



FACULTAD DE INGENIERÍA Y CIENCIAS APLICADAS

SISTEMA DE MONITOREO EN TIEMPO REAL PARA BICICLETAS

AUTORAS

CINTYA MAILY CORDERO VILLACORTE  
ANDREA MICHAELA SALGADO CUEVA

AÑO

2019



FACULTAD DE INGENIERÍA Y CIENCIAS APLICADAS

SISTEMA DE MONITOREO EN TIEMPO REAL PARA BICICLETAS.

Trabajo de Titulación presentado en conformidad con los requisitos establecidos para optar por el título de Ingenieras en Electrónica y Redes de Información.

Profesor Guía

MSc. Jorge Luis Rosero Beltrán

Autoras

Cintya Maily Cordero Villacorte.

Andrea Michaela Salgado Cueva

Año

2019

## **DECLARACIÓN DEL PROFESOR GUÍA**

"Declaro haber dirigido el trabajo, Sistema de Monitoreo en Tiempo Real para Bicicletas, a través de reuniones periódicas con las estudiantes Cintya Maily Cordero Villacorte y Andrea Michaela Salgado Cueva, en el semestre 201910, orientando sus conocimientos y competencias para un eficiente desarrollo del tema escogido y dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación".

---

Jorge Luis Rosero Beltrán

Master en Ciencias con Especialidad en Automatización

C.I.:1803610185

## **DECLARACIÓN DEL PROFESOR CORRECTOR**

"Declaro haber revisado este trabajo, Sistema de Monitoreo en Tiempo Real para Bicicletas, de Cintya Maily Cordero Villacorte y Andrea Michaela Salgado Cueva, en el semestre 201910, dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación".

---

David Fernando Pozo Espín  
Master Universitario en Automática y Robótica  
C.I.: 1717340143

## **DECLARACIÓN DE AUTORÍA DEL ESTUDIANTE**

“Declaro que este trabajo es original, de mi autoría, que se han citado las fuentes correspondientes y que en su ejecución se respetaron las disposiciones legales que protegen los derechos de autor vigentes.”

---

Cintya Maily Cordero Villacorte  
C.I.: 0401514104

---

Andrea Michaela Salgado Cueva  
C.I.: 1752068435

## AGRADECIMIENTO

Agradezco a Dios por guiarme siempre y ayudarme a concluir este trabajo de titulación. A mis padres ya que siempre cuento con su apoyo en todo lo que me propongo realizar, este logro es para ustedes.

A mi tutor el Ing. Jorge Rosero por ayudarnos a desarrollar el trabajo de titulación y guiarnos para tomar las mejores decisiones.

A mi amiga Michaela Salgado por su tiempo y dedicación para nuestro trabajo de titulación.

Cintya Cordero.

## AGRADECIMIENTO

Agradezco a Dios por brindarme la sabiduría y las fuerzas necesarias para culminar esta etapa de mi carrera.

A mi madre Soledad y a mi hermana Gaby gracias por estar conmigo en cada paso que doy y por apoyarme constantemente, a mi abuelito Ing. Carlos Cueva que desde el cielo me inspira a seguir adelante y nunca darme por vencida ante cualquier obstáculo y a mi enamorado, Bryan Ramos, gracias por estar a mi lado y por todos los consejos que me sirven para crecer como persona.

Michaela Salgado.

## DEDICATORIA

A mis padres Iván y Silvana por su amor incondicional y el apoyo brindado a lo largo de mi carrera universitaria, a mis hermanitos Erick y Paula quienes son mi inspiración en todo momento. A mis abuelitos por estar pendientes de todos mis logros, mis tías Lucía y Yuliana de quienes siempre he recibido su apoyo, en especial a mi tío Edison porque me inculcó el gusto por esta carrera.

A mis amigos y amigas más cercanos por apoyarme en todo momento durante mis estudios y darme ánimo para seguir adelante con mis sueños.

Cintya Cordero.



## DEDICATORIA

A mi madre Soledad, por estar conmigo en cada momento, por su amor incondicional, por creer en mí y sólo me queda decirle que te amo mucho y que siempre daré todo de mi para verte feliz, eres mi ejemplo que seguir. A mi hermana Gaby por siempre estar pendiente de mí y por nunca dejarme sola, a mis sobrinos Theo y Domenika por todos los buenos deseos. A mi abuelita Martha por todo el cariño y por siempre estar orgullosa de mis logros, a mi angelito Carlos, sé que desde el cielo estás muy orgullo de mí. A Bryan, mi enamorado, por siempre brindarme su apoyo y nunca soltar mi mano, te amo.

Michaela Salgado.

## RESUMEN

Hoy en día las bicicletas se han convertido en un medio de transporte alternativo para la sociedad, debido a sus beneficios en salud y al medio ambiente. La calidad de las bicicletas se ha ido incrementado por las importantes características que tiene, aumentando el costo y por esta razón el robo de mismas. Además, el respeto por los ciclistas no se muestra en las vías y han desencadenado en una gran cantidad de accidentes que pueden llegar a la muerte. Por los factores antes mencionados los usuarios tienen la necesidad de buscar un sistema que permita conocer la ubicación en Tiempo real y una notificación ante cualquier accidente.

Este proyecto resuelve las necesidades de los usuarios diseñando un prototipo de monitoreo en tiempo real con un sensor que permite alertar cuando se presenta un accidente. El prototipo está integrado con una página web para visualizar rutas, estadística de velocidad y altura.

Este sistema de monitoreo en tiempo real está formado por un microcontrolador de 8 bits como placa base, un módulo GPS, un módulo GSM y un sensor compuesto de un acelerómetro y un giroscopio.

El GPS da a conocer la ubicación de la bicicleta, así como también velocidad y altura. El GSM mediante comandos AT permiten realizar llamadas, mensajes y conexión a internet, lo cual permite enviar la información a la base de datos. El sensor mide la variación del movimiento de la bicicleta para conocer cuando se produjo una caída y notificarle al usuario.

El prototipo interactúa con el aplicativo web ([www.tubicisegura.com](http://www.tubicisegura.com)), tiene autenticación de usuario para acceder a dos perfiles: 1) Administrador y 2) Ciclista, que deben ser administrados según las características de cada uno. El perfil del ciclista permite monitorear el dispositivo en tiempo real y guardar las rutas dependiendo de una fecha definida para posteriores consultas y muestra datos estadísticos de velocidad y altura. El perfil Administrador permite crear y administrar los usuarios.

Además, se realizó una serie de pruebas que permitieron validar los objetivos propuestos para el prototipo, una de ellas fue realizar tres tipos de rutas en

diferentes lapsos de tiempo, esto sirvió para conocer la robustez y la fiabilidad de la base de datos ya que esto permitió realizar un promedio de las coordenadas obtenidas para conocer cuántas se perdieron.

El aplicativo web cuenta con una pestaña de monitoreo que permite conocer la ubicación en tiempo real del ciclista, esto se comprobó movilizándolo de un lugar a otro, en la página se visualiza el movimiento del ciclista.

Finalmente, se realizó una prueba con el sensor MPU6050, en la cual si varía la inclinación del ciclista dependiendo de los ejes x, y, esto provoca a encender un led lo cual representa un accidente y en esta acción el prototipo notificará al usuario mediante un correo electrónico.

## ABSTRACT

Today, bicycles have become an alternative means of transport for society, due to their benefits in health and the environment. The quality of the bicycles has been increased by the important characteristics that it has, increasing the cost and for this reason the theft of them. In addition, respect for cyclists is not shown on the tracks and have triggered a large number of accidents that can lead to death. Due to the aforementioned factors, users have the need to look for a system that allows knowing the location in real time and notification of any accident.

This project solves the needs of users by designing a real-time monitoring prototype with a sensor that alerts when an accident occurs. The prototype is integrated with a web page to visualize routes, statistics of speed and height.

This real-time monitoring system consists of an 8-bit microcontroller as a motherboard, a GPS module, a GSM module and a sensor composed of an accelerometer and a gyroscope.

The GPS discloses the location of the bicycle, as well as speed and height. The GSM through AT commands allow calls, messages and internet connection, which allows sending the information to the database. The sensor measures the variation of the movement of the bicycle to know when a fall occurred and notify the user.

The prototype interacts with the web application ([www.tubicisegura.com](http://www.tubicisegura.com)), has user authentication to access two profiles: 1) Administrator and 2) Cyclist, which must be managed according to the characteristics of each one. The profile of the cyclist allows to monitor the device in real time and save the routes depending on a defined date for later consultations and displays statistical data of speed and height. The Administrator profile allows you to create and manage users.

In addition, a series of tests was carried out that allowed to validate the proposed objectives for the prototype, one of them was to carry out three types of routes in different time periods, this served to know the robustness and reliability of the database since this allowed to perform an average of the coordinates obtained to know how many were lost.

The web application has a monitoring tab that allows knowing the location in real time of the cyclist, this was verified by moving the prototype from one place to another, on the page the movement of the cyclist is visualized.

Finally, a test was carried out with the MPU6050 sensor, in which if the inclination of the cyclist varies depending on the axes x, y, this proceeds to turn on a led which represents an accident and in this action the prototype will notify the user through an-email.

# ÍNDICE

|  |    |
|--|----|
| 1.Introducción .....                             | 13 |
| 1.1. Antecedentes .....                          | 13 |
| 1.2. Alcance .....                               | 14 |
| 1.3. Justificación .....                         | 15 |
| 1.4. Objetivo general.....                       | 15 |
| 1.5. Objetivos específicos.....                  | 16 |
| 2.Marco Teórico.....                             | 16 |
| 2.1. Microcontrolador .....                      | 16 |
| 2.2. Arduino.....                                | 17 |
| 2.3. GPS.....                                    | 18 |
| 2.4. GSM .....                                   | 19 |
| 2.5. Unidad Inercial .....                       | 19 |
| 2.6. Batería.....                                | 19 |
| 2.7. Entorno de desarrollo integrado (IDE) ..... | 20 |
| 2.8. Lenguaje de programación.....               | 21 |
| 2.9. Base de datos .....                         | 22 |
| 2.10. Servidor web .....                         | 23 |
| 2.11. Diagrama software.....                     | 23 |
| 2.12. Diagrama hardware .....                    | 24 |
| 3.Diseño.....                                    | 25 |
| 3.1. Diseño hardware.....                        | 25 |
| 3.1.1. GSM .....                                 | 28 |

|  |    |
|--|----|
| 3.1.2. GPS.....                                    | 30 |
| 3.1.3. Sensor MPU6050 .....                        | 32 |
| 3.1.4 Batería .....                                | 33 |
| 3.2. Implementación del prototipo .....            | 34 |
| 3.3. Diseño de software prototipo.....             | 37 |
| 3.4. Diseño de software página web .....           | 40 |
| 3.4.1. Hosting .....                               | 41 |
| 3.4.2. IDE (Entorno de desarrollo integrado) ..... | 43 |
| 3.4.3. Lenguaje de programación.....               | 44 |
| 3.4.4. Base de datos .....                         | 46 |
| 4.Pruebas y Resultado.....                         | 60 |
| 5.Conclusiones y Recomendaciones.....              | 70 |
| 5.1. Conclusiones .....                            | 70 |
| 5.2. Recomendaciones.....                          | 71 |
| REFERENCIAS .....                                  | 73 |
| ANEXOS.....  | 75 |

## ÍNDICE DE FIGURAS

|   |    |
|---|----|
| Figura 1. Plano de desarrollo de GPS con relación a satélites. .... | 18 |
| Figura 2. Componentes de la batería. ....                           | 20 |
| Figura 3. Lenguajes de programación. ....                           | 21 |
| Figura 4. Tipos de base de datos. ....                              | 22 |
| Figura 5. Diagrama de bloques de diseño de software. ....           | 24 |
| Figura 6. Diagrama de bloques de diseño de hardware. ....           | 24 |
| Figura 7. Pines utilizados del microcontrolador. ....               | 34 |
| Figura 8. Diagrama de leds utilizados. ....                         | 36 |
| Figura 9. Conexión componentes prototipo. ....                      | 37 |
| Figura 10. Código conexión GSM. ....                                | 38 |
| Figura 11. Código de enlace prototipo página web. ....              | 38 |
| Figura 12. Diagrama de flujo backup microSD. ....                   | 39 |
| Figura 13. Diagrama de flujo unidad inercial. ....                  | 40 |
| Figura 14. Página de inicio, aplicativo web. ....                   | 49 |
| Figura 15. Sección de información del proyecto. ....                | 49 |
| Figura 16. Mapa de ubicación del local. ....                        | 50 |
| Figura 17. Login de la página web. ....                             | 50 |
| Figura 18. Diagrama de flujo, login página web. ....                | 51 |
| Figura 19. Registro de un nuevo usuario. ....                       | 52 |
| Figura 20. Error de ingreso de un nuevo ciclista. ....              | 53 |
| Figura 21. Menú de perfil ciclista. ....                            | 54 |
| Figura 22. Selección de la búsqueda mediante del calendario. ....   | 54 |
| Figura 23. Gráfica de ruta con línea en el mapa. ....               | 55 |
| Figura 24. Calendario y selección de ruta. ....                     | 55 |
| Figura 25. Estadísticas de velocidad y altura. ....                 | 56 |
| Figura 26. Diagrama lógico base de datos. ....                      | 57 |
| Figura 27. Modelo físico base de datos. ....                        | 60 |
| Figura 28. Recepción de coordenadas en formato NMEA. ....           | 61 |
| Figura 29. Recepción de coordenadas formato TinyGPS. ....           | 61 |
| Figura 30. Código del ajuste de la zona horario de Ecuador. ....    | 62 |



|  |    |
|--|----|
| Figura 31. Información errónea en la base de datos. ....                   | 62 |
| Figura 32. Información correcta en la base de datos. ....                  | 62 |
| Figura 33. Envío de datos a la MicroSD. ....                               | 63 |
| Figura 34. Escribir en la MicroSD. ....                                    | 63 |
| Figura 35. Coordenas enviadas a la base de datos mediante el MicroSD. .... | 64 |
| Figura 36. Información del correo electrónico al usuario. ....             | 64 |
| Figura 37. Mensaje de alerta de un accidente del usuario. ....             | 65 |
| Figura 38. Gráfica del accidente en el mapa. ....                          | 65 |
| Figura 39. Falsa alerta de accidente. ....                                 | 66 |
| Figura 40. Alerta de accidente y envío de correo electrónico. ....         | 66 |
| Figura 41. Gráfica de la ruta de 15 minutos. ....                          | 68 |
| Figura 42. Gráfica de la ruta de 1 hora. ....                              | 68 |
| Figura 43. Gráfica de la ruta de 30 minutos. ....                          | 69 |

## ÍNDICE DE TABLAS

|   |    |
|---|----|
| Tabla 1. Tabla de requerimientos para el microcontrolador.....                | 26 |
| Tabla 2. Tabla comparativa microcontroladores.....                            | 26 |
| Tabla 3. Análisis de requerimientos en contraste a productos disponibles..... | 27 |
| Tabla 4. Requerimiento del GSM.....   | 28 |
| Tabla 5. Tabla relación característica módulo.....                            | 29 |
| Tabla 6. Tabla de análisis requerimiento módulo.....                          | 30 |
| Tabla 7. Tabla requerimientos GPS.....  | 30 |
| Tabla 8. Tabla relación característica módulo.....                            | 31 |
| Tabla 9. Tabla requerimiento módulo.....                                      | 31 |
| Tabla 10. Tabla requerimiento sensor.....                                     | 32 |
| Tabla 11. Tabla relación módulo característica.....                           | 32 |
| Tabla 12. Tabla relación requerimiento módulo.....                            | 33 |
| Tabla 13. Amperaje total del prototipo.....                                   | 33 |
| Tabla 14. Tabla conexión pines.....   | 34 |
| Tabla 15. Tabla relación ítem descripción.....                                | 40 |
| Tabla 16. Tabla de requerimientos de hosting.....                             | 41 |
| Tabla 17. Tabla comparativa de servicios de hosting.....                      | 41 |
| Tabla 18. Tabla relación de características de servicios.....                 | 42 |
| Tabla 19. Tabla relación requisitos justificación.....                        | 43 |
| Tabla 20. Tablas características de IDE.....                                  | 43 |
| Tabla 21. Tabla relación características IDE.....                             | 44 |
| Tabla 22. Tabla relación requisitos justificación.....                        | 45 |
| Tabla 23. Tabla de características de lenguajes de programación.....          | 45 |
| Tabla 24. Tabla relación requerimiento lenguaje de programación.....          | 46 |
| Tabla 25. Tabla relación requisitos justificación.....                        | 47 |
| Tabla 26. Tabla de características base de datos.....                         | 47 |
| Tabla 27. Tabla relación requerimiento base de datos.....                     | 48 |
| Tabla 28. Tabla de diccionario de datos de la tabla usuario.....              | 57 |
| Tabla 29. Tabla de diccionario de datos de la tabla gps.....                  | 58 |
| Tabla 30. Tabla de diccionario de datos de la tabla fotos.....                | 59 |

|   |    |
|---|----|
| Tabla 31. Tabla de rutas y tiempo .....                   | 67 |
| Tabla 32. Análisis de rutas, dependiendo del tiempo. .... | 67 |

## **1. Introducción**

El proyecto tiene como finalidad el diseño de un prototipo que permita conocer la ubicación en tiempo real de una bicicleta y un sensor que alerte al usuario ante un accidente. Además, cuenta con una página web que permite a los ciclistas conocer sus rutas definidas por fechas y estadísticas de velocidad y altura.

