN)

Las redes de sensores son un campo de estudio actualmente en auge, estas redes permiten ligar el mundo real con el mundo digital capturando y evaluando fenómenos o actividades que se suscitan diariamente a nuestro alrededor, convirtiendo todos estos datos en información la cual puede ser almacenada y procesada con el fin de realizar un análisis que permita tomar la mejor decisión para resolver la problemática [7].

Las redes de sensores pueden proveer un beneficio a la sociedad dada sus diferentes aplicaciones. Pueden usarse para prevenir fallas de infraestructura, monitorear bosques, incrementar la productividad de las empresas, evaluar el estado de salud en pacientes, mejorar la seguridad y permitir el desarrollo de nuevas tecnologías como las casas inteligentes entre otros. Hoy en día miles de redes de sensores están siendo usadas para monitorear, modelar y pronosticar contaminación ambiental e inundaciones, controlar el uso de agua y pesticidas que permitan mejorar la salud en los cultivos e incrementar la producción.

Dado que las redes de sensores recolectan grandes cantidades de datos, los mismos pueden ser evaluados para convertirse en información, si la información es compartida entre pares puede generar conocimiento y la unión de la experiencia con el conocimiento genera sabiduría, pero todo comienza con la adquisición de datos (Figura 2.2).



**Figura 2.2: Los seres humanos convierten datos en sabiduría. [8].**

“Los datos son la materia prima que se procesa para conseguir la información. Los datos individuales por sí mismos no son muy útiles, pero los grandes volúmenes de datos pueden identificar tendencias y patrones” [9]

### 2.3. Tarjeta de desarrollo

Los microcontroladores de bajo coste han facilitado en los últimos años la fabricación de pequeños sistemas en los que principalmente se maneja la

manipulación de señales digitales y analógicas, permitiendo así la integración del hardware con el software.

Las tarjetas de desarrollo son una placa de circuito impreso con múltiples puertos tanto digitales como análogos y un microcontrolador que se encuentra embebido dentro de la misma, esto permite facilitar e introducir a los usuarios en proyectos multidisciplinarios combinando electrónica y programación, aconsejando simplemente tener conocimiento de las bases fundamentales de estos temas.

Una de las ventajas de trabajar con estas tarjetas es que facilitan el proceso de diseño, impresión y ensamblaje de una placa de circuito impreso propia, invirtiendo así el tiempo ahorrado en la elaboración de algoritmos y estrategias de solución que permitan atacar el problema.

Entre las tarjetas disponibles en el mercado encontramos las que se detallan en el apartado Arduino en 2.3.1. y Raspberry en 2.3.2.

### **2.3.1. Microcontrolador Arduino**

La tarjeta de desarrollo Arduino se caracteriza por ser enteramente open-source lo cual significa que todos los diagramas para su fabricación se encuentran disponibles para todo el mundo sin excepción alguna. Basados en hardware y software consiste de una placa de circuito impreso con puertos analógicos y digitales y un microcontrolador facilitando el desarrollo de sistemas que interactúan con diferentes tipos de señales, a dichas tarjetas se les puede adicionar extensiones como sensores con los cuales se pueden controlar múltiples salidas para que interactúen con luces, motores y otros artefactos.

Estas tarjetas pueden ser programadas con la ayuda de un IDE gratuito y desarrollado por la misma compañía, aquí se puede interactuar con las diferentes partes de la tarjeta como puertos de entrada, salida y sensores en caso de haberse añadido, todo esto bajo el paradigma de programación lineal.

Al ser un dispositivo open hardware y software el soporte con la comunidad de desarrolladores siempre está en constante actualización para los nuevos integrantes de la familia, ayudando así a la educación de la sociedad estudiantil

y desarrolladores de plataformas abiertas. Entre los integrados más comunes en el mercado tenemos: (ver figura 2.3).



**Figura 2.3: Diferentes placas microcontroladores entre los cuales tenemos: A) Arduino UNO B). Pícaro+ C). Arduino MEGA y D). Arduino Micro [10]**

Arduino UNO R3.- integra un microcontrolador ATmega328 creado por Atmel dentro de la familia mega AVR basado en la combinación de 32kB ISP memoria flash con la capacidad de escribir mientras lee, entrada de voltaje entre 7-12V y cuenta con 14 pines digitales de entrada o salida y 6 salidas PWM, adicional a eso 6 entradas analógicas y cuenta con una memoria de 32k, su velocidad es de 16Mhz con ayuda de su reloj de cristal [11].

Arduino MEGA 2560 R3.- integra un microcontrolador Atmega2560 creado por Atmel dentro de la familia mega AVR basado en la combinación de 256kB ISP memoria flash con la capacidad de escribir mientras lee, entrada de voltaje entre 7-12V y cuenta con 54 pines digitales de entrada o salida y 14 salidas PWM, adicional a eso 16 entradas analógicas y cuenta con una memoria de 256k, su velocidad es de 16Mhz con ayuda de su reloj de cristal [12].

Arduino PICARO+. - una tarjeta Arduino Uno compatible, con un socket que permite la incorporación de los distintos módulos inalámbricos con el formato XBee como, por ejemplo: WiBee (WiFi), BlueBee (Bluetooth), GPRSbee (celular GPRS/GSM) incluyendo los módulos XBee Serie 1, Serie 2 y PRO. Cuenta con un microprocesador Atmega328P AU conector USB mini B, con 14 pines digitales y 6 analógicos reflejados, adicional cuenta con resistencias pullup disponibles en pines A4 y A5, fuente de voltaje 3.3V a 500mA y socket XBEE [13].

De todos estos microcontroladores se escogió el arduino micro debido a que consta de los pines necesarios para el prototipo y además por su reducido tamaño.

### 2.3.2. Sistema embebido Raspberry PI

Es un computador de placa única (SBC) esto quiere decir que contiene todo o la mayor parte de componentes de un computador, basada en ARM, se ejecuta en GNU/Linux OS Raspbian basado en Debian, cuenta con un procesador Quadcore 1.2 GHz Broadcom BCM2837 64 bit CPU con 1Gb RAM también con un incorporado de WIFI y Bluetooth, además 40 pines GPIO 4 USB 2 puertos, HDMI tamaño completo, conectividad CSI cámara y DSI pantalla. (ver figura 2.4).



Figura 2.4: Raspberry pi 3 modelo B [14]

## 2.4. El sensor

Un sensor es un dispositivo electrónico que tiene la capacidad de recopilar información acerca de los diferentes fenómenos y eventos que se suscitan en el ambiente. Esta tarea se realiza gracias a los transductores que llevan incorporados, cada transductor es capaz de captar un fenómeno físico-químico y convertirlo en una señal eléctrica la cual pasa por un circuito electrónico que se encarga de acondicionar la señal, filtrar el ruido y realizar una conversión análogo-digital para que los datos puedan ser procesados digitalmente, por lo general su fin es comunicarse con algún dispositivo electrónico el cual almacenará e interpretará los datos capturados.

### 2.4.1. Tipos de sensores

En el mercado actual existen diversos tipos de sensores que pueden ser usados dependiendo de la propiedad física que se desee medir o del evento que se desee controlar. La tabla 1 relaciona los principales tipos de parámetros a medir con el tipo de sensor que se puede usar.

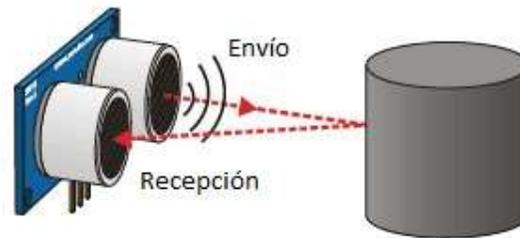
PARÁMETROS	TIPO DE SENSOR
Posicionamiento	GPS
Presión	Manómetros, barómetros
Distancia	Ultrasónico
Acústico	Resonadores, Micrófonos
Movimiento	Giroscopios Acelerómetros

Tabla 1: Tipos de sensores de acuerdo al fenómeno a medir.

### 2.4.2. Sensor ultrasónico HC-SR04

Este sensor hace uso de ondas ultrasónicas para poder medir distancias, el cabezal del sensor emite una onda y mide el tiempo en que la misma tarda en regresar al punto de origen.

A este fenómeno se lo conoce como reflexión de una onda, el mismo que cambia la dirección de la propagación al encontrarse con un objeto. Esta onda reflejada viaja con una velocidad de propagación de igual magnitud, pero en dirección opuesta (ver Figura 2.5).

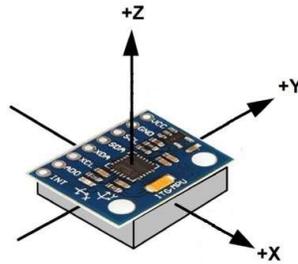


**Figura 2.5: Sensor ultrasónico. [15]**

Para mayor detalle técnico, revisar Anexos.