En el capítulo II, se detalla el marco teórico, que define cada componente necesario para el desarrollo del prototipo y de la página web, así mismo, muestra diagramas de bloques que permite conocer como está constituido. Estos conceptos presentados en el capítulo I son necesarios para elaborar el capítulo III, en el cual se detalla el diseño, que consta de dos partes hardware y software. En la primera se muestra el diseño del prototipo y en la segunda parte se define la programación de los componentes, la creación de la página web y la interacción con la base de datos.

También se realizaron pruebas y análisis de los resultados del prototipo en donde se verifica el comportamiento realizando un contraste con los objetivos propuestos para verificar el correcto funcionamiento, todo esto es definido en el capítulo IV. Finalmente, en el capítulo V, se detalla conclusiones y recomendaciones que se obtuvo a lo largo del proceso del diseño e implementación del prototipo de monitoreo en tiempo real.

### **1.1. Antecedentes**

Durante los últimos años el potencial de crecimiento del mercado de bicicletas en el Ecuador responde a la necesidad de los ecuatorianos para obtener soluciones de transporte más eficiente y económicos. Sin duda este medio de transporte también representa ciertas desventajas, una de las más representativas son los robos y los accidentes, según el diario El Comercio en el mes de octubre del 2017 fueron robadas dos bicicletas de gama alta en Quito, su precio oscilaba entre los 3000 a 5000 dólares. Este tipo de bicicletas representa una gran inversión por su alto costo y por el uso que se les da a las mismas, posteriormente estas bicicletas fueron encontradas desarmadas y en

cajas en un carro que quería cruzar a Colombia para allá comercializarlas. La policía comentó que se trataba de una red de delincuentes que estaban realizando estos robos constantemente (Comercio, 2017).

Diferentes proyectos se han especificado y desarrollado, para atender este tipo de emergencias con Arduino, por lo que la marca inglesa SpyBike ofrece dispositivos de alta tecnología para el rastreo de bicicletas. Además, la empresa cuenta con dos propiedades que lo caracterizan: activar y seguro. El primer aspecto se refiere a que el dispositivo puede ser activado remotamente por un correo o un mensaje de texto y por otro lado para la seguridad Spybike enviará un mensaje de texto cuando la bicicleta se ponga en movimiento y un servicio de seguimiento en tiempo real, que se puede visualizar en la aplicación móvil de SpyBike, que se complementa con Google Earth, todo esto se da gracias a que el sistema trabaja con una tarjeta SIM o chip de celular. Su precio es de 173 dólares aproximadamente y está pensada para bicicletas de gama alta (Bikelec, 2014).

Otro proyecto de diseño de un prototipo de monitoreo de bicicletas, realizado por Fabián Murillo, quien diseñó un sistema de monitorio de la bicicleta por medio de un GPS y un sensor magnético el cual es el encargado de obtener el evento de robo de esta y reportándolo al usuario por medio de tecnología GPSR/GSM (Briceño Murillo, 2016)

## **1.2. Alcance**

Este trabajo de titulación está concentrado en la elaboración de un prototipo electrónico aplicado para bicicletas el cual estará construido mediante un módulo GSM que se conectará con la red móvil para enviar los datos del GPS que se encuentra enlazado a un microcontrolador para que de esta manera indique la ubicación actual. Además, el prototipo tiene un sensor acelerómetro que permitirá saber si el ciclista sufrió una caída o un golpe fuerte, es decir esta

acción se notificará a una persona de confianza vía correo electrónico para de esta manera ayudar al ciclista accidentado.

En el caso de un robo, el usuario puede visualizar los datos de las rutas que se realizó en el día ya que se almacenarán en una base de datos la cual interactúa con la página web, la página se autenticará mediante usuario y contraseña, que el usuario podrá acceder a la misma en el momento que necesite.

### **1.3. Justificación**

Este prototipo está enfocado directamente en los ciclistas ya que según “El telégrafo” en nuestro país está creciendo el uso de la bicicleta considerablemente por ser un medio de transporte amigable con el ambiente. También se toma en cuenta la seguridad de los ciclistas por la cantidad de accidentes y robos que existen en la ciudad (El Telégrafo, 2016).

Este prototipo está diseñado para proteger al ciclista y la bicicleta ante cualquier escenario de robo o accidente, logrará mantener una conexión entre el ciclista y su bicicleta ya que este prototipo ayudará al ciclista a mantener su medio de transporte seguro, es decir, que sabrá las rutas exactas, estará al corriente de la velocidad y estar al tanto de las altitudes que tuvo cada día. Todo este resultado estará en una página web que el ciclista puede acceder desde cualquier lado y a cualquier hora. Esto facilita el rastreo del ciclista y la bicicleta ante un robo o accidente.

En la actualidad, más del 90% de todos los procesos en la industria están basados directa o indirectamente en la tecnología. Este es el caso de las operadoras móviles ya que una caída en sus servicios significaría un gran impacto sobre sus usuarios y esto a su vez daría como resultado la interrupción del servicio y por ende pérdidas económicas.

### **1.4. Objetivo general**

Implementar un sistema de rastreo satelital en tiempo real para bicicletas a través de una aplicación web, usando GSM.

### 1.5. Objetivos específicos

- Implementar un prototipo electrónico para el envío de la ubicación de la bicicleta en tiempo real.
- Desarrollar una aplicación web que permita visualizar la posición de la bicicleta en tiempo real.
- Realizar pruebas de funcionamiento del sistema implementado en un escenario real.

## 2. Marco Teórico

En el siguiente capítulo se realiza una descripción general de los componentes del sistema de monitoreo en tiempo real para bicicletas. Además, se muestran los diagramas de bloques en los que se especifica la relación que existen en los componentes de software y hardware.

### 2.1. Microcontrolador

Un microcontrolador es un circuito digital integrado conformado básicamente por un procesador (CPU), memorias y puertos de entrada/salida. Estos componentes, junto con otros más, se encuentran embebidos en una pequeña pastilla de silicio. En la actualidad estos dispositivos son reprogramables, lo que permite utilizarlos en diferente tipo de aplicaciones. El CPU es el encargado de realizar operaciones aritméticas, lógicas y de control, así como revisar la correcta ejecución de las instrucciones, conocidas también como programa (Torrente Artero, 2013).

Las memorias en un microcontrolador pueden ser de tres tipos: memoria SRAM, FLASH y EEPROM. La primera es una memoria volátil, utilizada para guardar datos de manera temporal y también contiene en sus registros las funcionalidades asociadas a los periféricos del microcontrolador. Mientras tanto, la FLASH permite almacenar el programa desarrollado por el usuario, y puede ser reprogramado según la aplicación que se le vaya a dar. Finalmente, la

EEPROM es una memoria no volátil, que permite almacenar datos de manera permanente y es usada para respaldar información de muy alta importancia (Valdés Pérez & Pallàs Areny, 2007).

Por otra parte, los puertos de entrada y salida pertenecen al grupo de los periféricos, los mismos que permiten la interacción entre el microcontrolador con el mundo exterior. Son utilizados para el ingreso y salida de datos a través de sensores y dispositivos de visualización (Torrente Artero, 2013).

En la actualidad los microcontroladores se encuentran en dispositivos o equipos que se usan cotidianamente, por ejemplo, los microondas, vehículos, teléfonos celulares, cámaras digitales, entre otros. Todos estos equipos cuentan con un cierto grado de automatización e incluso inteligencia que les permite facilitar las tareas a los seres humanos.

## **2.2. Arduino**

Arduino es una plataforma de prototipos electrónicos de hardware y software de código abierto que ayuda a los programadores a manejar esta herramienta de una forma más sencilla. El principal objetivo es crear una herramienta de hardware que sea fácil de programar por los usuarios y que no sea de costo elevado. Además, cuenta con un entorno de desarrollo integrado (IDE) y una herramienta de software que traduce el lenguaje de alto nivel (C++) a un lenguaje de bajo nivel o de máquina. Arduino fue creado con el principio de open source, es decir, que todos los programas son de dominio público, que sea libre para copiar y modificar conservando los derechos de autoría (Caicedo Pedrera, 2017).

El funcionamiento del Arduino se lo puede definir en cuatro partes importantes:

- Conexiones de entrada: recibir datos del exterior.
- Microcontrolador: es el cerebro del Arduino en donde se encarga de procesar los datos recibidos del exterior y los interpreta para que pueda decidir qué respuesta enviar.
- Conexiones de salida: es la salida de los datos que fueron programados con anterioridad por el usuario, así dando una respuesta a motores, pantallas o demás dispositivos conectados.



- Puertos/Buses de comunicación: SPI en la placa y ethernet, wifi, etc (Caicedo Pedrera, 2017)

Arduino es utilizado para realizar diferentes aplicaciones en distintas áreas como en la domótica que sirve para automatizar una casa o un sistema de riego, en la robótica creando proyectos para la construcción de robots inteligentes y en la telemática desarrollando aplicaciones de control por radiofrecuencia.

### 2.3. GPS

El sistema posicionamiento global (GPS) diseñado por el Departamento de Defensa de los Estados Unidos con fines militares para lograr la estimación de valores acerca de velocidad y tiempo mediante la triangulación, la altitud, la longitud y latitud de cualquier objeto del globo terrestre.

Como se muestra en la Figura 1. El objetivo de un sistema GPS es calcular la posición en cualquier punto del espacio en las coordenadas  $(x, y, z)$ , para obtener el cálculo de las distancias se debe realizar con mínimo 4 satélites. La distancia entre el receptor y el satélite se obtiene multiplicando el tiempo de vuelo del satélite por la velocidad de propagación.

A nivel civil se ha logrado obtener mejor precisión gracias al uso de varios algoritmos, estas aplicaciones han experimentado un gran crecimiento y en la actualidad se cuenta con más de 70 fabricantes de recepción GPS (Pozo Ruz, y otros, 2018).

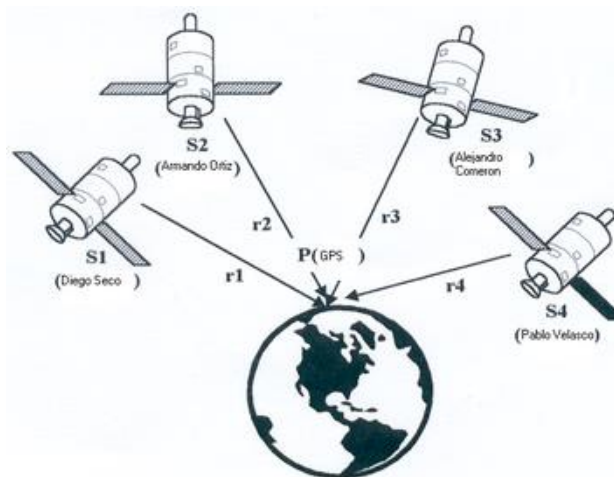


Figura 1. Plano de desarrollo de GPS con relación a satélites.

Tomado de (Pozo Ruz, y otros, 2018).

## **2.4. GSM**

GSM “*Global System for Mobile*” es una tecnología de radiofrecuencia utilizada para las comunicaciones móviles, soporta 2G, 2.5G y 3G que hacen referencia a las generaciones de telefonía móvil. Permite a los dispositivos integrar, dentro de una misma señal voz, mensajes de texto y datos.

Se utiliza básicamente para comunicación e intercambio de datos a través de dispositivos móviles (teléfonos celulares) que se enlazan a una estación base y a su vez a un sistema que administra la ruta para los diferentes servicios.

Esta tecnología permite realizar actividades cotidianas como llamar, enviar mensajes o conexiones a internet, tiene características relevantes como: funciona entre diferentes operadoras, posee roaming internacional y un SIM (Subscriber Identity Module) que permite que el dispositivo sea personalizado, ya que cuenta con un número único de identificación para llamadas e información del operador de red.

## **2.5. Unidad Inercial**

Es un sistema integrado por 6 ejes, usa la combinación de acelerómetro y giroscopio en 3 ejes respectivamente. Controla posición, velocidad y aceleración de un cuerpo, El acelerómetro mide los cambios de velocidad y el giroscopio mide velocidades de giro y da direcciones establecidas, un error de un minuto en una medición puede causar gran cantidad de errores en la información recibida por lo que es necesario que al construirse se mantenga normas estrictas, alineados de la manera correcta y reiniciarse cada intervalo de tiempo. Las unidades inerciales se usan actualmente en aviones, cohetes y satélites (Gregersen, 2018).

## **2.6. Batería**

Es un acumulador de energía compuesta comúnmente por una celda galvánica, encargada de la conversión de energía química a energía eléctrica. En la Figura 2, se muestra los componentes básicos de la batería, cada celda tiene un cátodo o terminal positiva y un ánodo o terminal negativo estos se encuentran separados por un electrolito que permite un flujo de iones, entre los terminales y mediante

el proceso anterior permite desarrollar suficiente fuerza voltaje y corriente para permitir el funcionamiento de los dispositivos. Toda la parte activa de la batería se encuentra en un ensamblaje que mantiene el aire afuera. La utilidad está limitada no solo por la capacidad sino también por la rapidez que se puede extraer la corriente, además están diseñadas basándose en la influencia del mercado, la calidad de los materiales y su complejidad, lo mencionado anteriormente es para cumplir los requerimientos del producto y garantizar su correcto funcionamiento (Britannica, 2018) .

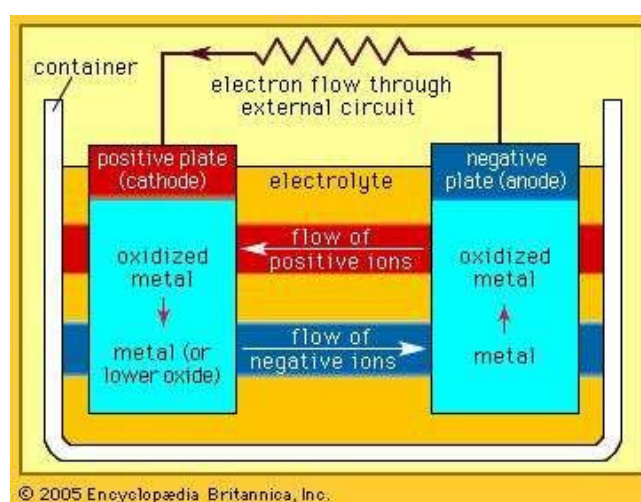


Figura 2. Componentes de la batería.

Tomado de (Britannica, 2018).

## 2.7. Entorno de desarrollo integrado (IDE)

Es una plataforma de programación para crear y ejecutar los códigos desarrollados por los usuarios. Un IDE está conformado por un compilador, depurador y un editor gráfico, el cual ayuda a los programadores o desarrolladores a realizar sus programas de forma fácil, sencilla y programar diferentes lenguajes en el mismo IDE (Mackenzie & Sharkey, 2003)

El funcionamiento del IDE es dar soporte a la preparación de los programas, es decir actividades de codificación y pruebas.

El IDE tiene tareas importantes de la fase de codificación como las siguientes:

- Edición del código fuente.
- Proceso y ejecución de programas (Mackenzie & Sharkey, 2003).

Los entornos de programación sirven para que los programadores o desarrolladores realicen sus aplicaciones de forma ágil y óptima. Además, estas herramientas cuentan con una interfaz amigable y con una facilidad de realizar programas con sencillez y flexibilidad para usuarios que no cuentan con experiencia.

## 2.8. Lenguaje de programación

Es un lenguaje artificial que puede ser utilizado para definir una secuencia de instrucciones para su procesamiento. El lenguaje de programación se caracteriza por ser utilizado para crear aplicaciones y software de cualquier tipo (Corrales Mora, 1994).

Se puede usar para crear programas y que permitan la comunicación entre el usuario con la máquina. Existen tres tipos de lenguajes de programación:

- Lenguaje de máquina: es el lenguaje más primitivo, depende directamente del hardware, éste solo puede utilizar dos únicos símbolos cero y uno, es por esto se lo denomina lenguaje binario. Utiliza nemónicos como instrucciones.
- Lenguaje de bajo nivel: necesita ser traducido al lenguaje de máquina y sigue siendo dependiente.
- Lenguaje de alto nivel: tiempo de programación relativamente corto, es independiente de la máquina (Villalobos, 2014).

Como se muestra en la Figura 3, existen varios lenguajes de programación que nos ayuda a crear proyectos.

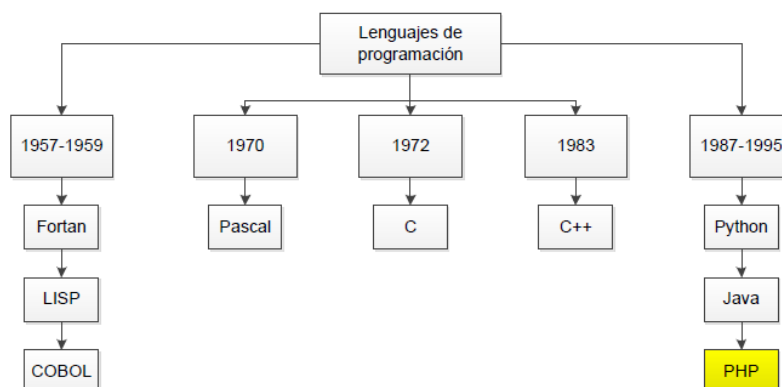


Figura 3. Lenguajes de programación.

PHP desde su creación es estimado como un lenguaje libre, es utilizado para el desarrollo de aplicaciones en el lado del servidor, son capaces de general contenido en la Word Wide Web. Es uno de los primeros lenguajes para la inserción en documentos HTML. Es posible la instalación de este lenguaje en cualquier tipo de sistema operativo de forma gratuita (Arias, 2017).

PHP es el lenguaje de programación más utilizado en todo el mundo para el desarrollo de páginas web, además PHP utilizan páginas estáticas y dinámicas (Torres Remon, 2014).

## 2.9. Base de datos

Es un conjunto de datos que se almacenan en un sistema computarizado, es decir, es una especie de armario electrónico para archivar datos en donde los usuarios pueden realizar diferentes funciones sobre los archivos como: visualizar, insertar, actualizar y borrar datos (Date, 2001).

Hoy en día las bases de datos sirven para guardar gran cantidad de información en servidores o computadoras que acumulan datos para diferentes empresas como: hospitales, gobierno, escuelas, comercio, etc.

Como se puede observar en la Figura 4:

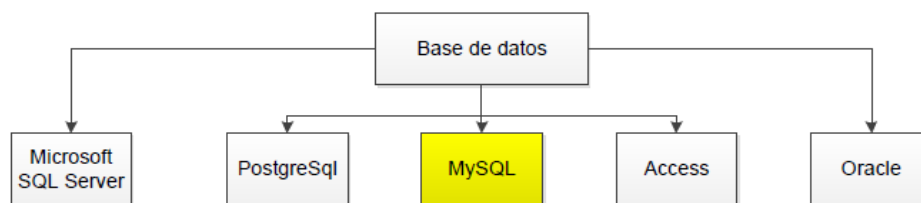


Figura 4. Tipos de base de datos.

MySql es una base de datos que conoció en la década de 1990 como un sistema de base de datos relacional, es muy adecuado para pequeños sistemas web, se lo valora mucho por ser gratuito, rápido y escalable (Combaudon, 2018).

Es un software libre, el cual permite la administración de base de datos de lo más simple a lo más complejo. Puede ser instalada en Windows y Linux (Thibaud, 2006).

### **2.10. Servidor web**

Es un programa que gestiona las aplicaciones en el lado del servidor realizando conexiones con el cliente generando una respuesta. El código recibido por el cliente es compilado y ejecutado por el navegador web. Además, para la transferencia de datos se utiliza el protocolo HTTP.

Los servidores web se encargan de administrar cada tipo de peticiones, de esta manera transmite la respuesta al cliente. El cliente y el servidor web se comunican a través de una conexión TCP, el cual se lo conoce como un sistema de petición/respuesta o cliente/servidor (EcuRed, 2019).

Los servidores web sirven para almacenar contenido de Internet y facilitar su disponibilidad constante y segura, así mismo son utilizados para la distribución de contenido web en redes internas o en el Internet (Digital Guide, 2019).

### **2.11. Diagrama software**

Se realiza un diagrama de bloques sobre el funcionamiento de la parte de software del proyecto, se visualiza la interacción entre el prototipo, base de datos y servidor web. En la Figura 5, se observa el ingreso de la información enviada desde el prototipo hacia la base de datos, en la cual se relaciona constantemente con la página web y de esta manera se grafican las rutas realizadas.

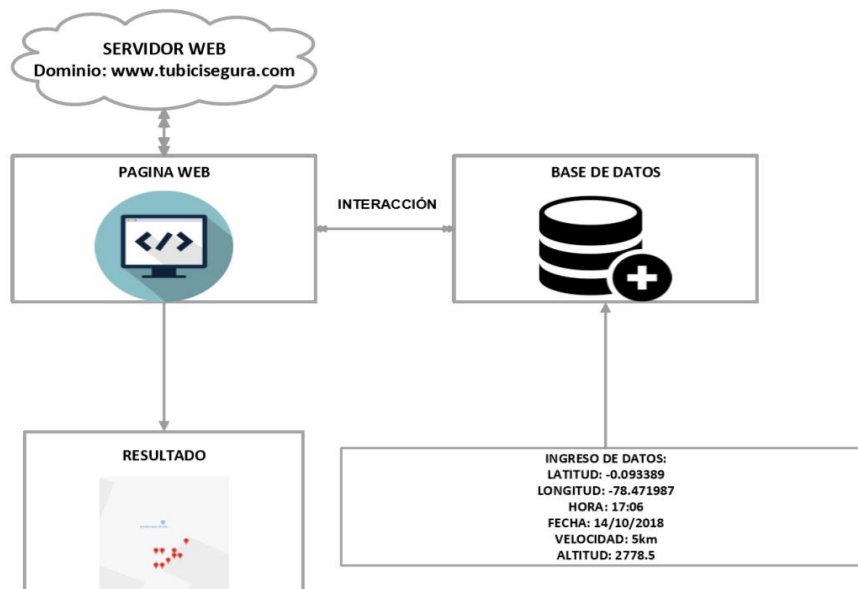


Figura 5. Diagrama de bloques de diseño de software.

## 2.12. Diagrama hardware

En este trabajo de titulación cuenta con una segunda parte, que es el hardware. En la Figura 6, se muestra un diagrama de bloques con los componentes del prototipo, estos están interactuando con el microcontrolador para que de esta manera los datos obtenidos se envíen de manera adecuada al aplicativo web.

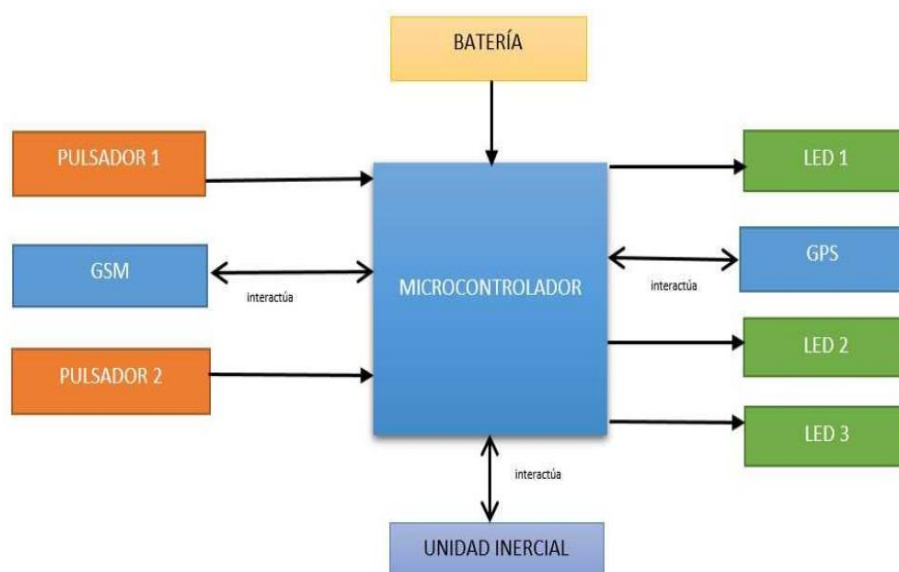


Figura 6. Diagrama de bloques de diseño de hardware.

### **3. Diseño**

En este capítulo se presenta el proceso de diseño del sistema de monitoreo, tomando en cuenta que está conformado de dos partes: diseño de hardware y diseño de software. En la primera, se especifican los componentes del prototipo, con las características que van a utilizarse en el sistema. Además, se compara entre diferentes dispositivos para elegir la mejor alternativa desde el punto de vista técnico y económico. La segunda parte, abarca todo lo referente a la página web, servidor web y base de datos, explicando la interacción entre las mismas. También se realizan comparaciones de acuerdo con sus características para escoger la mejor opción de acuerdo con el prototipo.

#### **3.1. Diseño hardware**

El prototipo está conformado por los componentes que se muestran en la Figura 6 del Capítulo II. En primer lugar, tenemos la parte principal del prototipo que es el módulo GPS, este proporciona la ubicación en tiempo real, velocidad y altura del ciclista. Se cuenta con un sensor que permite conocer cuando la bicicleta ha sufrido una caída y se notifica al usuario registrado mediante un correo electrónico, la interacción mediante el prototipo y la página es posible gracias al módulo GSM que permite enviar la información tanto por datos móviles y por mensajes de texto. Finalmente, todo el sistema será alimentado por una batería, que será dimensionada de acuerdo con la cantidad de amperaje y el voltaje necesario para el funcionamiento del circuito.

Los módulos que componen el prototipo interactúan entre sí mediante instrucciones que se programan de acuerdo con los objetivos. En el desarrollo del prototipo es importante conocer los requerimientos, para que de esta manera se elija los componentes adecuados. En la Tabla 1, se muestra una lista de requerimientos con respecto a los puertos necesarios entre la placa base y los módulos.



Tabla 1.

*Tabla de requerimientos para el microcontrolador.*

| <b>NOMBRE</b>  | <b>CANTIDAD<br/>PUERTOS<br/>(NÚMERO)</b> | <b>JUSTIFICACIÓN</b>    |
|----------------|--|-------------------------|
| UART           | 2  | Conexión GPS,<br>SIM900 |
| I/O            | 4  | Leds y pulsadores       |
| I2C            | 1  | Giroscopio              |
| SPI            | 1  | Micro SD                |
| Interrupciones | 2  | Acelerómetro            |
|                | <b>MEMORIA (KB)</b>                      |                         |
| Flash          | > 32                                     | Datos del GPS           |
| RAM            | > 2                                      | Datos del GPS           |

Los microcontroladores están embebidos en diferentes placas de las cuales se escogió tres tipos para realizar una comparación de características relevantes para el proyecto, como se muestra en la Tabla 2.

Tabla 2.

*Tabla comparativa microcontroladores.*

| <b><u>CARACTERÍSTICA/<br/>PLACA</u></b> | <b>ARDUINO<br/>(ARDUINO<br/>MEGA)</b> | <b>RASPEBRR<br/>Y PI 3</b>     | <b>STM<br/>(STM32F4Disc<br/>overy)</b> |
|---|---------------------------------------|--------------------------------|--|
| PROCESADOR/MICROCONTROLADOR             | ATmega2560                            | Chipset<br>Broadcom<br>BCM2387 | STM32F407VG<br>T6                      |
| VOLTAJE DE<br>FUNCIONAMIENTO            | 5 V                                   | 5 V                            | 5V                                     |
| VOLTAJE DE<br>ENTRADA                   | 7-12 V                                | 7-12 V                         | 3.3-12 V                               |
| VELOCIDAD DEL CPU                       | 16 MHz                                | 1.2 GHz                        | 72 MHz                                 |
| I/O DIGITAL                             | 54                                    | 27                             | 26                                     |
| SRAM                                    | 8 Kb                                  | 1 Gb                           | 192Kb                                  |
| FLASH                                   | 256 Kb                                | microSD                        | 1 Mb                                   |
| UART                                    | 4                                     | 2                              | 2                                      |

|  |                                  |                                 |                               |
|--|----------------------------------|---------------------------------|-------------------------------|
| I2C  | 1                                | 1                               | 2                             |
| SPI  | 1                                | 1                               | 2                             |
| COSTO  | \$ 20                            | \$ 50                           | \$32                          |
| LENGUAJE DE PROGRAMACIÓN / SISTEMA OPERATIVO | Arduino, C, C++ /Multiplataforma | Linux, Windows /Multiplataforma | Arduino, C++ /Multiplataforma |
| INTERRUPCIONES                               | 6                                | -                               | En todos los puertos I/O      |

Tabla 3.

Análisis de requerimientos en contraste a productos disponibles.

| REQUERIMIENTO/PLACA | ARDUINO | RASPBERRY<br>3 | STM<br>(STM32Discovery) |
|---------------------|---------|----------------|-------------------------|
| UART                | •       | •              | •                       |
| I/O                 | •       | •              | •                       |
| I2C                 | •       | •              | •                       |
| SPI                 | •       | •              | •                       |
| INTERRUPCIONES      | •       |                | •                       |
| Flash > 32 Kb       | •       |                | •                       |
| RAM > 2KB           | •       | •              | •                       |

Adaptado de (Arduino, 2018).

Con los datos de la Tabla 1. Se eligió el circuito base un microcontrolador, que es un circuito digital integrado programable que permite la ejecución de varios programas para realizar diferentes acciones. Además, cuenta con puertos de salida y entrada para el control de periféricos y memorias para el almacenamiento de la información.

En la Tabla 3. Se realiza un análisis entre los requerimientos y las placas para identificar las características que cumple cada una de las mismas.

Las placas Raspberry Pi y Stm32Discovery tienen propiedades necesarias para el prototipo, pero cuentan con características extras que llevaría a una pérdida

de recursos como Raspberry Pi 3 que es una minicomputadora que tiene ethernet, Wifi y STM32Discovery desarrolla aplicaciones de audio, conectividad ethernet, control de pantalla LCD. Además, en una comparación a nivel económico, estas placas tienen un costo elevado.

Para este prototipo se escogió la placa Arduino Mega, ya que tiene el número de pines necesarios para los diferentes módulos, para el almacenamiento de programas cuenta con 256 Kb de memoria flash que permite almacenar los programas y 8 Kb de SRAM que manipula las variables a ejecutarse. También, Arduino cuenta con características como:

- Precio adecuado, debido a que, así como existen placas originales, existen placas genéricas que su funcionalidad no varía y permite realizar las pruebas necesarias antes de realizar el prototipo final.
- Cuenta con gran cantidad de documentación, Arduino tiene programas open source e información en internet que permiten programar lo que se desee ya que se puede encontrar varios ejemplos relacionados con lo que se necesita realizar y cambiar de acuerdos a los objetivos de cada proyecto.

### **3.1.1. GSM**

El sistema tiene el objetivo de enviar la información hacia una base de datos por lo que es necesario buscar un módulo que permita realizar este proceso de manera inalámbrica debido a que se necesita los datos en tiempo real y enviar un correo electrónico ante una posible caída del ciclista, se necesita una lista de requerimientos para elegir el mejor módulo, se realiza la misma en la siguiente tabla.

Tabla 4.

*Requerimiento del GSM.*

| NECESIDAD | DESCRIPCIÓN                            |
|-----------|--|
| UART      | Conectividad con el sistema de control |
| GPRS      | Conexión de datos inalámbrica          |
| 5V        | Voltaje                                |

Existe algunos módulos de comunicación que permiten el envío de información y con los requerimientos establecidos, se realiza una tabla comparativa entre tres módulos de comunicación como se muestra en la siguiente tabla.

Tabla 5.

*Tabla relación característica módulo.*

| CARACTERÍSTICA/<br>MÓDULO       | GSM (SIM<br>900)       | ARDUINO<br>ETHERNET<br>SHIELD | MÓDULO<br>WIFI<br>(ESP8266) |
|---------------------------------|------------------------|-------------------------------|-----------------------------|
| PROCESADOR/<br>MICROCONTROLADOR | SIM 900                | Wiznet<br>W5100               | ESP8266MO<br>D              |
| VOLTAJE DE<br>FUNCIONAMIENTO    | 3.4 a 5 V              | 3.3 a 5 V                     | 3.0 a 3.6 V                 |
| BANDA DE<br>FRECUENCIA          | 850 a 1900<br>MHz      |                               | 2.4 GHz                     |
| VELOCIDAD DE<br>TRANSMISIÓN     | Configurable           | 10/100Mbps                    | Máximo 60<br>Mbps           |
| CONSUMO DE<br>CORRIENTE         | 15 mA                  | 50 mA                         | < 10uA                      |
| UART                            | 2                      | 1                             | 1                           |
| COSTO                           | \$ 20                  | \$ 50                         | \$5                         |
| USO SIM                         | SI                     | NO                            | NO                          |
| ALCANCE                         | Cobertura red<br>móvil | Cable de red                  | Máximo 50<br>metros         |

Tabla 6.