### 2.4.3. Sensor acelerómetro MMA 7361

Los acelerómetros tienen la capacidad de medir la aceleración en los ejes X, Y, Z. Todo cuerpo sin importar su masa posee un lugar donde se equilibran todas sus fuerzas, a el mismo se le denomina centro de gravedad. Cuando a un objeto se le aplica una fuerza externa, esta se la puede descomponer en cada uno de sus ejes con respecto a su centro de masa. La variación de estas fuerzas con respecto a un marco referencial son las que mostrará el acelerómetro de forma nativa. (Figura 2.6)



**Figura 2.6: Sensor acelerómetro. [16]**

Para mayor detalle técnico, revisar Anexos.

#### **2.4.4. Sistema de posicionamiento global**

El sistema de posicionamiento global (GPS), determina en toda la tierra la posición en la cual se encuentra un objeto, con una precisión de centímetros si se hace uso de un GPS diferencial. Si hacemos uso de un GPS tradicional obtenemos rangos de error de unos cuantos metros [17].

Esta tecnología fue desarrollada, implementada y empleada por el departamento de defensa de Estados Unidos. Con la ayuda de 24 satélites ubicados a cierta altura estos pueden determinar la posición mediante una triangulación de la señal, es decir que al menos tres satélites son los que encierran la señal del receptor.

Gracias a los sensores de gran precisión que se encuentran ubicados internamente en los satélites se puede calcular de manera muy exacta la posición de los receptores o instrumentos que se encuentran transmitiendo señales que esperan ser ubicadas con sus respectivas coordenadas, este tipo de sistemas de posicionamiento global a menudo son usados en vehículos, celulares, botes y demás objetos tecnológicos. En la figura 2.7 se encuentra el GPS usado en el prototipo.



**Figura 2.7: Módulo Adafruit Ultimate GPS. [18]**

## **2.5. Comunicación**

### **2.5.1. LAN inalámbrica o WLAN**

Una red Wireless Local Area Network (WLAN) es un tipo de red que proporciona una comunicación inalámbrica, este tipo de redes se rige bajo el estándar IEEE 802.11.

El intercambio de información se lo hace mediante ondas electromagnéticas usando como único medio el aire, recurriendo a modulaciones y señales portadoras de onda.

Este tipo de tecnología posibilita a los usuarios una amplia movilidad ya que no depende de cables o alguna forma de conexión física para permanecer conectado a la red, uno de los usos más comunes por la sociedad es en la conexión vía WIFI, el cual dispone a los dispositivos de una conexión de internet inalámbrica [19].

Una de sus principales debilidades la encontramos en su seguridad de acceso ya que al no haber un medio físico guiado para establecer conexión se puede acceder a la red desde cualquier punto dentro del rango de cobertura contando con las credenciales correctas, dando paso a que personas malintencionadas puedan interceptar los paquetes enviados a través de la red para posteriormente analizar los requerimientos de los usuarios, es por esto que se recomienda cifrar

el contenido de los datos enviados con dispositivos de acceso mucho más robustos. Sin embargo, WLAN posee destacadas ventajas al ofrecer conexión inalámbrica desde cualquier punto dentro del rango de cobertura, con lo cual permite crear sistemas que no necesitan de cables.

### **2.5.2. ZigBee basado en IEEE 802.15.**

ZigBee es un estándar que se define en un conjunto de protocolos para el levantamiento de varias redes inalámbricas de corto alcance y de baja transferencia de datos, opera en diferentes bandas entre ellas las 2,4GHz y 915MHz.

Fue desarrollado por Alianza ZigBee [20] quienes trabajan con protocolos adaptados a IEEE 802.15. Una de sus principales características es que se desarrollan especialmente para ser usados en redes que necesiten de conectividad inalámbrica para la obtención de datos brindados por sensores, encontramos uno de sus principales usos en el área de la domótica la cual se centra en la automatización de edificaciones y viviendas de cualquier tipo, gestionando energía, seguridad y acceso.

El estándar IEEE 802.15.4 define la capa física y de control de acceso al sistema de redes inalámbricas de área personal con bajas tasas de transmisión de información (low-rate wireless personal area network, LR-WPAN), la actual revisión del estándar se aprobó en 2006.

Este estándar es el sustento donde se da soporte a lo que se define como redes ZigBee, cuyo propósito es brindar una solución íntegra en redes que requieren comunicaciones seguras con baja tasa de envío de datos y eficiencia en el consumo energético.

El bajo consumo energético se obtiene gracias a que el sistema zigbee permanece dormido la mayor parte del tiempo en el cual no se encuentra transmitiendo datos, además transmite en tasas de hasta 250 kbits/s una de las tasas más bajas en comparación con tecnologías existentes.

### **MODULO XBEE PRO S2 zigbee**

El módulo XBEE es una solución integrada que brinda un medio inalámbrico para la comunicación entre dispositivos, este módulo hace uso del estándar antes mencionado ZigBee [20] con el fin de poder crear redes personales ya sean punto-punto o punto-multipunto se caracterizan por ser de tamaño reducido (2.438cm x 2.761cm) y por ser utilizados en aplicaciones que requieren alto tráfico de datos en sensores y baja latencia. Mayormente son usados para crear enlaces de transmisiones inalámbricas dando un rango de cobertura desde 40 m (indoor) hasta 120 m (outdoor). Además, cuentan con un conector tipo rosca el cual permite adaptar diferentes tipos de antena. (ver figura 2.8).



**Figura 2.8: Módulo XBEE PRO RPSMA Conector [21].**

### **2.6. Base de datos**

En general los sistemas informáticos utilizan una base de datos para poder gestionar su información, debido a que cada día se genera gran cantidad de información es necesario que esta permanezca almacenada de una manera organizada.

El termino base de datos surgió en 1963, en la informática una base de datos consiste en una colección de datos interrelacionados y un conjunto de programas para acceder a dichos datos, una colección de datos donde los datos están lógicamente relacionados entre sí.

## **MYSQL**

MySQL es un sistema que ayuda a gestionar bases de datos relacionales para su desarrollo, considerada una base de datos open source, entre las más famosas del medio informático. Es una plataforma muy rápida en resultados de respuesta de lectura ya que utiliza un motor no transaccional, pero puede ocasionar problemas en la integridad del entorno de alta concurrencia al momento de modificar miles de datos, sin embargo, en aplicaciones que involucran altas tasas de lectura de datos MySQL es la opción más ideal.

### **2.7. Aplicación web**

Se denomina aplicación web a todas las herramientas que los usuarios pueden utilizar accediendo a un servicio o servidor web a través de internet o por medio de una intranet. Las aplicaciones web son muy utilizadas debido a su práctico uso ya que el cliente utiliza un navegador para acceder a la aplicación independientemente del sistema operativo usado o si accede por medio de computadora, celular, Tablet o algún otro dispositivo.

#### **Framework web node.js express**

Este framework permite hacer uso de una extensa infraestructura para crear aplicaciones web bajo Node.js al mismo tiempo proporciona un conjunto de características para la creación de aplicaciones web y móviles. Permite renderizar de manera rápida archivos HTML, además hace uso de re-direccionamiento dinámico gracias a los métodos HTTP y URL que se encuentran implementados.

## CAPÍTULO 3

### 3. DESARROLLO.

El desarrollo del prototipo está basado en el concepto de Smart Cities, así como en el uso de telemetría la cual permite realizar mediciones de manera remota. Los datos usados para detectar choques fueron adquiridos por los propios sensores y posteriormente llevados a un análisis gráfico con el fin de determinar el valor umbral que toma cada variable durante una colisión.

Una vez que se comprendió como cambian las variables arrojadas por los sensores al momento de ser expuestos a un choque se desarrolló un protocolo que permita intercambiar paquetes de datos entre los nodos del prototipo, pudiendo de esta manera crear una red que alerte al sistema cuando ocurra un siniestro vehicular.

Posteriormente todos estos datos tienen que ser transformados en información útil a los usuarios del sistema, por tal motivo se creó una plataforma web en la cual se puede alertar sobre un accidente nuevo y además consultar a manera de reportes choques suscitados en el pasado.

Finalmente se expondrán los ajustes necesarios en el diseño del mismo para que el prototipo pueda ser implementado en una carretera real, se analizarán y recomendarán los tipos de elementos que se deben sustituir o potenciar para que la red funcione de manera óptima.

#### 3.1. Descripción del escenario

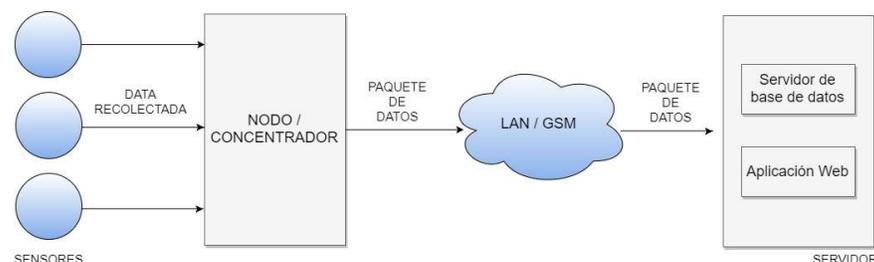
El diseño de la red de sensores inalámbrica será realizado tomando en cuenta parámetros y condiciones ambientales ideales a lo largo del tramo coliseo - facultad EDCOM del campus Gustavo Galindo de la Escuela Superior Politécnica del Litoral (ESPOL) en el cual se considerará la poca densidad poblacional, la que de existir podría incurrir en la pérdida de potencia de las señales transmitidas por los nodos hasta llegar al concentrador. En este tramo de carretera se

realizaron las pruebas de cobertura de nuestros nodos, así como pruebas de latencia.

El escenario específico para las pruebas corresponde a un área sin cobertura GSM. Normalmente este tipo de escenarios se los encuentra en carreteras que conectan diferentes ciudades debido a que existe un incorrecto despliegue de antenas, una alta densidad poblacional o de vegetación, también influyen temas físicos como condiciones climáticas o zonas de mucha interferencia debido a valles, colinas, árboles, etc.

### 3.2. Arquitectura general de la red inalámbrica de sensores

La red de sensores se basa en la arquitectura de red ad-hoc en conjunto con el modelo cliente-servidor. Los componentes principales de la arquitectura son: los sensores, los nodos y la red por la cual se comunica el servidor de base de datos con cada nodo. Esta red puede ser una GSM o WLAN. Además, encontramos al servidor de base de datos que en conjunto con la aplicación web permite gestionar la información recolectada por los sensores. (Figura 3.1)



**Figura 3.1: Arquitectura general de la red de sensores inalámbrica.**

### 3.3. Diseño del dispositivo sensor

Para poder entender mejor el diseño del sensor, podemos dividir el mismo en tres diferentes tipos: sensor de aceleración, detección de proximidad y GPS.

De estos tres se ha establecido como nivel de prioridad alto al sensor de la aceleración ya que este brindará información de los cambios de velocidad que experimenta el vehículo. El sensor acelerómetro obtiene valores adimensionales correspondientes a los ejes de coordenadas X, Y, Z mientras que el sensor de distancia establece que tan cerca se encuentra el vehículo de un obstáculo.

El acelerómetro es un sensor muy sensible por lo que la aceleración en cada uno de sus ejes oscila constantemente entre un rango de valores, debido a aquello se realizaron pruebas en campo en las cuales se recolectaron los datos arrojados por el sensor cuando se recreaban frenados por colisión, al realizar una gráfica de los datos se notó que al momento de un siniestro los sensores de aceleración alcanzaban valores pico, mientras que al censar velocidades con muy poco cambio en su magnitud el valor de los mismos oscilaba parcialmente, en cuanto al efecto del frenado continuo este valor de oscilación se atenúa o se reduce con una variación parsimoniosa.

Una vez evaluado el sensor de acelerómetro se buscó una condición para el sensor de proximidad, esto se lo hizo conforme a las dimensiones frontales del vehículo, si se censa un obstáculo con una distancia poco prudencial desde el sensor hasta la punta delantera del vehículo entonces se considerará la probabilidad de una colisión.

La evaluación de los valores arrojados por estos dos sensores permitirá modelar un algoritmo que permita detectar el siniestro automovilístico.

De igual forma es necesario obtener los valores arrojados por el GPS con lo que se debe separar los datos de latitud y longitud que arroja el sensor, con el fin de poder transmitir los mismos de manera más estructurada al concentrador.

Al momento de transmitir la trama de datos esta pasa por un proceso de encapsulación de modo que cada trama comienza y termina con paréntesis y a su vez cada dato perteneciente a la trama se lo separa con un pipe según la ecuación 3.1:

$$\text{Trama} = (\text{bandera} | \text{placa} | \text{coordenadas}) \quad (3.1)$$

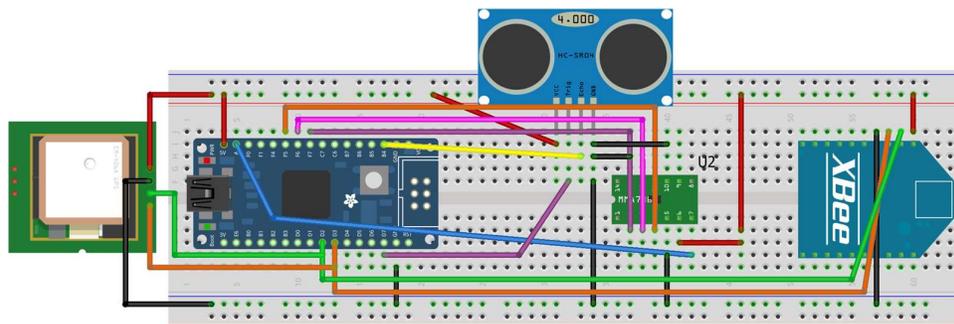
Finalmente, una vez creada la trama esta puede ser enviada mediante la red desde el concentrador al servidor.

La tabla 2 nos indica la forma en la cual deben ser conectados los PINES del transmisor y de los sensores con respecto al Arduino.

ID Arduino nano	HC-SR04	MMA 7361	XBEE PRO S2	GPS
VCC	VCC (1)	5V	VCC(1)	VCC(1)
GND	GND (4)	GND G select self-test	GND(4)	GND(2)
RX			TX(3)	TX(3)
TX			RX(2)	RX(4)
A0		X		
A1		Y		
A2		Z		
D7	Trig(2)			
D8	Echo(3)			
AREF		SLEEP		

**Tabla 2: Asignación de PINES del sensor**

Una vez realizada la conexión de cada PIN con su correspondiente entrada / salida obtuvimos el dispositivo sensor que será instalado dentro del vehículo, la figura 3.2 nos da una vista general del diagrama de conexiones del sensor.

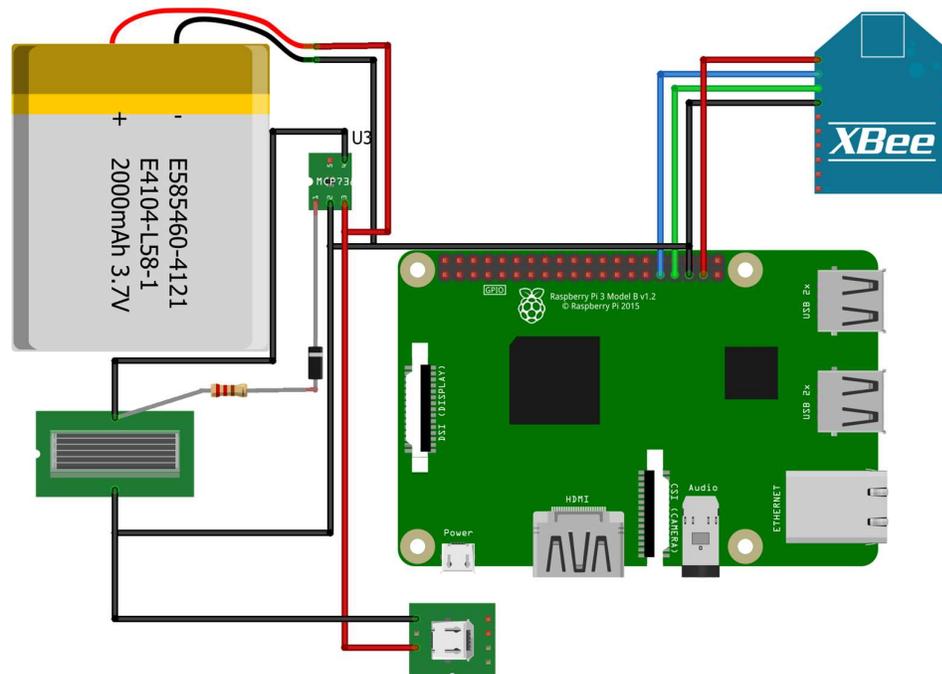


**Figura 3.2: Diagrama de conexiones del sensor.**

### 3.4. Diseño del concentrador

La principal tarea del concentrador es la de gestionar la conexión con los sensores, recibir la trama de datos enviada, procesarla e inyectar información a la base de datos. Para el diseño del mismo se optó por dos módulos; un módulo GSM y un módulo de comunicación XBEE. El módulo GSM nos permite tener acceso a la nube mediante una infraestructura de telecomunicaciones existente mientras que el módulo XBEE nos permite acceder a la red de sensores Zigbee.

La integración de estos dos módulos permite que el sistema embebido actúe como un concentrador de la red, adicionalmente el uso de una batería externa le da autonomía al dispositivo, en la figura 3.3 se muestra el diagrama de conexión del concentrador.



**Figura 3.3: Diagrama de conexión del concentrador**

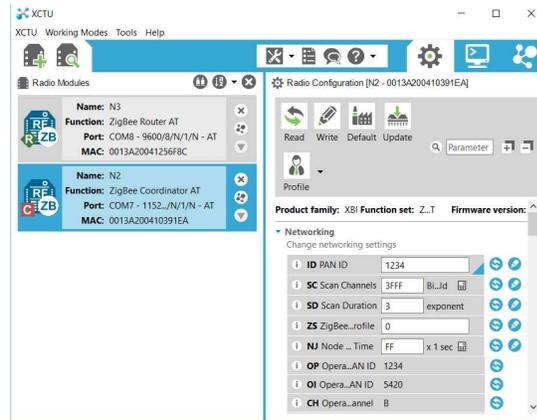
### 3.5. Diseño de red de comunicación estructurada

#### 3.5.1. Comunicación sensor-concentrador

Para poder establecer la comunicación entre las antenas se hizo uso de la herramienta del distribuidor XCTU, con el fin de levantar la red Zigbee entre el dispositivo sensor y el concentrador.