*Tabla de análisis requerimiento módulo.*

| REQUERIMIENTO /MÓDULO | GSM (SIM 900) | ARDUINO ETHERNET SHIELD | MÓDULO WIFI (ESP8266) |
|-----------------------|---------------|-------------------------|-----------------------|
| UART                  | •             | •                       | •                     |
| GPRS                  | •             |                         |                       |
| 5V                    | •             | •                       | •                     |

Como módulo de comunicación se elige GSM ya que proporciona al prototipo independencia para el envío de información porque posee una tarjeta SIM que permite al prototipo enviar información por datos móviles y por las características enumeradas en la Tabla 5.

Arduino Ethernet Shield logra la conectividad a internet sin un sistema operativo intermediario, se conecta a la red mediante un cable RJ45 y el módulo Wifi ESP8266 incluye TCP/IP. Los dos tipos de placas mencionadas anteriormente no se escogieron porque la conexión del primero no es inalámbrica y el segundo no permite enviar los datos en tiempo real (Arduino, 2018).

Según el análisis realizado en la Tabla 6. Se elige el GSM SIM 900, se comunica con el microcontrolador mediante puertos UART, incluye TCP/IP que permite la conexión a internet mediante programación en sockets con comandos AT, es compatible con la mayoría de las operadoras porque soporta 4 bandas internacionales de GSM.

### 3.1.2. GPS

El sistema de control necesita un módulo GPS que es uno de los principales, para obtener datos de ubicación, velocidad y altura. Se realiza una lista de requerimientos en la siguiente tabla.

Tabla 7.

*Tabla requerimientos GPS.*

| NECESIDAD | JUSTIFICACIÓN                  |
|-----------|--------------------------------|
| UART      | Conexión al sistema de control |

|           |               |
|-----------|---------------|
| HASTA 5V  | Voltaje       |
| PRECISIÓN | Datos exactos |

En el mercado existen algunos tipos de módulos GPS, se realiza una comparación en la siguiente tabla.

Tabla 8.

*Tabla relación característica módulo.*

| CARACTERÍSTICA/<br>MÓDULO    | NEO 6M      | NEO 6G       | NEO 6Q      |
|------------------------------|-------------|--------------|-------------|
| VOLTAJE DE<br>FUNCIONAMIENTO | 2.7 a 3.6 V | 1.75 a 2.0 V | 2.7 a 3.6 V |
| FRECUENCIA                   | 5Hz         | 5Hz          | 5Hz         |
| CONSUMO DE<br>CORRIENTE      | 15 mA       | 50 mA        | < 10uA      |
| UART                         | SI          | SI           | SI          |
| SPI                          | SI          | SI           | SI          |
| COSTO                        | \$ 15       | -            | -           |
| DISPONIBILIDAD<br>MERCADO    | SI          | NO           | NO          |

Tabla 9.

*Tabla requerimiento módulo.*

| REQUERIMIENTO<br>/MÓDULO | NEO 6M | NEO 6G | NEO 6Q |
|--------------------------|--------|--------|--------|
| UART                     | •      | •      | •      |
| HASTA 5V                 | •      | •      | •      |

Los módulos GPS NEO-6 son una familia de receptores GPS que permite utilizarse en sistemas que son de tamaño reducido, encuentran satélites al instante y el costo es reducido. Usa el algoritmo PPP que ayuda a la precisión para las aplicaciones que ser requiera como agricultura, cartografía, marina, entre otros.

Según el análisis realizado en la Tabla 8. Se escoge el módulo NEO 6M ya que se encuentra disponible en el mercado y además es de costo bajo.

### 3.1.3. Sensor MPU6050

El sistema tiene un sensor que permite conocer cuando la bicicleta ha sufrido una caída mediante un mensaje de texto, para este fin se necesita algunos requerimientos que se presentan en la siguiente tabla.

Tabla 10.

Tabla requerimiento sensor.

| NECESIDAD | JUSTIFICACIÓN                  |
|-----------|--------------------------------|
| SPI       | Conexión al sistema de control |
| HASTA 5V  | Voltaje                        |
| PRECISIÓN | Datos exactos                  |

Para conocer el cambio de posición de la bicicleta se realiza la comparación de varios tipos de sensores, que miden el cambio de posición de un elemento, se presenta en la siguiente tabla.

Tabla 11.

Tabla relación módulo característica.

| CARACTERISTICA/MÓDULO        | ACELERÓMETR<br>O<br>ADXL345 | UNIDAD<br>INERCIAL<br>MPU6050 |
|------------------------------|-----------------------------|-------------------------------|
| VOLTAJE DE<br>FUNCIONAMIENTO | 3.6 V                       | 3.5 V                         |
| FRECUENCIA                   | 5Hz                         | 5Hz                           |
| CONSUMO DE CORRIENTE         | 45 uA                       | 3.6 mA                        |
| I2C                          | SI                          | SI                            |
| SPI                          | SI                          | NO                            |
| RANGO DE MEDICIÓN            | 2g/4g/8g/16g                | 2g/4g/8g/16g                  |
| COSTO                        | \$ 10                       | \$4                           |
| DISPONIBILIDAD MERCADO       | SI                          | SI                            |

Tabla 12.

*Tabla relación requerimiento módulo.*

| REQUERIMIENTO /MÓDULO | ACELERÓMETRO ADXL345 | UNIDAD INERCIAL MPU6050 |
|-----------------------|----------------------|-------------------------|
| SPI                   | •                    | •                       |
| HASTA 5V              | •                    | •                       |
| PRECISIÓN             | •                    | •                       |

Un acelerómetro permite medir la aceleración gravitatoria, respondiendo a las vibraciones asociadas cuando se produce un movimiento, mientras que un giroscopio permite medir una la variación de una posición mediante la velocidad angular.

Se necesita usar una unidad inercial, que es un sensor que tiene integrado un acelerómetro y un giroscopio este permite tener mayor flexibilidad al momento de programar ya que se puede hacer uso de las características de dos sensores, se escoge MPU6050 porque el costo es bajo y existe amplia documentación que permite realizar la programación del dispositivo para mediar la caída de la bicicleta y enviar una alerta.

### 3.1.4 Batería

La alimentación del circuito se lo realizará mediante una batería para que este sea autónomo. Es necesario realizar un análisis de la cantidad de amperios que se consume en total para encontrar la batería adecuada, se realiza en la siguiente tabla.

Tabla 13.

*Amperaje total del prototipo*

| DISPOSITIVO/ AMPERAJE | mah  |
|-----------------------|------|
| ARDUINO               | 50   |
| GSM                   | <590 |



|                 |     |
|-----------------|-----|
| GPS             | 15  |
| UNIDAD INERCIAL | 3.6 |

El consumo total del circuito es de 658.6 mah, por lo que se puede alimentar mediante una batería externa con estas características.

### 3.2. Implementación del prototipo

El sistema de control tiene como base un microcontrolador, los pines a utilizarse se muestran en la Figura 7.

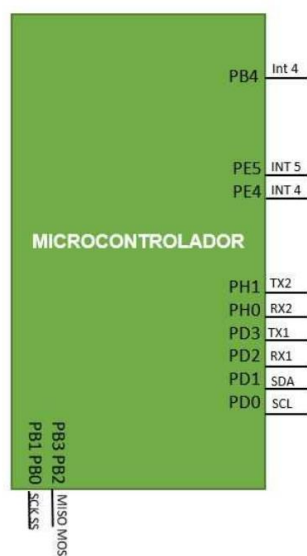


Figura 7. Pines utilizados del microcontrolador.

En la siguiente tabla. Se muestra cómo se conecta los pines de cada módulo al microcontrolador

Tabla 14.

Tabla conexión pines.

|            | MICROCONTROLADOR |          |
|------------|------------------|----------|
| <b>GPS</b> |                  |          |
| RX1        | TX1 (pin 18)     | Serial 1 |
| TX1        | Rx1 (pin 19)     |          |
| <b>GSM</b> |                  |          |
| RX2        | Tx2 (pin 16)     | Serial 2 |
| TX2        | Rx2(pin 17)      |          |

| <b>MICROSD</b>         |                      |             |
|------------------------|----------------------|-------------|
| SS                     | pin 53               | SPI         |
| SCK                    | pin 52               |             |
| MOSI                   | pin 51               |             |
| MISO                   | pin 50               |             |
| PULSADOR 1             | interrupción (pin 2) | inicio ruta |
| PULSADOR 2             | interrupción (pin 3) | fin ruta    |
| LED 1                  | pin 10               |             |
| LED 2                  | pin 11               |             |
| LED 3                  | pin 12               |             |
| <b>UNIDAD INERCIAL</b> |                      |             |
| SCL                    | pin 21               | I2C         |
| SDA                    | pin 20               |             |

El módulo GSM SIM 900 permite que la información que ingresa al Arduino pueda ser enviada hacia la base de datos mediante datos móviles, para lograr esto se programa el GSM mediante comandos AT.

Los comandos AT son una serie de instrucciones que permiten que el GSM interactúe con el Arduino para enviar la información mediante la red móvil.

Se muestra una lista cada comando con su funcionalidad a continuación:

- AT: Comprueba el estado del módulo
- AT+CPIN="XXXX": Introduce el pin del SIM, si este tuviera
- AT+CREG?: Comprueba la conexión de red
- AT+CGATT?: Conecta a la red GPRS
- AT+CIPSHUT: Cierra el contexto PDP del GPRS
- AT+CSTT="internet.movistar.com.ec": Permite configurar la APN, en este caso movistar que es el chip que estamos usando
- AT+CIICR: Activa la conexión inalámbrica
- AT+CIFSR: Obtiene la IP local

Además, en el Tabla 13, se observa que el prototipo cuenta con 3 leds, cada uno cumple con una función específica. Como se indica en la Figura 8, el led rojo realiza una validación de conexión de los componentes utilizados, cuando los servicios estén completamente activos, alertando al usuario que puede realizar la ruta. El led naranja se enciende mediante un pulsador para conocer cuando

se inicia la ruta y otro pulsador termina la ruta. Finalmente, el led verde mediante titilación ayuda a conocer el envío de datos del GSM al aplicativo web.

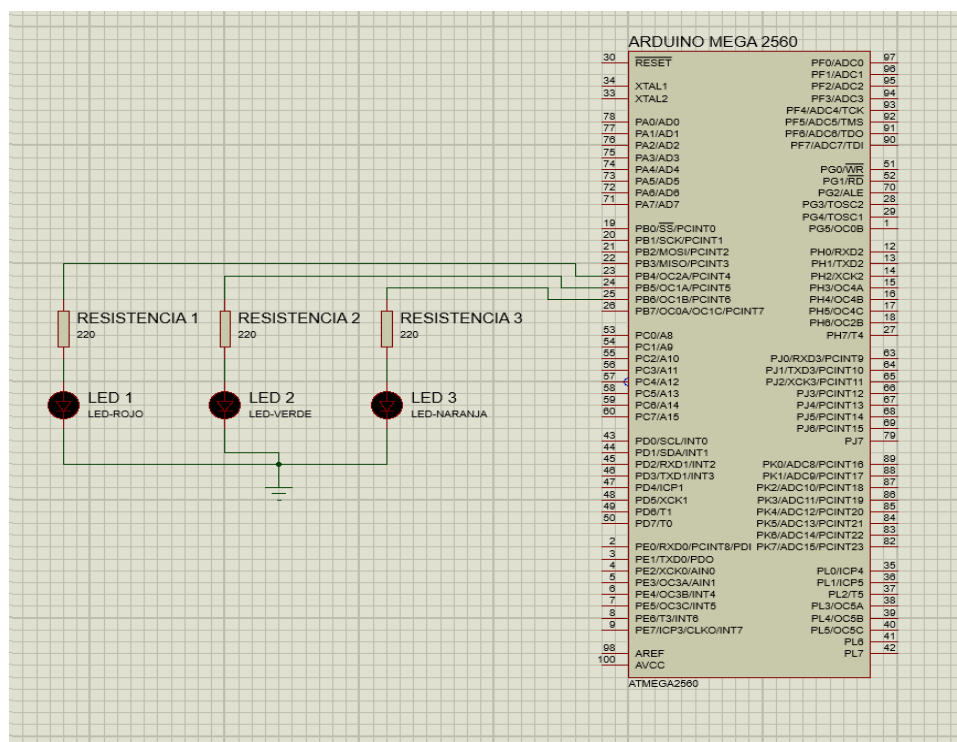
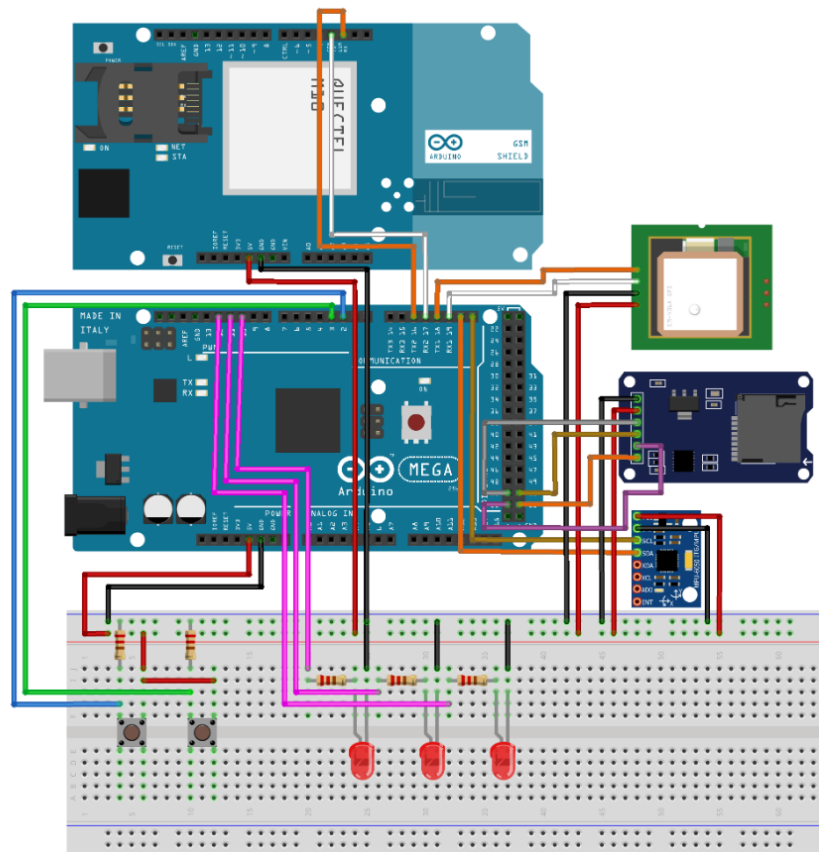


Figura 8. Diagrama de leds utilizados.

En la Figura 9, se visualiza un bosquejo con los elementos reales de cómo está desarrollado el prototipo.



*Figura 9.* Conexión componentes prototipo.

En el Anexo 3 y 4 se muestra el diseño de la placa realizado en Proteus donde podemos visualizar la integración de los módulos, el anexo 5 muestra una imagen con el prototipo, en el anexo 6 se puede visualizar una vista del diseño 3D lateral y en el anexo 7 una vista superior en 3D.

### 3.3. Diseño de software prototipo

**3.3.1. GMS:** el prototipo necesita una conexión a internet, esto se realiza con los comandos AT que son instrucciones codificadas, también es necesario tener un APN que en este caso se utiliza CNT porque es una operadora que no utiliza un usuario y contraseña para la configuración y, además, su cobertura es buena. En la Figura10, se indica el código utilizado para realizar la conexión. La primera configuración que se realiza es el GPRS que nos ayuda a conectarnos a internet y se establece la APN de la operadora que se va a utilizar.

```

Serial.println(F("Conectado a la red.));
enviarAT("AT+CGATT?", (char*)"OK", 1000); //Iniciamos la conexión GPRS
enviarAT("AT+SAPBR=3,1,\"CONTYPE\", \"GPRS\", (char*)"OK", 3000);
enviarAT("AT+CSTT=\"internet.cnt.net.ec\", \"\", \"\", (char*)"OK", 3000);
enviarAT("AT+SAPBR=1,1", (char*)"OK", 3000);

```

Figura 10. Código conexión GSM.

Finalmente, en la en la Figura 11, se inicia la conexión TCP, para enviar los datos del prototipo hacia al aplicativo web (*“write\_data.php”*).

```

enviarAT("AT+HTTTPARA=\"URL\", \"http://tubicisegura.com/write_data.php?\" + infor + \"\", (char*)"CONNECT OK", 3000);

```

Figura 11. Código de enlace prototipo página web.

### 3.3.2. GPS

Para la realización del código del GPS, es necesario dividir los campos que manda el sensor, ya que si no se realiza este proceso el GPS envía de forma unida y desordenada. Además, es importante tener en cuenta que la hora que se obtiene es de la línea equinoccial, por esta razón se debe realizar la conversión necesaria para obtener la hora correcta.

### 3.3.3. MicroSD

La microSD es importante porque se realiza un *“backup”* en caso de la pérdida de la señal del GSM. En la Figura 12, se puede observar la manera en la que trabaja este elemento. Así mismo, estos datos se almacenan en un archivo *“.txt”* y guardan el último número de la ruta realizada.

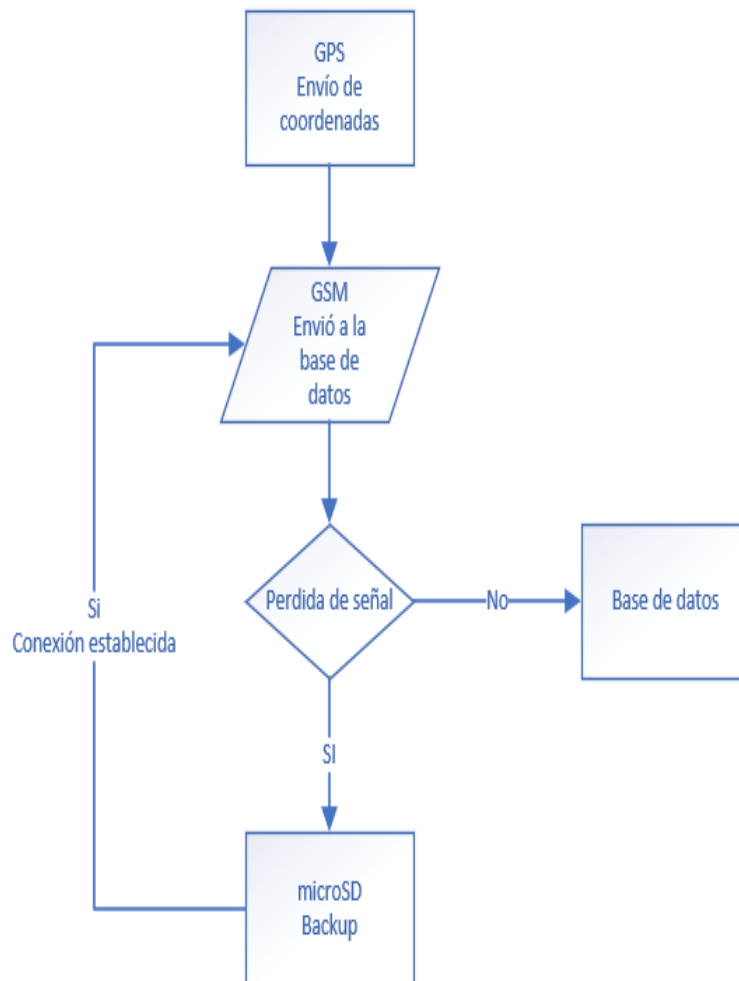


Figura 12. Diagrama de flujo backup microSD.

### 3.3.4. Unidad Inercial

Este sensor permite mediante la velocidad angular conocer la variación de la posición del ciclista que puede ser un accidente sufrido por el mismo. En la Figura 13, se visualiza un diagrama de flujo en caso de un accidente que tenga el ciclista.

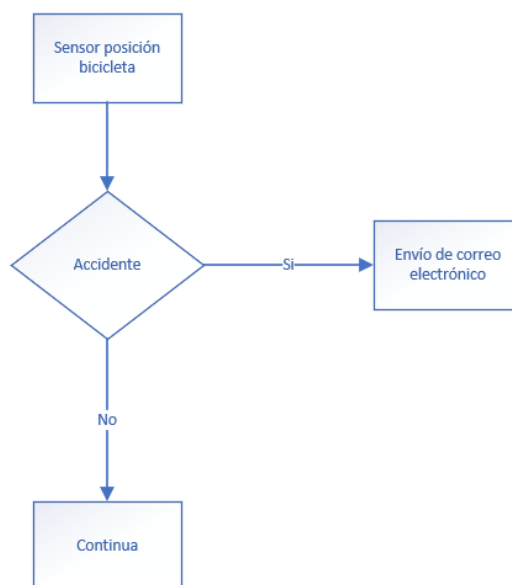


Figura 13. Diagrama de flujo unidad inercial.

### 3.4. Diseño de software página web

El diseño de software es una parte fundamental para el proyecto, ya que éste es la interacción entre los ciclistas y la página web. Cada usuario puede acceder a su perfil a través de un “login”, en el cual se puede observar mediante un mapa las rutas y visualizar estadísticas de velocidad o altitud. Toda la información se encuentra almacenada en una base de datos, lo que permite a la página web realizar consultas y mostrar a los usuarios los resultados en un tiempo determinado.

Para la elaboración del prototipo se requieren los siguientes servicios:

Tabla 15.

Tabla relación ítem descripción.

| ÍTEM       | DESCRIPCIÓN  |
|------------|--|
| HOSTING    | Espacio en donde se alojan todos los servicios necesarios. |
| SERVIDORES | Cuenta con un servidor web y un servidor de base de datos. |
| IDE        | Plataforma en donde se ejecutan los programas.             |

|                          |   |
|--------------------------|---|
| LENGUAJE DE PROGRAMACIÓN | Ayuda a la creación del diseño de la página con métodos o clases. |
| BASE DE DATOS            | Sirve para almacenar la información del aplicativo.               |

### 3.4.1. Hosting

Para la implementación del prototipo se debe adquirir un servicio de hosting que es un espacio de alojamiento para almacenar los datos de la página web, para que de esta manera siempre ésta esté online y los usuarios puedan acceder en cualquier momento a la página web. En la Tabla 15, están los requerimientos necesarios para la implementación del aplicativo web.

Tabla 16.

*Tabla de requerimientos de hosting.*

| SERVICIOS              | Justificación                   |
|------------------------|---------------------------------|
| DOMINIO                | www.tubicisegura.com            |
| DISCO DURO             | 1 GB                            |
| MEMORIA                | 1GB                             |
| SERVIDOR WEB           | Almacenar la página web         |
| SERVIDOR BASE DE DATOS | Base de datos                   |
| CUENTA DE CORREO       | 2 cuentas de correo electrónico |

Existen varias empresas que ofrecen este tipo de servicio con diferentes características, funciones y costos. Como se muestra en la Tabla 16, la comparación de varias entidades que presentan este servicio de hosting:

Tabla 17.

*Tabla comparativa de servicios de hosting.*

|                        | NODO RED | GODADDY | WEBEMPRESA |
|------------------------|----------|---------|------------|
| HOSTING<br>[ANUAL] USD | \$25     | \$95.88 | \$79       |
| DOMINO                 | \$12.99  | \$14.99 | NO         |
| DISCO DURO             | 2 GB     | 10 GB   | 5GB        |



|                   |            |           |           |
|-------------------|------------|-----------|-----------|
| [ALMACENAMIENTO]  |            |           |           |
| BANDA ANCHA       | 20 GB      | Ilimitada | 200 GB    |
| CUENTAS DE CORREO | 20 cuentas | 5 cuentas | Ilimitada |

Además, es muy importante tener en cuenta que para la implementación hay que comprar un dominio que es un nombre único que identifica una página web en Internet, para este caso se adquirió un dominio que se llama [www.tubicisegura.com](http://www.tubicisegura.com), con éste se podrá ingresar a la página web y realizar las funciones que se necesiten.

El hosting contiene dos servidores: servidor web que es el lugar en donde se aloja la página web y un servidor de base de datos que es el archivo electrónico del aplicativo, es aquí donde se almacenan todos los datos que ingresan.

Tabla 18.

*Tabla relación de características de servicios.*

| SERVICIOS         | NODORED | GODADDY | WEBEMPRESA |
|-------------------|---------|---------|------------|
| DOMINIO           | •       | •       |            |
| ALMACENAMIENTO    | •       | •       | •          |
| MEMORIA           | •       | •       | •          |
| CUENTAS DE CORREO | •       | •       | •          |
| BANDA ANCHA       | •       | •       | •          |
| SERVIDORES        | •       | •       | •          |

Finalmente con la Tabla 17, que presenta un checklist de los servicios, se puede visualizar que las empresas NodoRed y GoDaddy ya que tienen los mismos servicios, pero se debe tener en cuenta los requerimientos que se mencionaron con anterioridad y es por esa razón que seleccionamos la empresa NodoRed, ya que en su reducido costo ofrece las características necesarias para la implementación del prototipo y así mismo, al tener una interfaz sencilla las prestaciones del hosting cubren las necesidades de los requerimientos.

### 3.4.2. IDE (Entorno de desarrollo integrado)

Para el desarrollo de la página web es importante algunos requisitos para la implementación del aplicativo. En la Tabla 18, se muestran los servicios necesarios que debe ofrecer el IDE, este debe ser compatible con diferentes lenguajes de programación.

Tabla 19.

*Tabla relación requisitos justificación.*

| REQUISITOS      | JUSTIFICACIÓN   |
|-----------------|---|
| PHP             | Que sea compatible con la versión PHP 7.0             |
| JQUERY          | Compatible con este elemento para animaciones y mapa. |
| HTML            | Realiza el diseño de la página web                    |
| MULTIPLATAFORMA | Sea compatible con cualquier sistema operativo        |

Como se observa en la Tabla 19, existen varias plataformas IDE que ayudan a realizar la programación de forma ágil y sencilla, además se muestran las ventajas y desventajas que tienen los IDE.

Tabla 20.

*Tablas características de IDE.*

| IDE      | VENTAJAS                           | DESVENTAJAS                       |
|----------|------------------------------------|-----------------------------------|
| NETBEANS | Instalación y actualización simple | No posee un editor HTML           |
|          | Posee soporte para PHP             | Lentitud                          |
|          | Multiplataforma                    | Requiere un interprete            |
|          | Múltiples lenguajes                | Algunas herramientas tienen costo |
| ECLIPSE  | Compilación en tiempo real         | Consumo de recursos               |
|          | Editor de texto                    | Lentitud                          |
|          | Plataforma ligera                  | Carece de soporte para webapps    |
|          | Gratuito                           | Plugins no potentes ni sencillos  |

|               |   |   |
|---------------|---|---|
| VISUAL STUDIO | Aprendizaje rápido                        | No soporta programación orientada a objetos |
|               | Librerías DDL y componentes ActiveX       | Ejecutables solo para Windows               |
|               | Orientado a capas                         | Es de pago                                  |
|               | Adecuado para realizar prototipos rápidos | Ejecutables demasiado lentos                |

En la Tabla 20, se realiza un checklist de los requerimientos necesarios para la implementación del aplicativo web.

Adaptado de (Fernández, 2018).

Tabla 21.

*Tabla relación características IDE.*

|                 | NetBeans | Eclipse | Visual Studio |
|-----------------|----------|---------|---------------|
| PHP             | •        | •       |               |
| JQuery          | •        | •       |               |
| HTML5           | •        | •       |               |
| Multiplataforma | •        | •       | •             |

En la Tabla 20, se puede visualizar que NetBeans y Eclipse cumplen con los requerimientos necesarios para el prototipo. Pero para realizar la página web se escogió NetBeans porque es una plataforma gratuita, ayuda a gestionar proyectos grandes, cuenta con una gran variedad de librerías que son encontradas de manera fácil y contiene acceso a con diferentes bases de datos, además, este IDE contiene una interfaz amigable en donde el usuario puede realizar los programas sin tener mucha experiencia.

### 3.4.3. Lenguaje de programación

Para realizar la aplicación web se debe tener algunos aspectos en cuando a los lenguajes de programación, es necesario saber que cuando se contrata un servicio de hosting, este debe ser compatible con algunos lenguajes de programación y es por esta razón que se hace un análisis de requerimientos.

Tabla 22.

*Tabla relación requisitos justificación.*

| REQUISITOS                       | JUSTIFICACIÓN                               |
|----------------------------------|---|
| PROGRAMACIÓN ORIENTADA A OBJETOS | Sirve para tener una buena estructura       |
| SERVIDOR WEB                     | Compatible con el servidor web              |
| BASE DE DATOS                    | Compatible con el servidor de base de datos |
| MULTIPLATAFORMA                  | Compatible con diferentes IDE               |

En la Tabla 22, se muestran las ventajas y desventajas de los lenguajes de programación.

Tabla 23.

*Tabla de características de lenguajes de programación.*

| LENGUAJE PROGRAMACIÓN | VENTAJAS                          | DESVENTAJAS                         |
|-----------------------|-----------------------------------|-------------------------------------|
| PHP                   | Programación orientada a objetos  | Instalación en un servidor          |
|                       | Multiplataforma                   | Dificulta la organización por capas |
|                       | Base de datos MySql, PostgreSQL.  | Afecta la mezcla de PHP con HTML    |
|                       | Open source                       |                                     |
| ASP                   | Es un Framework de desarrollo     | No es multiplataforma               |
|                       | Ejecución en el lado del servidor | Costo elevado                       |
|                       | Base de datos SQL Server          | Alojamiento demasiado costos        |
|                       | Mayor seguridad                   | Demasiado consumo de recursos       |

|        |                                  |                              |
|--------|----------------------------------|------------------------------|
| Python | Programación orientada a objetos | Lenguaje relativamente lento |
|        | Multiplataforma                  | Programación web compleja    |
|        | Portable                         |                              |
|        | Código fuente abierto            |                              |

Con los datos de la Tabla 21, se realizó un checklist de los lenguajes de programación que existen en la actualidad y dependiendo de eso se realizará un análisis sobre estos.

Tabla 24.

*Tabla relación requerimiento lenguaje de programación.*

| REQUERIMIENTOS                   | PHP | ASP | PYTHON |
|----------------------------------|-----|-----|--------|
| PROGRAMACIÓN ORIENTADA A OBJETOS | •   |     | •      |
| SERVIDOR WEB                     | •   | •   | •      |
| BASE DE DATOS                    | •   | •   | •      |
| MULTIPLATAFORMA                  | •   |     | •      |

Finalmente, con la Tabla 23, se seleccionó el lenguaje de programación PHP, porque es un lenguaje que hoy en día varios usuarios utilizan ya que es un código abierto y puede ser instalado en cualquier sistema operativo de manera gratuita. Además, se utiliza HTML para el diseño de la página web, este lenguaje ayuda hacer dinámicos los elementos como texto, imágenes y videos, también interactúa con jQuery que sirve para manipular eventos y desarrollo de animaciones.

#### **3.4.4. Base de datos**

Para la implementación de la página web es necesario tener una base de datos sólida para que de esta manera pueda realizar consultas de forma ágil y veloz. Como se muestra en la Tabla 24, se visualiza los requerimientos necesarios para la construcción de la base de datos.

Tabla 25.

*Tabla relación requisitos justificación.*

| REQUISITOS                | JUSTIFICACIÓN   |
|---------------------------|---|
| PHP                       | Conexión con el lenguaje de programación y la base de datos |
| SERVIDOR DE BASE DE DATOS | Compatible el servidor que ofrece el servicio hosting       |
| MULTIPLATAFORMA           | Compatible con cualquier sistema operativo                  |
| TRANSACCIONES             | La velocidad con la que se realizan las operaciones         |

En la Tabla 25, contiene las ventajas y desventajas de las diferentes bases de datos que hoy en día son las más utilizadas.

Tabla 26.

*Tabla de características base de datos.*

| BASE DE DATOS | VENTAJAS                           | DESVENTAJAS                         |
|---------------|------------------------------------|-------------------------------------|
| MYSQL         | Multiplataforma                    | Capacidad limitada                  |
|               | Bajo costo                         | No es intuitivo con otros programas |
|               | Open source                        |                                     |
|               | Velocidad al realizar operaciones  |                                     |
| SQL SERVER    | Seguro                             | Utiliza demasiada RAM               |
|               | Estabilidad                        | Es de pago                          |
|               | Entorno gráfico                    | Utiliza muchos recursos             |
|               | Soporta procedimientos almacenados | Tiene restricciones                 |
| POSTGRESQL    | Multiplataforma                    | Lento                               |

|  |                                  |                             |
|--|----------------------------------|-----------------------------|
|  | Open source                      | Vulnerable                  |
|  | Gran capacidad de almacenamiento | No es intuitivo             |
|  | Gran escalabilidad               | Consume demasiados recursos |

Tabla 27.