Para esta conexión se consideró manejar comunicación AT debido a que esta es una comunicación transparente, es decir no utiliza una estructura de mensaje dentro de una trama, por lo tanto, se implementó un encapsulado para mejorar el rendimiento al momento de que el mensaje sea interpretado por el concentrador.

La configuración de las antenas debe ser de tal modo que una de ellas se encuentre en modo ruteador y la otra en modo coordinador que es quien levanta la red de comunicación entre las antenas que presenten el mismo valor de PAN ID.(ver figura 3.4)



**Figura 3.4: Configuración de las antenas en XCTU**

Para poder recibir el mensaje del dispositivo sensorial de los vehículos este debe estar en el rango de cobertura del concentrador, con el fin de establecer una red diferenciada y así garantizar la recepción de los datos transmitidos por los sensores hacia el concentrador.

### 3.5.2. Comunicación concentrador-servidor

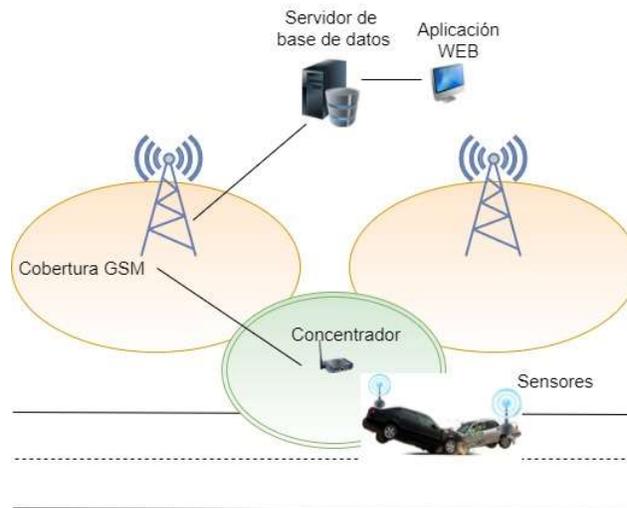
La comunicación entre estos dos dispositivos se la realizó de forma serial, una vez que el mensaje es enviado por el sensor mediante la red Zigbee hacia el concentrador este se aloja en el buffer de entrada de lectura del puerto serial para inmediatamente ser analizado por el concentrador, este procesamiento de datos se lo realiza por medio de un script escrito en Python el cual por medio de un algoritmo basado en estados logra la lectura del mensaje y la segmentación de los datos pertenecientes a la trama los cuales son procesados como texto. Dentro del mismo script se muestra la función en la que se establece conexión segura con la base de datos para aquello tanto el servidor como el concentrador deben estar dentro de la misma red para así poder inyectar datos a la base, esto con el fin de retransmitir y almacenar la información de forma persistente en un modelo relacional.

Para una mayor comprensión del script, revisar el anexo 1.

### 3.6. Diseño del Sistema de información

El sistema de información está dividido en dos partes, una involucra el hardware mientras que la otra involucra el software desarrollado.

La figura 3.5 describe de una manera visual las dos partes principales del sistema de información. La primera consiste de los sensores y del concentrador mientras que la segunda consta del servidor de bases de datos y la aplicación web.



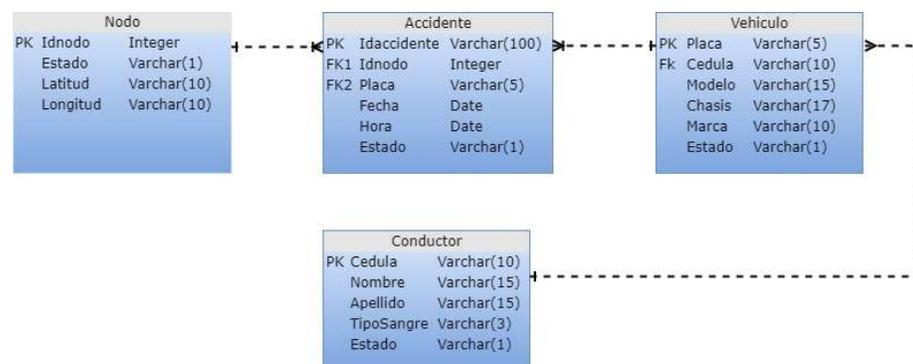
**Figura 3.5: Partes involucradas en el flujo de información**

La integración de hardware y software fue lo que permitió realizar el prototipo, el cual funciona de la siguiente manera. Se puede visualizar dos antenas con cobertura GSM radial, el choque automovilístico dentro del escenario ocurre en el área entre las dos antenas donde la cobertura GSM es nula, aquí los datos arrojados por los sensores son transmitidos hacia el concentrador para posteriormente ser evaluados. Una vez que el concentrador mediante su algoritmo detecta un siniestro, el mismo retransmite los paquetes de datos a los diferentes concentradores de la red hasta llegar a establecer conexión con la red GSM más cercana. Finalmente, dicho paquete es enviado mediante la red GSM

al servidor de base de datos donde con la ayuda de una aplicación web se podrán visualizar los datos.

### 3.6.1. Modelo entidad relación de la base de datos

Debido a que se manejan grandes cantidades de datos recopilados por los sensores, los mismos deben ser modelados y agrupados de modo que sean de fácil y rápido acceso, debido a esto se decidió implementar una estructura de datos la cual se divide en cuatro tablas relacionales que representan entidades del mundo real con sus respectivos atributos, a continuación, la figura 3.6 describe el modelo entidad relación de la base de datos.



**Figura 3.6: Modelo entidad relación de la base de datos.**

El diseño de la base de datos consta de las principales entidades que conforman el proyecto: Nodo, Accidente, Conductor, Vehículo. La tabla "Nodo" almacenará todos los datos referentes al concentrador, la tabla 3 indica los tipos de datos de los campos y la descripción de cada uno de ellos.

<b>Campo</b>	<b>Tipo de dato</b>	<b>Descripción</b>
<b>Idnodo (PK)</b>	Integer	Llave primaria de la tabla nodo, tendrá un valor único
<b>Estado</b>	Varchar	Almacena el estado del nodo, este puede ser Activo (A) o Inactivo (I)
<b>Latitud</b>	Varchar	Almacena la latitud en la cual se encuentra el nodo
<b>Longitud</b>	Varchar	Almacena la longitud en la cual se encuentra el nodo

**Tabla 3: Diccionario de datos de la tabla nodo.**

La tabla "Conductor" será poblada con los datos del usuario del prototipo. La tabla 4 indica los tipos de datos de los campos y la descripción de cada uno de ellos.

<b>Campo</b>	<b>Tipo de dato</b>	<b>Descripción</b>
<b>Cedula</b>	Varchar	Llave primaria de la tabla Conductor, tendrá un valor único
<b>Nombre</b>	Varchar	Almacena el nombre del usuario
<b>Apellido</b>	Varchar	Almacena el apellido del usuario
<b>TipoSangre</b>	Varchar	Almacena el tipo de sangre del usuario
<b>Estado</b>	Varchar	Almacena el estado de ese usuario, este puede ser Activo (A) o Inactivo (I)

**Tabla 4: Diccionario de datos de la tabla conductor.**

La tabla "Vehículo" permitirá relacionar un conductor con un vehículo. La tabla 5 indica los tipos de datos de los campos y la descripción de cada uno de ellos.

<b>Campo</b>	<b>Tipo de dato</b>	<b>Descripción</b>
<b>Placa (PK)</b>	Varchar	Llave primaria de la tabla vehículo, tendrá un valor único
<b>Cedula (FK)</b>	Varchar	Llave foránea de la tabla vehículo que hace referencia a la tabla conductor, permitiendo la asociación de un conductor con un vehículo
<b>Modelo</b>	Varchar	Almacena el modelo del vehículo
<b>Chasis</b>	Varchar	Almacena el chasis del vehículo
<b>Marca</b>	Varchar	Almacena la marca del vehículo
<b>Estado</b>	Varchar	Almacena el estado del vehículo, este puede ser Activo (A) o Inactivo (I)

**Tabla 5: Diccionario de datos de la tabla vehículo.**

La tabla "Accidente" permitirá relacionar el nodo que registra el accidente con el vehículo afectado, debido a que el vehículo está relacionado fuertemente con un conductor podemos obtener mediante relaciones los datos del mismo. La tabla 6 indica los tipos de datos de los campos y la descripción de cada uno de ellos.

<b>Campo</b>	<b>Tipo de dato</b>	<b>Descripción</b>
<b>Idaccidente (PK)</b>	Varchar	Llave primaria de la tabla accidente, tendrá un valor único
<b>Idnodo (FK)</b>	Integer	Llave foránea de la tabla accidente que hace referencia a la tabla nodo, permitiendo la asociación de un accidente con un nodo
<b>Placa (FK)</b>	Varchar	Llave foránea de la tabla accidente que hace referencia a la tabla vehículo, permitiendo la asociación de un accidente con un vehículo
<b>Fecha</b>	Date	Almacena la fecha del accidente
<b>Hora</b>	Varchar	Almacena la hora del accidente
<b>Estado</b>	Varchar	Almacena el estado del accidente, este puede ser Activo (A) o Inactivo (I)

**Tabla 6: Diccionario de datos de la tabla accidente.**

Finalmente, toda la información almacenada es presentada en una página web a modo de reportería, dicha página consta de 4 secciones más un pop up que permite informar en tiempo real el siniestro automovilístico.