*Tabla relación requerimiento base de datos.*

| REQUERIMIENTOS            | MYSQL | SQL SERVER | POSTGRESQL |
|---------------------------|-------|------------|------------|
| PHP                       | •     |            | •          |
| Servidor de base de datos | •     | •          | •          |
| Multipataforma            | •     |            | •          |
| Transacciones             | •     | •          | •          |

En la Tabla 26, para la realización del diseño de la base de datos, se tomó en cuenta la Tabla 25, en donde se observan las diferencias que existen, en este caso para la implementación se escogió MySQL, que es una base de datos relacional, sólida, flexibles y recomendada para ser utilizada en páginas web.

### 3.5. Implementación de la página web

Como primer paso la página web realiza una conexión con la base de datos, la cual utiliza una clase llamada "conexión.php", ésto sirve para que se pueda ejecutar sentencias en la base de datos.

La pantalla principal está dividida en diferentes secciones como se muestra en la Figura 14.

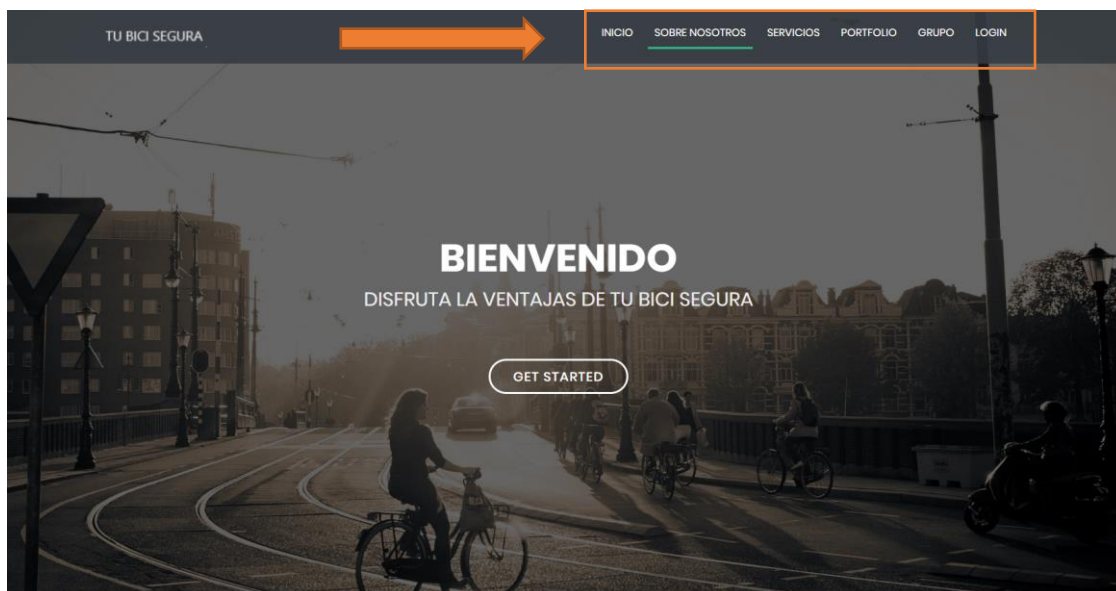


Figura 14. Página de inicio, aplicativo web.

Como se muestra en la Figura 15, la primera sección de la página web cuenta con una breve descripción sobre la información del prototipo y porque se realizó este sistema, además tiene una poca información sobre los servicios que contiene el aplicativo y el logo.

### Un poco sobre nosotras

Somos una empresa que ofrece servicios de GPS en tiempo real para bicicletas y además, cuenta un servicio de accidentes, que de esta manera se podrá informar a algún familiar sobre si se tuvo un accidente. Nuestro afán de la creación de este sistema fue la inseguridad que tiene un ciclista en la ciudad. Y es por ese motivo que se comenzó a diseñar un sistema de monitoreo que facilite al usuario ver su ubicación en tiempo real.

Rastreo y control de alertas las 24 horas y los 365 días del año.

Control de velocidad y altura

Panel del usuario (Monitoreo en tiempo)

Batería de respaldo



Figura 15. Sección de información del proyecto.

En la segunda sección como se puede visualizar en la Figura 16, contiene un portafolio de las imágenes del prototipo y como está diseñado el prototipo. Y en la última sección tenemos el mapa del local en donde se puede contratar los servicios de nuestro sistema de monitoreo en tiempo real para bicicletas.





Figura 16. Mapa de ubicación del local.

Además, cuenta con un botón de login en donde los administradores o ciclistas podrán ingresar a su respectivo perfil. El login como se muestra en la Figura 17, cuenta con un usuario único y una contraseña, si ésta es errónea mostrará un mensaje de error, caso contrario ingresará a la página siguiente.

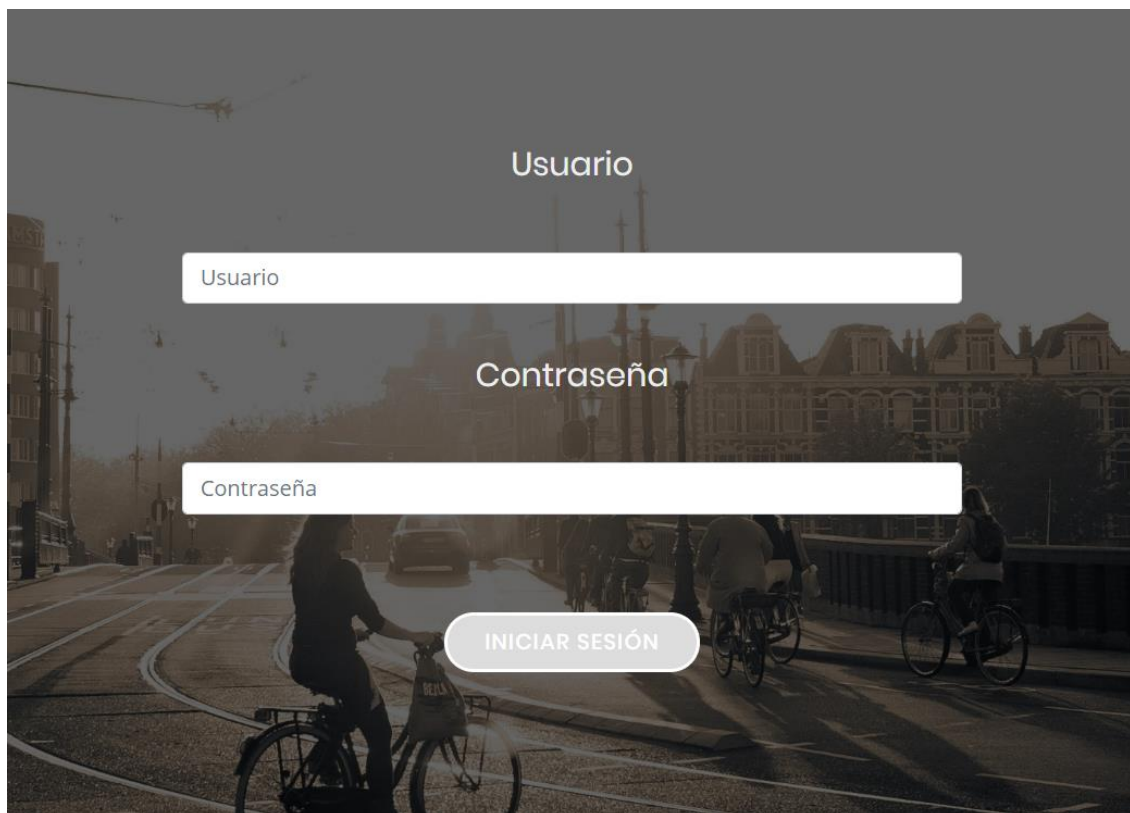


Figura 17. Login de la página web.

En la Figura 18, se observará el diagrama de flujo de ingreso al perfil en donde se comprueba el usuario y contraseña y de esta manera podrá determinar el tipo de acceso del usuario.

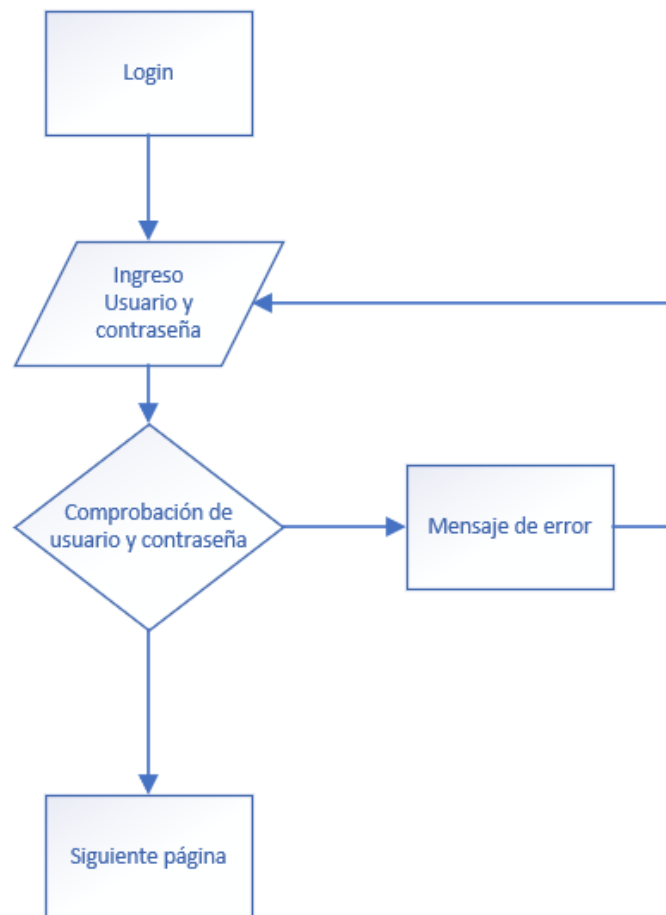


Figura 18. Diagrama de flujo, login página web.

- Rol de Administrador:  
El rol del administrador cumple una función específica que es crear un nuevo ciclista. El perfil del administrador cuenta con una pantalla principal y con un botón para el ingreso de un nuevo ciclista como se observa en la Figura 19.  
Para el ingreso de un nuevo ciclista el administrador tendrá que llenar los siguientes campos obligatorios como se muestra en la Figura 19. Además, es importante tener en cuenta que la foto que se cargue debe tener un formato jpg.

## Registro Ciclista

|   |  |
|---|--|
| Usuario   | Cédula   |
| <input type="text" value="Usuario"/>                                    | <input type="text" value="Cédula"/>              |
| Nombre  | Apellidos  |
| <input type="text" value="Nombres"/>                                    | <input type="text" value="Apellidos"/>           |
| Dirección   | Correo   |
| <input type="text" value="Dirección"/>                                  | <input type="text" value="ejemplo@dominio.com"/> |
| Teléfono  | Contraseña                                       |
| <input type="text" value="Telefono"/>                                   | <input type="text" value="Password"/>            |
| <input type="text" value="Foto"/> <input type="button" value="Browse"/> |  |
| <input type="button" value="Guardar"/>                                  |  |

*Figura 19.* Registro de un nuevo usuario.

En caso de ingresar un campo de manera errónea el programa se encargará de mostrar un mensaje de error indicando que campo está incorrecto y con su corrección como se muestra en la Figura 20.

## Registro Ciclista

|  |   |
|--|---|
| Usuario<br><input type="text" value="hmolina"/>        | Cédula<br><input type="text" value="1752068435"/>       |
| Nombre<br><input type="text" value="Nombres"/>         | Apellidos<br><input type="text" value="Ramón Ramos"/>   |
| Dirección<br><input type="text" value="Barrio nuevo"/> | Correo<br><input type="text" value="cramos@gmail.com"/> |
| Teléfono<br><input type="text" value="0984661756"/>    | Contraseña<br><input type="password" value="....."/>    |
| Foto <span style="float: right;">Browse</span>         |   |
| <b>Error!</b><br>Ingrese sus dos nombres.              |   |
| <input type="button" value="Guardar"/>                 |   |

*Figura 20.* Error de ingreso de un nuevo ciclista

Figura 6. Pantalla ingreso de nuevo usuario, ingreso de dato erróneo.

También en la pantalla principal del perfil del administrador contiene una tabla en donde muestra todos los ciclistas ingresados con su respectivo número de serie del dispositivo. Los campos que se muestran en la tabla son los siguientes:

- Usuario.
- Nombre y apellido del ciclista.
- Número de serie del dispositivo.
- Rol de Ciclista:

El perfil del ciclista está conformado por un menú donde tiene varias opciones como inicio, mapa, estadísticas y tiempo real. Como se muestra en la Figura 21.



Figura 21. Menú de perfil ciclista.

- Mapa: en esta sección los ciclistas pueden observar las rutas realizadas. Este proceso se hace mediante un calendario que guardan las rutas realizadas en el día. En la Figura 22, se indica cómo los ciclistas podrán realizar una búsqueda de una determinada fecha y de esta manera poder graficar las rutas, esta consiste en dos banderas que indican el inicio y fin del recorrido, se grafica con una línea indicando el tramo realizado como puede observar en la Figura 23. Así mismo, esta sección cuenta con una tabla de promedio de velocidad y altura, esto se lo realiza para que los ciclistas tengan un control de sus rutas.

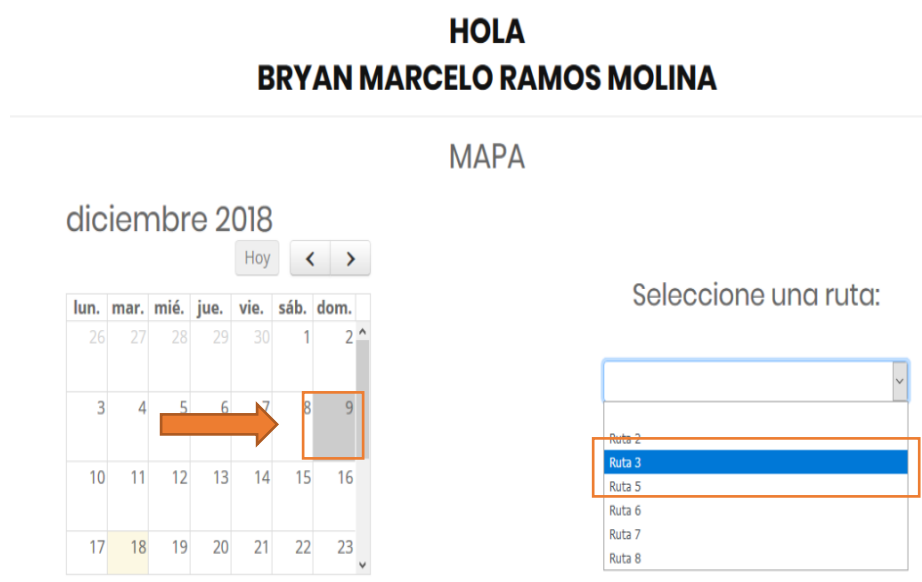


Figura 22. Selección de la búsqueda mediante del calendario.

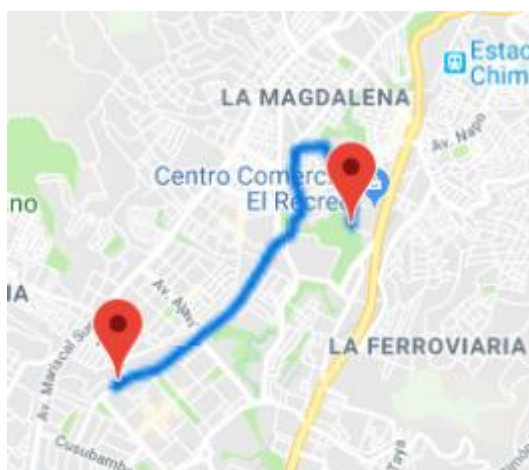


Figura 23. Gráfica de ruta con línea en el mapa.

- Estadísticas: en esta sección los ciclistas podrán visualizar estadísticas realizadas. En la Figura 22, se observa como los ciclistas pueden realizar consultas mediante el día y se desplegara todas las rutas realizadas esa determinada fecha. Una vez seleccionada la ruta se muestra las estadísticas de velocidad y altura, como se indica en la Figura 23, además, con estas gráficas los ciclistas pueden ver mediante un puntero que indica cada coordenada con su respectiva velocidad o altura en una determinada hora.



Figura 24. Calendario y selección de ruta.

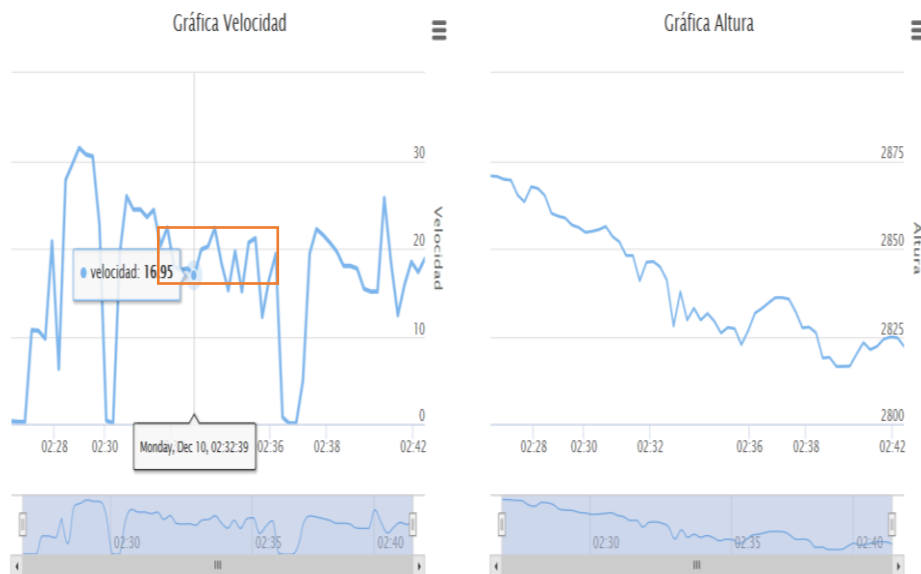


Figura 25. Estadísticas de velocidad y altura.

- Monitoreo: esta sección el ciclista podrá observar su ubicación en tiempo real. Cada vez que se mueva el ciclista, el puntero le seguirá. Además, con esta sección se guardan cada ruta realizada por el ciclista.

### 3.6. Diseño base de datos

El segundo paso fue la creación de un diagrama lógico que describe las necesidades de una organización y también representa entidades, atributos, relaciones, identificadores únicos, subtipos y restricciones entre relaciones (Rivero, 2004).

En la Figura 24, se muestra el modelo lógico, en el cual se puede observar las tablas necesarias para la implementación de la base de datos, las relaciones que existen y los atributos requeridos. Es importante generar este modelo porque ayuda a visualizar de forma sencilla como está constituida la base de datos.

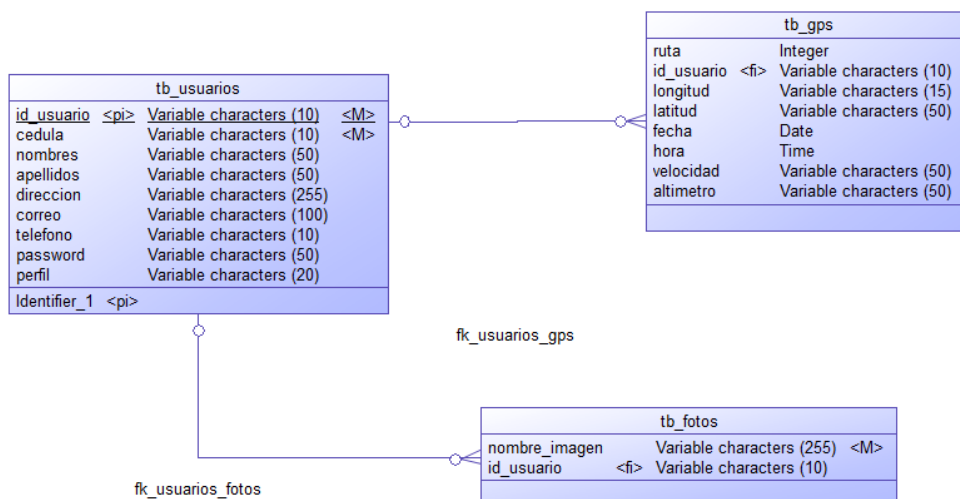


Figura 26. Diagrama lógico base de datos.

Adicional se elaboró el diccionario de datos que contiene una breve descripción y características de los atributos que se van a utilizar en la base de datos, a continuación, se detalla esta información:

- Tb\_usuarios:

Tabla 28.

Tabla de diccionario de datos de la tabla usuario.

| Nombre del campo | Código     | Tipo de datos | Descripción del campo           | Tamaño del campo |
|------------------|------------|---------------|---------------------------------|------------------|
| Id_usuario       | ID_USUARIO | varChar       | Identificador único del usuario | 10               |
| Cédula           | CEDULA     | varChar       | Identificador único             | 10               |
| Nombres          | NOMBRES    | varChar       | Nombres completos del usuario   | 50               |
| Apellidos        | APELLIDOS  | varChar       | Apellidos completos del usuario | 50               |
| Dirección        | DIRECCION  | varChar       | Dirección donde vive el usuario | 255              |
| Correo           | CORREO     | varChar       | Correo electrónico del usuario  | 100              |



|          |          |         |  |    |
|----------|----------|---------|--|----|
| Teléfono | TELEFONO | varChar | Número de teléfono de ser celular.     | 10 |
| Password | PASSWORD | varChar | Contraseña puede ser números o letras. | 50 |
| Perfil   | PERFIL   | varChar | Ciclista o Administrador               | 20 |

- Tb\_gps:

Tabla 29.

*Tabla de diccionario de datos de la tabla gps.*

| Nombre del campo | Código     | Tipo de datos | Descripción del campo                                  | Tamaño del campo |
|------------------|------------|---------------|--|------------------|
| Ruta             | RUTA       | Int           | Ruta única   |                  |
| Id_usuario       | ID_USUARIO | varChar       | Atributo de otra tabla, clave foránea                  | 10               |
| Longitud         | LONGITUD   | varChar       | Coordenada de longitud                                 | 50               |
| Latitud          | LATITUD    | varChar       | Coordenada latitud                                     | 50               |
| Fecha            | FECHA      | Date          | Fecha, formato año-mes-día                             |                  |
| Hora             | HORA       | Time          | Hora formato 24 horas, formato hora: minutos: segundos |                  |
| Velocidad        | VELOCIDAD  | varChar       | Velocidad en la que está yendo el ciclista formato 5km | 50               |
| altimetro        | ALTIMETRO  | varChar       | Altitud con nivel al mar                               | 50               |

- Tb\_fotos:

Tabla 30.

*Tabla de diccionario de datos de la tabla fotos.*

| Nombre del campo | Código        | Tipo de datos | Descripción del campo                  | Tamaño del campo |
|------------------|---------------|---------------|--|------------------|
| Nombre_imagen    | NOMBRE_IMAGEN | varChar       | Nombre de imagen el formato de ser jpg | 255              |
| Id_usuario       | ID_USUARIO    | varChar       | Atributo de otra tabla, clave foránea  | 10               |

De igual manera se implementó un modelo físico que representa objetos de datos relacionales con sus respectivas relaciones. Un modelo físico puede generar sentencias DDL y que se pueden desplegar a un servidor de base de datos (Barranco de Areba, 2001).

En la Figura 25, se puede mostrar el modelo físico de la base de datos del diseño. Con este modelo se puede generar el script que se sube en la base de datos.

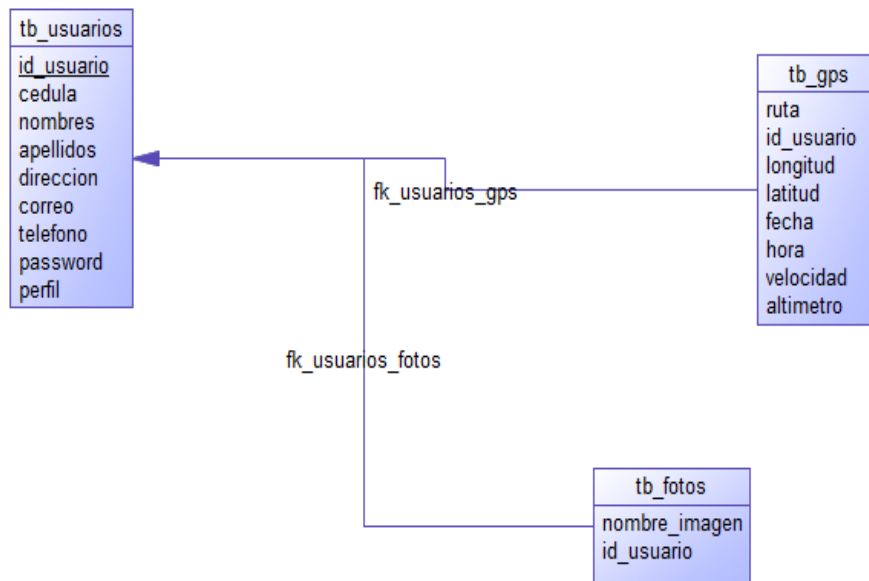


Figura 27. Modelo físico base de datos.

#### 4. Pruebas y Resultado

El prototipo cuenta con características específicas que requieren comprobarse mediante diferentes pruebas, estas ayudan a validar las falencias que se presentan, para que de esta manera se pueda comparar con los objetivos definidos en este proyecto. Es por esta razón que se logra conocer que mejoras podremos realizar posteriormente y también conocer el alcance del sistema.

La primera prueba que se realizó fue la configuración del GPS, este permite mostrar la ubicación exacta del ciclista. En la Figura siguiente, se indica como este sensor envía las coordenadas, es decir, lo realiza en un formato NMEA.

```

$GPGSA,A,3,19,22,23,07,09,06,48,17,18,,,,,1.90,0.85,1.71*02
$GPGSV,4,1,14,01,48,041,,03,27,125,19,06,17,212,28,07,80,005,19*70
$GPGSV,4,2,14,09,34,197,40,17,34,289,32,18,19,037,28,19,21,258,17*71
$GPGSV,4,3,14,22,22,100,21,23,21,164,23,28,13,337,23,30,45,333,21*75
$GPGSV,4,4,14,48,28,270,36,51,56,271,*79
$GPGLL,0015.99224,S,07832.69149,W,031203.00,A,D*6D
$GPRMC,031204.00,A,0015.99224,S,07832.69144,W,0.061,,040119,,,D*74
$GPVTG,,T,M,0.061,N,0.113,K,D*22
$GPGGA,031204.00,0015.99224,S,07832.69144,W,2,09,0.85,2882.7,M,13.7,M,,0000*62
$GPGSA,A,3,19,22,23,07,09,06,48,17,18,,,,,1.90,0.85,1.71*02
$GPGSV,4,1,14,01,48,041,,03,27,125,19,06,17,212,28,07,80,005,19*70
$GPGSV,4,2,14,09,34,197,40,17,34,289,33,18,19,037,28,19,21,258,17*70
$GPGSV,4,3,14,22,22,100,22,23,21,164,23,28,13,337,23,30,46,333,21*75
$GPGSV,4,4,14,48,28,270,36,51,56,271,*79
$GPGLL,0015.99224,S,07832.69144,W,031204.00,A,D*67
$GPRMC,031205.00,A,0015.99223,S,07832.69142,W,0.024,,040119,,,D*75
$GPVTG..T..M.0.024.N.0.045.K.D*21

```

Figura 28. Recepción de coordenadas en formato NMEA.

Es por esta razón que se tiene que separar los datos para que de esta manera se tenga una mejor administración de la información y a su vez poder utilizarlas en diferentes configuraciones.

```

GPS GY-GPS6MV2 Leantec
---Buscando senal---

Latitud/Longitud: -0.26657, -78.54485
Fecha: 4/1/2019 Hora: 3:14:30.0
Altitud (metros): 1000000.00
Rumbo (grados): 0.00
Velocidad(kmph): 0.06

Latitud/Longitud: -0.26657, -78.54485
Fecha: 4/1/2019 Hora: 3:14:31.0
Altitud (metros): 1000000.00
Rumbo (grados): 0.00
Velocidad(kmph): 0.04

```

Figura 29. Recepción de coordenadas formato TinyGPS.

En la Figura 29, se muestra de forma ordenada la información del GPS. Mediante el uso de la librería “TinyGPS”, esta se encarga de recibir los datos del sensor y a través de sus funciones nos permite crear un formato sencillo de los elementos. Además, es importante tener en cuenta el huso horario dependiendo del país ya que el sensor recibe información de los satélites, en los cuales la hora tiene como referencia el huso horario cero. Es por este motivo que se debe ajustar al horario del Ecuador, en donde la diferencia con la zona cero es de menos cinco horas. Para esto se desarrolló de la solución que se indica en la siguiente Figura.

```

if ((int)hour < 5) {
    int sobra = ((int)hour) - 5;
    hor = 24 + sobra;
    dia = (int)day - 1;
} else {
    hor = ((int)hour) - 5;
    dia = (int)day;
}

```

Figura 30. Código del ajuste de la zona horario de Ecuador.

En la segunda prueba se configuro el envío de información hacia la base de datos. Se realizaron varias pruebas en la cual, se pudo identificar que la configuración de la APN depende de la operadora móvil que se utilice. Así mismo, el prototipo interactúa con la base de datos y el aplicativo web.

Y se realizaron varias pruebas en las cuales resultaron erróneas al momento de enviar información a la base de datos como se puede observar en la Figura 31.

| ruta | id_usuario | longitud | latitud | fecha      | hora     | velocidad | altimetro |
|------|------------|----------|---------|------------|----------|-----------|-----------|
| 20   | bramos     | 0        | 0       | 0000-00-00 | 00:00:00 | 0         | 0         |
| 20   | bramos     | 0        | 0       | 0000-00-00 | 00:00:00 | 0         | 0         |
| 20   | bramos     | 0        | 0       | 0000-00-00 | 00:00:00 | 0         | 0         |
| 20   | bramos     | 0        | 0       | 0000-00-00 | 00:00:00 | 0         | 0         |
| 20   | bramos     | 0        | 0       | 0000-00-00 | 00:00:00 | 0         | 0         |
| 20   | bramos     | 0        | 0       | 0000-00-00 | 00:00:00 | 0         | 0         |

Figura 31. Información errónea en la base de datos.

Finalmente, conociendo la información errónea, se procedió a realizar los cambios necesarios para el envío a la base de datos.

| ruta | id_usuario | longitud  | latitud  | fecha      | hora     | velocidad | altimetro |
|------|------------|-----------|----------|------------|----------|-----------|-----------|
| 18   | bramos     | -78.54494 | -0.26656 | 2019-01-03 | 22:53:10 | 0.44      | 2859.2    |
| 18   | bramos     | -78.54494 | -0.26657 | 2019-01-03 | 22:53:21 | 0.09      | 2860      |
| 18   | bramos     | -78.54494 | -0.26658 | 2019-01-03 | 22:53:32 | 0.2       | 2859.7    |
| 18   | bramos     | -78.54494 | -0.26658 | 2019-01-03 | 22:53:43 | 0.46      | 2860.9    |
| 18   | bramos     | -78.54494 | -0.26658 | 2019-01-03 | 22:53:54 | 0.09      | 2860.9    |

Figura 32. Información correcta en la base de datos.

En la tercera prueba se realizó la pérdida de señal del GSM, cuando suceda este caso todas las coordenadas serán enviadas a un backup, es decir, el prototipo está conformado por una MicroSD que ayuda almacenar dichas coordenadas en un ".txt", en el cual este archivo se crea automáticamente. El proceso que se

realiza es el siguiente, se comienza la ruta después de un minuto se detectó la pérdida de parte del GSM, lo cual se comienza a almacenar los datos en el backup como se muestra en la Figura 33 y, además, después de 2 minutos la señal regreso y todo continua normalmente.

```

---Buscando senal---
SD card is ready to use.
Latitud/Longitud: -0.26660, -78.54486
Fecha: 4/1/2019 Hora: 4:15:55Altitud (metros): 1000000.00
Rumbo (grados): 0.00
Velocidad(kmph): 0.17

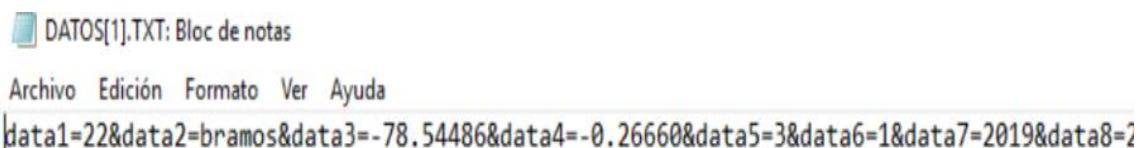
Writing to file...
Done.
Latitud/Longitud: -0.26660, -78.54486
Fecha: 4/1/2019 Hora: 4:15:56Altitud (metros): 1000000.00
Rumbo (grados): 0.00
Velocidad(kmph): 0.59

Writing to file...
Done.
Latitud/Longitud: -0.26660, -78.54486
Fecha: 4/1/2019 Hora: 4:15:57Altitud (metros): 1000000.00
Rumbo (grados): 0.00
Velocidad(kmph): 0.33

Writing to file...
Done.

```

Figura 33. Envío de datos a la MicroSD.



DATOS[1].TXT: Bloc de notas

Archivo Edición Formato Ver Ayuda

data1=22&data2=bramos&data3=-78.54486&data4=-0.26660&data5=3&data6=1&data7=2019&data8=2

Figura 34. Escribir en la MicroSD.