### 3.6.2. Aplicación web

La página web fue desarrollada de manera local, usando computadoras personales, para acceder a la página se habilitó el puerto 9000 por medio de la dirección IP local. Esta página web contiene secciones importantes que se detallan a continuación:

#### Menú principal

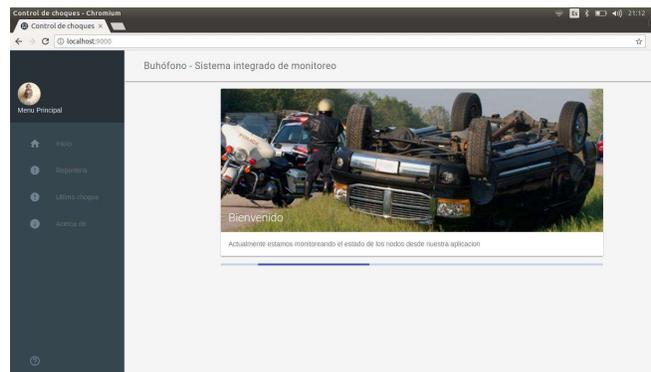
El menú principal de la aplicación consta de una barra lateral, en ella encontramos todas las ventanas de nuestro sitio web por las cuales podemos navegar, la figura 3.7 detalla las partes principales de la página.



**Figura 3.7: Menú principal 1) Inicio 2) Reportería 3) Último choque 4) Acerca de.**

### **Pantalla de Inicio.**

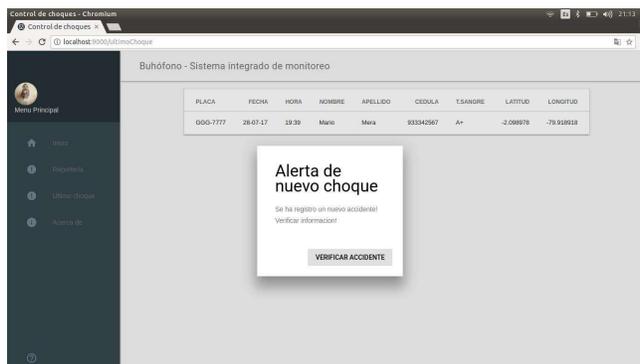
Al momento de ingresar a la aplicación web se muestra la pantalla de bienvenida donde podemos observar el estado del monitoreo actual, mostrando el mensaje "Actualmente estamos monitoreando el estado de los nodos desde nuestra aplicación". La figura 3.8 muestra la pantalla inicial.



**Figura 3.8: Ventana de pantalla inicial.**

### **Notificación en tiempo real**

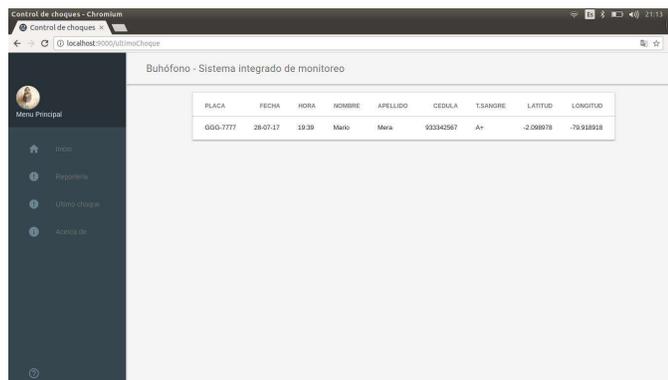
Este pop-up o elemento emergente aparece en tiempo real al suscitarse un choque automovilístico, en ella se puede apreciar un mensaje indicando: "Alerta de nuevo choque" seguido de un botón en su parte inferior derecha indicando: "Verificar accidente", este botón al ser presionado direccionará automáticamente a la ventana de último accidente registrado, donde podremos verificar el registro correspondiente al último accidente suscitado. La figura 3.9 muestra la notificación en tiempo real.



**Figura 3.9: Pop-up de notificación tiempo real.**

### Ultimo accidente registrado

Esta ventana es de gran utilidad para la entidad de socorro ya que permite conocer de primera mano toda la información acerca del último choque automovilístico, es posible acceder a la página mediante el menú principal o inmediatamente después de que ocurra un siniestro vehicular. La figura 3.10 muestra el último accidente registrado.



**Figura 3.10: Ventana de último accidente registrado.**

## Reportería

En esta ventana podemos encontrar la reportería general del sistema, aquí se podrá consultar la información del historial de choques registrados por la red de sensores, es muy similar a la ventana de "último accidente registrado" con la diferencia de que presentan todos los registros provenientes de la base de datos. La figura 3.11 muestra la reportería de la aplicación.

PLACA	FECHA	HORA	NOMBRE	APELLIDO	CEDULA	T.SANGRE	LATITUD	LONGITUD
GGG-7777	Fri Aug 18	16:17:25	Mario	Mera	93342567	A+	-2.088978	-79.918918
GGG-7777	Fri Aug 18	16:19:36	Mario	Mera	93342567	A+	-2.088978	-79.918918
GGG-7777	Fri Aug 18	16:19:50	Mario	Mera	93342567	A+	-2.088978	-79.918918
GGG-7777	Fri Aug 18	16:19:59	Mario	Mera	93342567	A+	-2.088978	-79.918918
GGG-7777	Fri Aug 18	16:20:47	Mario	Mera	93342567	A+	-2.088978	-79.918918
GGG-7777	Fri Aug 18	16:20:49	Mario	Mera	93342567	A+	-2.088978	-79.918918
GGG-7777	Fri Aug 18	16:20:55	Mario	Mera	93342567	A+	-2.088978	-79.918918
GGG-7777	Fri Aug 18	16:23:26	Mario	Mera	93342567	A+	-2.088978	-79.918918
GGG-7777	Fri Aug 18	16:25:29	Mario	Mera	93342567	A+	-2.088978	-79.918918
GGG-7777	Fri Aug 18	16:25:34	Mario	Mera	93342567	A+	-2.088978	-79.918918
GGG-7777	Fri Aug 18	16:26:05	Mario	Mera	93342567	A+	-2.088978	-79.918918

**Figura 3.11: Ventana de Reportería.**

### 3.7. Arquitectura de la red inalámbrica robusta

Para que el sistema sea desplegado en un entorno real debemos de indicar cómo debería de adecuarse nuestro prototipo cuando exista o no cobertura GSM, además debemos de definir antenas y ruteadores que se ajusten a las necesidades, sugerir cuanta distancia debe de existir entre cada nodo, entre otras cosas.

Considerando las variables de diferenciación de una red GSM usualmente existe perdida de cobertura debido a la densidad de usuarios o vegetación, una red que presenta este tipo de pérdidas genera decadencia en el servicio GSM para la cual se debe considerar un despliegue con antenas de mayor alcance. Para poder suplir las necesidades de este escenario se puede hacer uso de las Yagi antenas ya que tienen la capacidad de focalizar en secciones el alcance de su cobertura mostrando en los mejores casos un alcance de cobertura de 30km+ por lo que

situándolas de manera adecuada se ajustan a aliviar nuestro problema, adicional se debe colocar un ruteador con comunicación GSM en un área donde si exista cobertura.

Luego de considerar el tipo de antena es necesario proporcionar un mejoramiento del prototipo con los dispositivos. La tabla 7 nos muestra los cambios necesarios que debe tomar el prototipo para poder cumplir con el objetivo.

Componente	Prototipo	Macro
<b>Sensor</b>	Arduino , sensores,XBEE	Sin cambios
<b>Concentrador</b>	Raspberry pi 3 B Modulo XBEE S2 Batería 5V	MikroTik RB912UAG-2HPnD AirMax sector AM-2G15-120 3g miniPCle card Phocos CMLmppt 10A Yingli solar Serie YGE 72 Cell BOSCH S6 High Performance
<b>Servidor</b>	Cpu core i5, Linux SO, 6Gb Ram,	Lenovo - ThinkServer Tower Server - 1 x Intel Xeon E3-1235 3.20 GHz

**Tabla 7: Mejoras del prototipo para una red robusta.**

Uno de los propósitos es el de aprovechar todas las ventajas del prototipo e integrarlas a una estructura de comunicación que se encuentran normalmente en carreteras logrando cubrir la mayoría de lugares de difícil cobertura, se detalla a continuación la robustez de los componentes a usar.

MikroTik RB912UAG-2HPnD este dispositivo funciona como un intérprete en remplazo de la Raspberry ya que puede manejar bandas duales con una memoria RAM de 64MB y un almacenamiento de 128MB además de eso maneja un chip

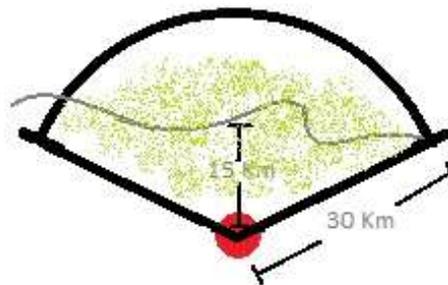
Wireless 2.4 GHz AR9342 estándar 802.11b/g/n con transmisión MCS7 de 24 dBm, -73 sensibilidad de recepción a 251 mW poder de Transmisión [22].

AirMax sector AM-2G15-120 esta antena tiene la apertura de un ángulo de 120 grados de sector de cobertura, con buena resistividad para exteriores maneja frecuencia de 2.4GHz con una ganancia de 15dBi, cuenta con una interfaz para sim card con soporte 3G con la ayuda de 3g miniPCle card para el soporte GSM [23].

Phocos CMLmppt 10A dispone un control de temperatura para manejar los algoritmos de carga en relación al máximo aprovechamiento del panel solar, conectado y administrado por software, se complementa con la versatilidad en 12V y 24V con su sistema automático de detección de voltaje, con ayuda de otros elementos presenta la comunicación por LEDs para mostrar la carga de la batería y notificar cuando esta esté con poco voltaje. [24]

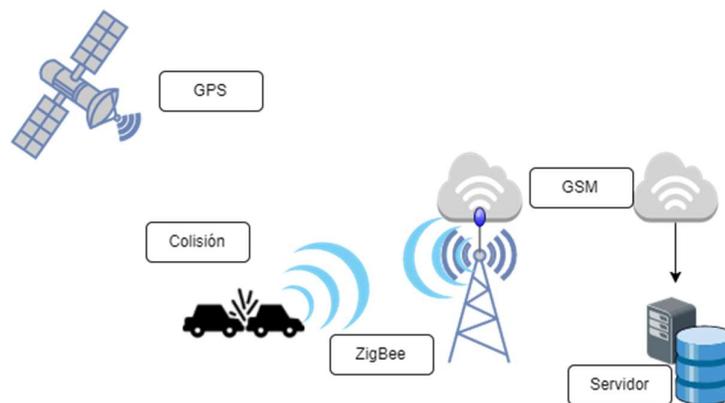
Yingli solar Serie YGE 72 Cell es un panel solar de alta tecnología siendo el más robusto y el de mayor dimensión contiene cristal de alta transmisión y cuenta con una capa anti reflejante única que permite dirigir más luz hacia las celdas solares, con una máxima entrega de 15A y soporta una temperatura de 85 grados Celsius. [25]

Teniendo en consideración las características de los componentes que son brindadas por el fabricante, hemos definido que para mantener una cobertura eficaz de la red se debe de ubicar un concentrador cada 50 km, cada uno brinda una cobertura radial a un ángulo de 120 grados ubicándolos a 30km de los puntos lejanos. La distancia ideal en el punto medio de la zona de cobertura es de 15km para garantizar la misma. La figura 3.12 muestra el radio de cobertura y la distancia ideal en la cual se deben ubicar los nodos.



**Figura 3.12: Amplitud de la zona de cobertura.**

Cada punto de las antenas representa la arquitectura de la red Zigbee con sus concentradores, es decir que en cada estación existirá un concentrador que se conecta por medio de GSM al servidor. La figura 3.13 muestra detalladamente los componentes de la red robusta y su forma de comunicación.



**Figura 3.13: Diagrama de la red robusta.**

## CAPÍTULO 4

### 4. ANÁLISIS DE RESULTADOS

Esta sección describirá los resultados de las pruebas realizadas en el prototipo descrito en la sección 3. Estos resultados demuestran la capacidad del mismo para detectar choques de manera precisa y el análisis de cobertura de sus antenas.

#### 4.1. Visión general de la plataforma de experimentos

Todos los experimentos de configuración, montaje y detección de choques del prototipo fueron realizados en el Centro de Investigación, Desarrollo e Innovación de Sistemas Computacionales (CIDIS) ubicado dentro del campus Prosperina de la Escuela Superior Politécnica del Litoral (ESPOL), usando como herramientas microcontrolador arduino uno equipados con sensores de distancia (HC-SR04), y acelerómetro (MMA7361), sistema embebido Raspberry pi model B con sistema operativo Raspbian. Laptop Samsung con core i5 corriendo Linux Ubuntu con un sistema gestor de base de datos MySQL y servidor HTTP con Node.js.

Los datos recolectados por los sensores fueron descargados y analizados en el aplicativo Excel, el análisis de la conectividad según la distancia se realizó de forma empírica evaluando la recepción de los paquetes de mensajes variando la distancia. Esta última prueba fue realizada entre el coliseo y la facultad EDCOM de la institución antes mencionada. La figura 4.1 muestra momentos previos a las pruebas realizadas.



**Figura 4.1: Momentos previos a la prueba.**

#### **4.2. Resultados empíricos de los sensores**

Para poder detectar choques fue necesario realizar pruebas de forma empírica, en las cuales se arrojaron datos de los sensores para que sean posteriormente evaluados. Dado que los sensores se encuentran instalados en el interior del carro prototipo estos experimentan las mismas fuerzas que se manifiestan dentro del vehículo, así podemos identificar lo que sucedió durante el accidente. Los sensores de acelerómetro miden la aceleración experimentada durante una frenada repentina. El decremento de la aceleración en los tres ejes es un indicador de la fuerza inducida hacia el dispositivo debido al choque a medida que el carro alcanza el reposo repentino.

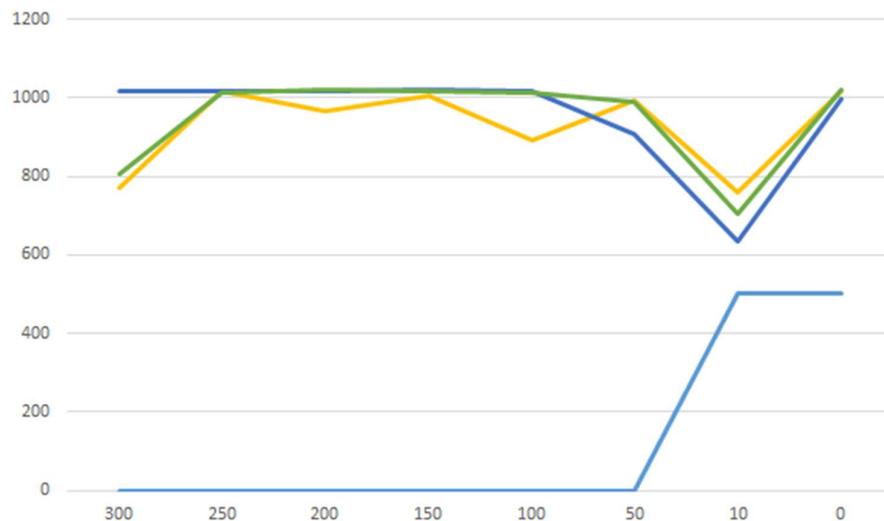
En la tabla 8 observamos los datos recogidos por los sensores mientras el prototipo se encontraba en movimiento. Con el fin de poder graficar de una mejor manera la forma de detectar un choque mediante el sensor de distancia se lo programó de tal manera que arroje una lectura estática al momento de censar una distancia menor a 10 cm.

<b>Distancia [cm]</b>	<b>X</b>	<b>Y</b>	<b>Z</b>	<b>Ultrasónico</b>
<b>300</b>	1019	906	770	0
<b>250</b>	1016	1013	1016	0
<b>200</b>	1019	1020	967	0
<b>150</b>	1020	1018	1007	0
<b>100</b>	1018	1014	894	0
<b>50</b>	907	990	994	0
<b>10</b>	634	703	758	450
<b>0</b>	996	1020	1016	450

**Tabla 8: Datos recogidos por los sensores según la distancia.**

La gráfica de la figura 4.2 muestra el comportamiento de los diferentes ejes que conforman el acelerómetro conforme va avanzando el prototipo hasta colisionar contra otro vehículo. Podemos observar como la gráfica alcanza picos alrededor de los 10 [cm] de distancia, la misma que es el disparador al momento de detectar el evento de choque. Antes de esta medida se observa como los sensores van variando de manera independiente debido a que tanto el prototipo como los sensores instalados se encuentran en movimiento, en este periodo de tiempo también se observan picos mucho más suaves que el que se presenta al momento de un choque.

Las tres graficas superiores representan los ejes de los sensores de acelerómetro x, y, z los cuales son amarillo, verde y azul respectivamente mientras que la gráfica inferior representa al sensor de distancia.



**Figura 4.2: Respuesta de sensores ante una colisión**

Uniando las respuestas de los sensores podemos definir el escenario en el cual ocurre un choque, esto es cuando se censa una distancia menor a 10 cm contra un obstáculo y cuando el acelerómetro cambia bruscamente de valores estando por debajo del número 800.

### 4.3. Análisis de cobertura vs distancia.

Con el objetivo de verificar el rango de cobertura de nuestro prototipo en un entorno real se ha procedido a realizar pruebas de conectividad dentro del escenario indicado con anterioridad, dichas pruebas permitieron medir de manera empírica la cobertura del sistema a medida que se varía la distancia (el prototipo se aleja de la antena receptora).

Para dichas pruebas se elaboró un algoritmo el cual envía tramas de datos de manera secuencial y sin interrupción, una vez que se deje de recibir dichas tramas de datos se entenderá que la comunicación entre el prototipo y el servidor ha dejado de establecerse. En la figura 4.3 se muestra el trayecto recorrido en la prueba, así como los puntos referenciales (íconos verdes) en los cuales se iba

registrando la distancia desde el punto referencial de partida (ícono verde con visto). El punto donde se perdió la conexión está representado por el ícono rojo.



**Figura 4.3: Trayectoria de recorrido de prueba**

En la tabla 9 se muestran las diferentes coordenadas de latitud y longitud del recorrido con su respectiva distancia desde el punto de referencia, además se evalúa si existe o no cobertura en cada uno de esos puntos.

Orden	Coordenadas	Distancia [m]	Cobertura
1	-2.1454365253448486,-79.96392822265625	0	SI
2	-2.1450469493865967,-79.96316528320312	98	SI
3	-2.1446731090545654,-79.96267700195312	165	SI
4	-2.14394474029541,-79.96185302734375	295	SI
5	-2.143794298171997,-79.96109771728516	357	SI
6	-2.1445999145507812,-79.96051025390625	480	NO

**Tabla 9: Prueba de cobertura según distancia**

Al realizar dichas pruebas se pudo notar cierto retraso en el envío de paquetes a partir de la lectura 4, el retraso que se pudo constatar fue en promedio de 3 segundos. Sin embargo, este tiempo de retraso es ínfimo comparado al tiempo de reacción promedio de una persona para poder realizar una llamada de emergencia en caso de un siniestro.

## CONCLUSIONES Y RECOMENDACIONES

Un sistema de detección de choques que utiliza hardware Open Source sería una herramienta muy útil para las instituciones de tránsito y unidades de socorro, a la vez fomentaría el uso de herramientas tecnológicas con el fin de elevar la calidad de vida del país.

La topología de red adecuada en este tipo arquitectura de sensores es la centralizada, la cual involucra múltiples clientes o nodos que se comunican hacia un servidor de base de datos.

La información recolectada por los sensores al momento de un siniestro permite comprender y cuantificar de una mejor manera dichos sucesos. En base a dicha información se puede realizar un análisis de datos para encontrar tendencias o patrones en los accidentes vehiculares, como lo es horas pico donde ocurren accidentes, tipos de vehículos que lo ocasionan, personas reincidentes, etc.

De existir dos vehículos equipados con sensores se registrarán dos choques en las mismas coordenadas geográficas, se puede crear un algoritmo que discrimine uno de los dos choques al azar, sin embargo, se recomienda que se registren ambos ya que eliminar uno de ellos afectaría a la bitácora y no habría coherencia en los datos. Realizar un mantenimiento cada cierto periodo de tiempo a la Base de Datos con el fin de migrar la información obsoleta a tablas históricas.

Estudiar el caso específico en el cual el conductor del vehículo sea alguien ajeno al mismo, estos casos se dan cuando se presta el vehículo a familiares o amigos.

## GLOSARIO DE TÉRMINOS

**EDCOM:** escuela de diseño y comunicación visual

**CISCO:** es una empresa global principalmente dedicada a la fabricación, venta, mantenimiento y consultoría de equipos de telecomunicaciones.

**ARM:** Advanced RISC Machine

**GNU:** es un sistema operativo de tipo Unix desarrollado por y para el Proyecto GNU, y auspiciado por la Free Software Foundation.

**GPIO:** (General Purpose Input/Output), Entrada/Salida de Propósito General.

**IEEE:** Instituto de Ingeniería Eléctrica y Electrónica

**GSM:** El sistema global para las comunicaciones móviles

**GPS:** sistema de posicionamiento global, permite detectar un objeto con gran precisión.

## BIBLIOGRAFÍA

- [1] El Tiempo (2014, noviembre). Tránsito en la región cumple una norma: sálvese quien pueda Diario [Noticiero] Disponible en: <http://www.eltiempo.com/archivo/documento/CMS-14777583>
- [2] Andes (Agencia Publica de Noticias del Ecuador y Sudamérica), (2014, julio 23) Ecuador segundo país con la tasa de siniestralidad vial más preocupante de América Latina Diario [Noticiero] Disponible en: <http://www.andes.info.ec/es/noticias/ecuador-segundo-pais-tasa-siniestralidad-vial-mas-preocupante-america-latina.html>
- [3] ANT. Estadísticas de transporte terrestre y seguridad vial, disponible en: <http://www.ant.gob.ec/index.php/noticias/estadisticas#.Wan2vMgjHIV>
- [4] I.A. Cardiel, lenguaje de sistema genérico, multiparadigmático: España: UNED, 2016
- [5] 20minutos (2016, agosto) la llamada de emergencia eCall un salvavidas en carretera [Noticiero] disponible en: <http://www.20minutos.es/noticia/2791739/0/llamada-emergencia-automatica-ecall-salvavidas-carretera/>
- [6] DIGIBYTES (2015, abril) ecall in all new cars from April 2018 [News], disponible en: <https://ec.europa.eu/digital-single-market/en/news/ecall-all-new-cars-april-2018>
- [7] E. Sosa, "Contribuciones al establecimiento de una red global de sensores inalámbricos interconectados", tesis doctoral UNLP, febrero 2011
- [8] IBSG CISCO, abril del 2011.
- [9] D. Evans, abril del 2011, CISCO Internet of Things,
- [10] Arduino (2017, agosto 1) Products [informativo] disponible en: <https://www.arduino.cc/en/Main/Products>
- [11] Arduino (2015, enero), "Arduino uno" [Online]. Disponible en: <https://store.arduino.cc/usa/arduino-uno-rev3>

- [12] Arduino (2015), "Arduino Picaro+" [Online]. Disponible en: <http://arduino.cl/picaro/>
- [13] Arduino (2015, enero), "Arduino Mega 2560" [Online]. Disponible en: <https://store.arduino.cc/usa/arduino-mega-2560-rev3>
- [14] Raspberry (2017, agosto) Products [informativo] disponible en: <https://www.raspberrypi.org/products/>
- [15] Arduino (2014, noviembre) Forum [colaborativo] disponible en: <https://forum.arduino.cc/index.php?topic=277722.0>
- [16] Naylamp (2016, julio) Forum [colaborativo] disponible en: [http://www.naylampmechatronics.com/blog/45\\_Tutorial-MPU6050-Aceler%C3%B3metro-y-Giroscopio.html](http://www.naylampmechatronics.com/blog/45_Tutorial-MPU6050-Aceler%C3%B3metro-y-Giroscopio.html)
- [17] H. Castillo (2013, nov.) Medicion con GPS. disponible en: <https://es.slideshare.net/HugoCastillo4/gps-diferencial>
- [18] L. Ada, Adafruit Industries (2015, febrero) [En línea]. Disponible en: <https://learn.adafruit.com/downloads/pdf/adafruit-ultimate-gps.pdf>.
- [19] M. Rouse (2012, nov) LAN inalámbrica, WLAN [informativo] disponible en: <http://searchdatacenter.techtarget.com/es/definicion/LAN-inalambrica-WLAN-o-Wireless-Local-Area-Network>
- [20] ZigBee Specification (2008, enero) disponible en Zigbee Alliance, [www.zigbee.org](http://www.zigbee.org)
- [20] S. Farahani, "ZigBee Wireless Networks and transceivers", Elsevier ,2008
- [21] Sparkfun (2017, junio) XBee pro RPSMA disponible en: <https://cdn.sparkfun.com//assets/parts/2/4/9/1/09099-03-L.jpg>
- [22] Mikrotik (2017, agosto) productos [Online] disponible en: <https://mikrotik.com/product/RB912UAG-2HPnD>
- [23] Ubiquiti Networks (2017, agosto) productos [Online]. Disponible en: <https://www.ubnt.com/airmax/airmax-sector-antenna/>
- [24] Phocos (2017, agosto) Portafolio [informativo] disponible en: <http://www.phocos.com/blog/portfolio/cml-mppt-5010/>

[25] yinglilar (2017, agosto) Productos [informativo comercial] Disponible en:  
<http://www.yinglilar.com/al/products/multicrystalline/yge-72-cell-series/>

## ANEXOS

### Interprete de mensaje en concentrador

```

import serial
import time
import requests
#import random
import MySQLdb
from datetime import datetime
from random import random
def enviamensaje(placa,lat,lon):
    fecha=""
    longitud=lon
    latitud=lat
    placa2='GGG-7777'
    fecha= datetime.now().ctime()[0:11]
    hora= datetime.now().ctime()[11:19]
    if (lat==0 or lon==0):
        longitud=str(random()*-100)[0:9]
        latitud=str(random()*-100)[0:9]
    else:
        latitud=lat
        longitud=lon
    cursor.execute ("INSERT INTO accidente (idnodo, idvehiculo, fecha, hora, estado, latitud,
longitud) VALUES(2, '"+placa2+"','"+fecha+"','"+hora+"','A','"+str(latitud)+"','"+str(longitud)+"'")")
    conn.commit ()
    r=requests.post('http://192.168.43.127:8000/newnode', data = {'latitud':123, 'longitud':342});
port = serial.Serial(
    port='/dev/ttyUSB0',
    baudrate=9600,
    parity=serial.PARITY_NONE,
    stopbits=serial.STOPBITS_ONE,
    bytesize=serial.EIGHTBITS,
)
conn = MySQLdb.connect (
    host = "192.168.43.127",
    port=3306,
    user = "root",
    passwd = "3davisitooo",
    db = "intdesa")
band = 0
i=0
mensaje=""
latitud=""
longitud=""
inicio=~'
delimitador='&'
lat='$'
lon='*'
val = "

```

```

b1=0
b2=0
b3=0
b4=0
i=1
j=0
k=1
cursor = conn.cursor ()
port.reset_input_buffer()
while not band:
    val=port.read()
    time.sleep(0.050)
    if (b1==1 and b2 ==1 and val !=delimitador):
        if (val== lat):
            b3=1
        elif(val==lon):
            b3=0
            b4=1
        elif(b3==1):
            latitud+=val
        elif(b4==1):
            longitud+=val
        else:
            mensaje += val
    elif (b1==1 and b2 ==1 and val ==delimitador):
        latitud=float(latitud)/-100
        longitud=float(longitud)/-100
        print ("mensaje: " + mensaje +"latitud: "+str(latitud)+" longitud: " + str(longitud)+"\n")
        #print datetime.now().ctime()
        enviamensaje(mensaje,latitud,longitud)
        b1=0
        b2=0
        b3=0
        b4=0
        mensaje=""
        longitud=""
        latitud=""
    elif (b1==1 and b2 ==0 and val == delimitador):
        b2=1
    elif (b1==0 and val==inicio):
        b1=1
port.close()
cursor.close ()
conn.close ()

```

### Código detector de evento y lector GPS

```

#include <Adafruit_GPS.h>
#include <SoftwareSerial.h>
SoftwareSerial mySerial(3, 2);
Adafruit_GPS GPS(&mySerial);
#define GPSECHO false
boolean usingInterrupt = false;
void useInterrupt(boolean); // Func prototype keeps Arduino 0023 happy
long tiempo;
int x,y,z;
int disparador = 7; // trigger
int entrada = 8; // echo
float distancia2;
int valor;
String longitud="0", latitud="0";
void setup()
{
  pinMode(disparador, OUTPUT);
  pinMode(entrada, INPUT);
  analogReference(EXTERNAL);
  Serial.begin(115200);
  Serial.println("Adafruit GPS library basic test!");
  GPS.begin(9600);
  GPS.sendCommand(PMTK_SET_NMEA_OUTPUT_RMCGGA);
  GPS.sendCommand(PMTK_SET_NMEA_UPDATE_1HZ); // 1 Hz update rate
  GPS.sendCommand(PGCMD_ANTENNA);
  useInterrupt(true);
  delay(1000);
  mySerial.println(PMTK_Q_RELEASE);
}
SIGNAL(TIMER0_COMPA_vect) {
  char c = GPS.read();
  // if you want to debug, this is a good time to do it!
#ifdef UDR0
  if (GPSECHO)
    if (c) UDR0 = c;
    // writing direct to UDR0 is much much faster than Serial.print
    // but only one character can be written at a time.
#endif
}

void useInterrupt(boolean v) {
  if (v) {
    OCR0A = 0xAF;
    TIMSK0 |= _BV(OCIE0A);
    usingInterrupt = true;
  } else {
    TIMSK0 &= ~_BV(OCIE0A);
    usingInterrupt = false;
  }
}

```

```

    }
  }

  uint32_t timer = millis();
  void loop()          // run over and over again
  {

    //READING VALUES
    x=analogRead(A0);
    y=analogRead(A1);
    z=analogRead(A2);
    if(x>=900 or y>=900 or z>=900 or x<=30 or y<=30 or z<=30){
      distancia2 = GetDistancia();
      if (distancia2>2 and distancia2<10){
        // in case you are not using the interrupt above, you'll
        // need to 'hand query' the GPS, not suggested :(
        if (! usingInterrupt) {
          // read data from the GPS in the 'main loop'
          char c = GPS.read();
          // if you want to debug, this is a good time to do it!
          if (GPSECHO)
            if (c) Serial.print(c);
        }

        // if a sentence is received, we can check the checksum, parse it...
        if (GPS.newNMEAreceived()) {
          if (!GPS.parse(GPS.lastNMEA())) // this also sets the newNMEAreceived() flag to
false
            return; // we can fail to parse a sentence in which case we should just wait for
another
          }
          if (timer > millis()) timer = millis();
          if (millis() - timer > 2000) {
            timer = millis(); // reset the timer
            if (GPS.fix) {
              latitud=(GPS.latitude);
              longitud=(GPS.longitude);
              Serial.print("~&GGG-7777$");
              Serial.print(latitud);
              Serial.print ("*");
              Serial.print (longitud);
              Serial.println("&");
            }else{
              Serial.print("~&GGG-7777$");
              Serial.print(latitud);
              Serial.print ("*");
              Serial.print (longitud);
              Serial.println("&()");
            }
          }
        }
        delay (70);
      }else if (distancia2>300 or distancia2 <2){
        delay (70);
      }
    }
  }
}

```

```
        }else{
            delay (10);
        }
    }
float GetDistancia(){
    long tiempo;
    float distancia;
    digitalWrite(disparador, HIGH);
    delayMicroseconds(10);
    digitalWrite(disparador, LOW);
    tiempo = (pulseIn(entrada, HIGH)/2);
    distancia = float(tiempo * 0.0343);
    return distancia;
}
```

## Datasheet de fabricante

### Ultrasonic Ranging Module HC - SR04

#### Product features:

Ultrasonic ranging module HC - SR04 provides 2cm - 400cm non-contact measurement function, the ranging accuracy can reach to 3mm. The modules includes ultrasonic transmitters, receiver and control circuit. The basic principle of work:

- (1) Using IO trigger for at least 10us high level signal,
- (2) The Module automatically sends eight 40 kHz and detect whether there is a pulse signal back.
- (3) IF the signal back, through high level , time of high output IO duration is the time from sending ultrasonic to returning.

Test distance = (high level time×velocity of sound (340M/S) / 2,

#### Wire connecting direct as following:

5V Supply

Trigger Pulse Input

Echo Pulse Output

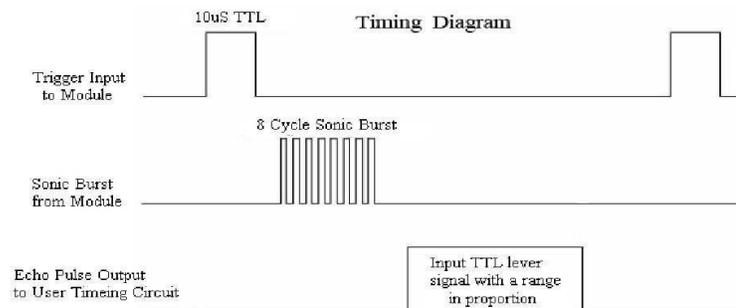
0V Ground



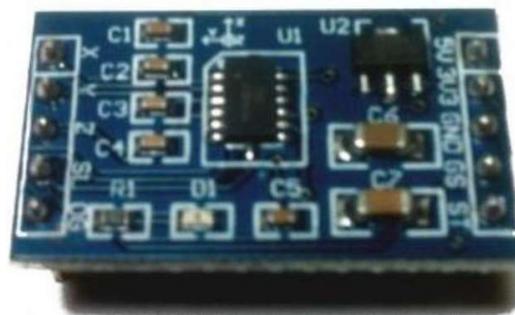
### Trig Echo GND

#### Timing diagram

The Timing diagram is shown below. You only need to supply a short 10uS pulse to the trigger input to start the ranging, and then the module will send out an 8 cycle burst of ultrasound at 40 kHz and raise its echo. The Echo is a distance object that is pulse width and the range in proportion. You can calculate the range through the time interval between sending trigger signal and receiving echo signal. Formula:  $\mu\text{S} / 58 = \text{centimeters}$  or  $\mu\text{S} / 148 = \text{inch}$ ; or: the range = high level time \* velocity (340M/S) / 2; we suggest to use over 60ms measurement cycle, in order to prevent trigger signal to the echo signal.



#### MMA7361 3-Axis Accelerometer Module



### Pin Configuration:



### Description of Pins:

X-axis	This is the analog signal out along the X-axis
Y-axis	This is the analog signal out along the Y-axis
X-axis	This is the analog signal out along the Z-axis
Sleep	This pin will put the chip to sleep when pulled low to consume less power and will resume operation when pulled high.
0g detect	This pin will go high when 0g is detected on all 3 axis. Useful to detect free fall
5v in	This pin is attached to a built in regulator that will bring the 5v down to the 3.3v that the chip is required to run on.
3.3v in	This pin bypasses the regulator for 5v in for those who have a pre-regulated 3.3v bus.
Ground	This pin must be connected to the ground of the circuit.
Sense select	This pin is used to select between the two sensitivities. If this pin is low it is in 1.5g mode. When high, it switches to 6g mode.