Las coordenadas que se encuentran guardadas se enviarán a la base de datos después de que se finalice la ruta como se observa en la Figura 35 y de esta manera el ciclista no se encuentra afectado en la gráfica de su ruta.

| ruta | id_usuario | longitud  | latitud | fecha      | hora     | velocidad | altimetro |
|------|------------|-----------|---------|------------|----------|-----------|-----------|
| 22   | bramos     | -78.54486 | -0.2666 | 2019-01-03 | 23:15:55 | 0.17      | 1000000   |
| 22   | bramos     | -78.54486 | -0.2666 | 2019-01-03 | 23:15:56 | 0.59      | 1000000   |
| 22   | bramos     | -78.54485 | -0.2666 | 2019-01-03 | 23:15:57 | 0.33      | 1000000   |
| 22   | bramos     | -78.54485 | -0.2666 | 2019-01-03 | 23:15:58 | 1.52      | 1000000   |
| 22   | bramos     | -78.54485 | -0.2666 | 2019-01-03 | 23:15:59 | 0.72      | 1000000   |

Figura 35. Coordenas enviadas a la base de datos mediante el MicroSD.

La cuarta prueba que se realizó fue las notificaciones de accidentes del ciclista, estas se realizaron mediante un correo electrónico. Estas notificaciones se envían a un familiar del usuario. En la Figura 36, se puede observar de donde está llegando el mensaje, en este caso es de NodoRed que es el servidor web seleccionado y con una firma del aplicativo web (www.tubicisegura.com).

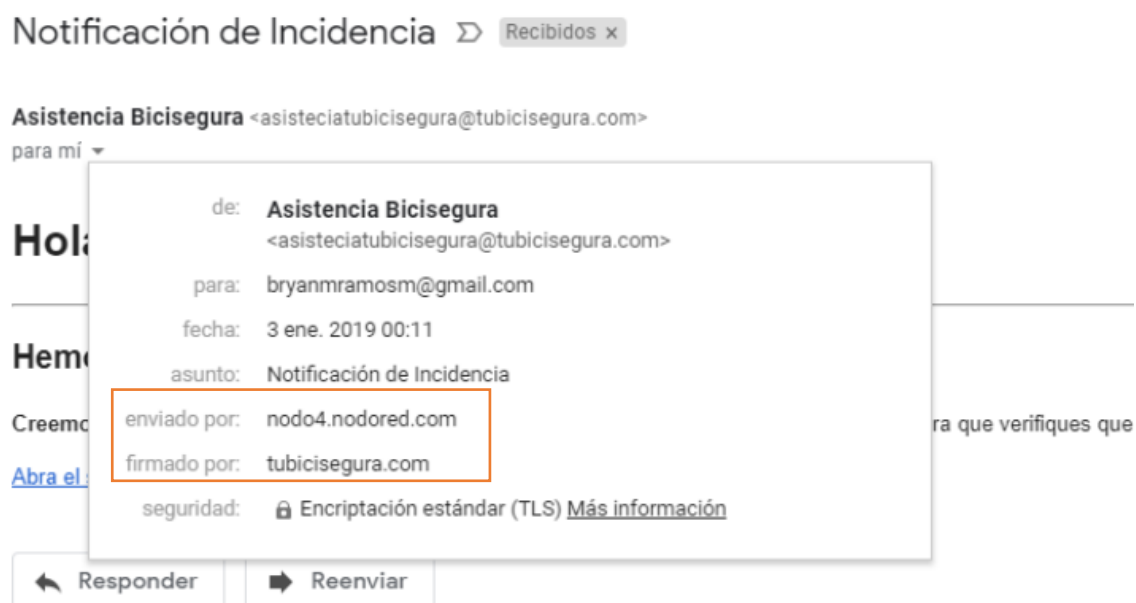


Figura 36. Información del correo electrónico al usuario.

Además, la notificación cuenta con un mensaje de alerta al familiar, como se observa en la Figura 37. Así mismo, este mensaje cuenta con un enlace. Este ayuda al familiar a ver la ubicación exacta del ciclista en caso de algún incidente. El accidente se graficará en un mapa dentro del aplicativo web. Como se muestra en la Figura 38.

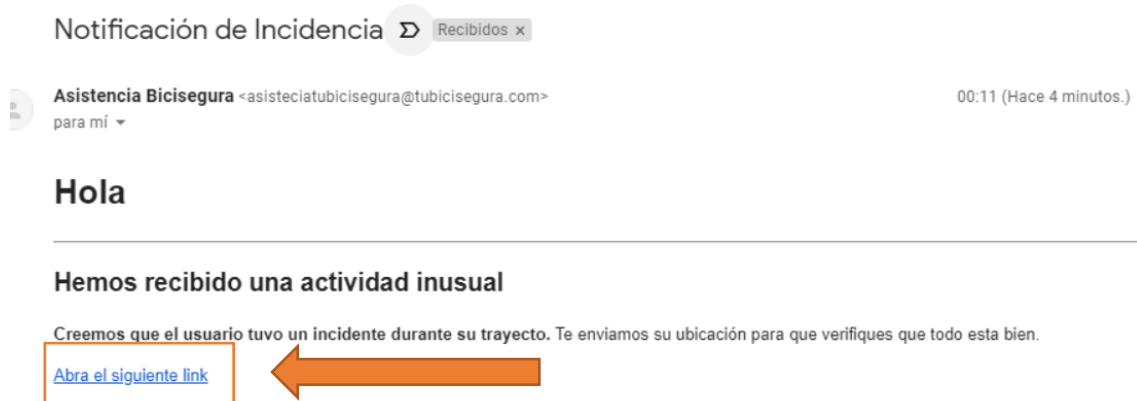


Figura 37. Mensaje de alerta de un accidente del usuario.



Figura 38. Gráfica del accidente en el mapa.

El prototipo realiza una validación en caso de un accidente, esta se lo realiza cuando se tenga una falsa alerta, es decir, si el usuario tiene algún accidente el prototipo confirma por cinco veces la alerta, en caso de que se cumpla ésta, se notificara al familiar del ciclista que tuvo algún incidente, caso contrario la alerta será falsa y no se enviará ningún tipo de notificación.

En la Figura 39, se puede observar que se realizó una prueba, en el cual el prototipo detecto tres alertas, en este caso no se mandó ningún correo electrónico.



```
Ruta: 11
giroscopio iniciado correctamente
---OK---
NO
NO
ALERTA
ALERTA
ALERTA
NO
NO
ALERTA
ALERTA
ALERTA
NO
Writing to file...
Done.
```

Figura 39. Falsa alerta de accidente.

Además, en la siguiente Figura, se puede visualizar que el prototipo detecto cinco alertas las cuales transcurren alrededor de un minuto y mando una notificación con su ubicación exacta del accidente.

```
Ruta: 7
giroscopio iniciado correctamente
---OK---
NO
NO
ALERTA
ALERTA
ALERTA
ALERTA
ALERTA
ALERTA ENVIADA
ALERTA
Writing to file...
Done.
Read:
Ruta: 9
```

Figura 40. Alerta de accidente y envío de correo electrónico.

En la quinta prueba como se muestra en la Tabla 30, se realizaron tres rutas en diferentes tiempos. El número de ruta corresponde al valor con el que se almacena en la base de datos.

Tabla 31.

*Tabla de rutas y tiempo.*

| Ruta | Tiempo     |
|------|------------|
| 2    | 1 hora     |
| 3    | 15 minutos |
| 5    | 30 minutos |

Cada ruta tiene una cierta cantidad de coordenadas y varía dependiendo al tiempo que se lo realiza, estas rutas son graficadas en el mapa del aplicativo web marcadas con dos punteros que muestra su inicio y su fin. El error es calculado mediante la siguiente fórmula matemática:

$$\text{Error de coordenadas} = 100\% - \left( \frac{\text{Puntos Reales} * 100}{\text{Puntos Teóricos}} \right)$$

Tabla 32.

*Análisis de rutas, dependiendo del tiempo.*

|                    | RUTA 2                               | RUTA 3   | RUTA 5                   |
|--------------------|--------------------------------------|--|--------------------------|
| Inicio             | Av. General Quisquis (Sur de Quito)  | Ubicación: Avenida Cardenal de la Torre (Sur de Quito) | Avenida De Los Granados  |
| Fin                | Av. de Los Granados (Norte de Quito) | Rodrigo Torres (Sur de Quito)                          | Antonio Nuñez (Carcelén) |
| # Puntos (Teórico) | 327                                  | 81   | 150                      |
| # Puntos (Real)    | 300                                  | 75   | 148                      |
| Error (%)          | 8                                    | 7  | 2                        |

En el Tabla 31, se puede observar la pérdida que tuvieron en cada ruta con su respectivo tiempo. Se comprobó mediante porcentajes que la pérdida de datos es mínima por lo que no influye en la gráfica de las rutas.

Así mismo, se puede observar las gráficas en el mapa con los diferentes tiempos. Las gráficas se realizaron con puntos, es una prueba para verificar de cómo se muestran en el mapa como se indica en las Figuras 41, 42 y 43.

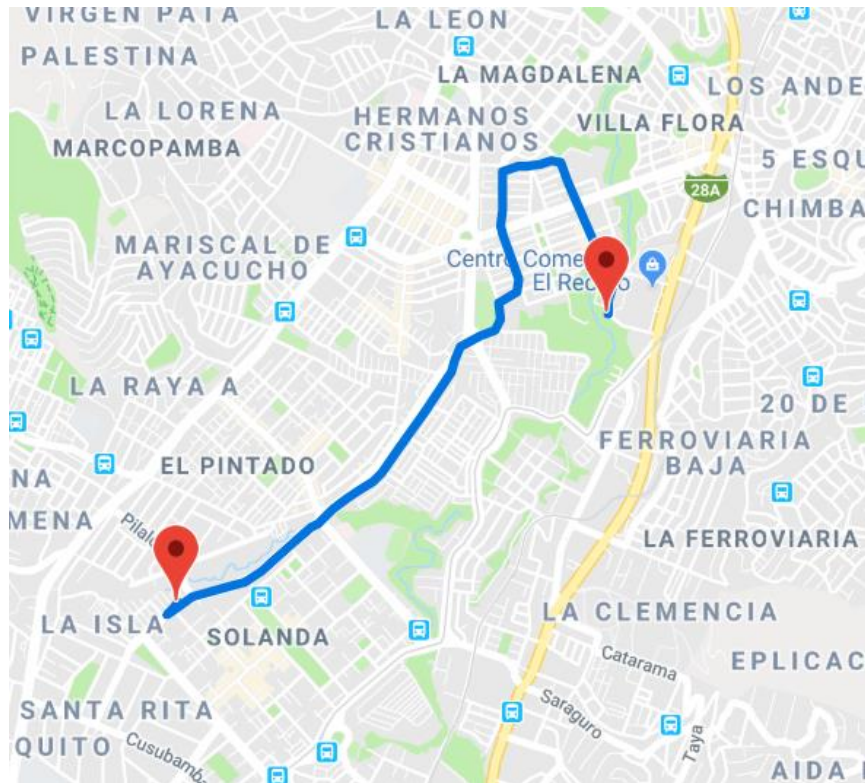


Figura 41. Gráfica de la ruta de 15 minutos.

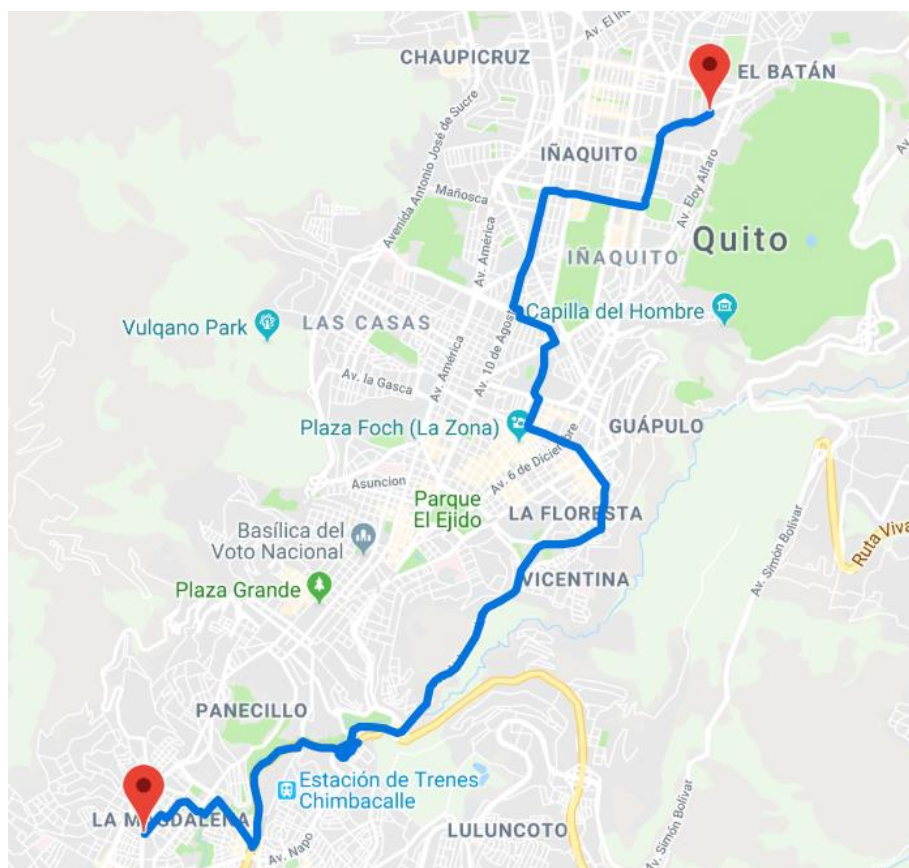


Figura 42. Gráfica de la ruta de 1 hora.

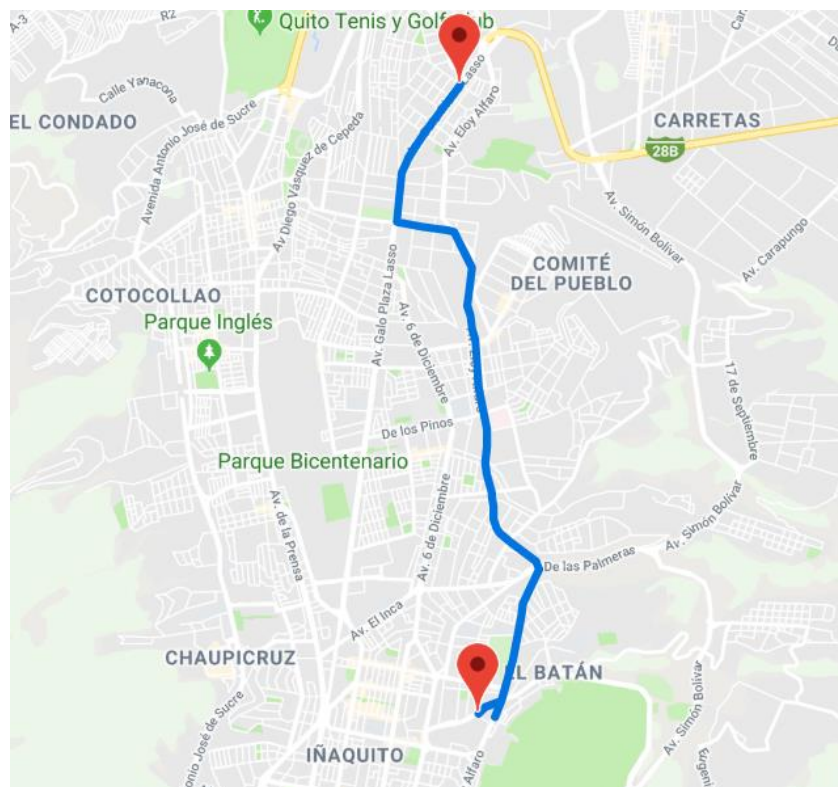


Figura 43. Gráfica de la ruta de 30 minutos.

## 5. Conclusiones y Recomendaciones

### 5.1. Conclusiones

Con el desarrollo del proyecto de titulación se puede concluir que el sistema de monitoreo en tiempo real permite mantener un control continuo de un elemento, en este caso de una bicicleta, para así poder conocer la ubicación exacta y características como velocidad y altura para tener más información del elemento rastreado.

El modelo de GPS que se utilice debe tener la precisión adecuada para el prototipo, debido a que en este proyecto se encuentra al aire libre cualquier tipo de GPS podría ayudarnos con la calibración de acuerdo al objetivo, en cambio en un lugar cerrado se debe usar un dispositivo que no pierda la señal.

En la configuración inicial el módulo GPS envía datos desordenados acerca de la ubicación, altura y velocidad, es importante que se filtre esta información para poder mantener un orden y utilizar los datos que sean necesarios para poder dividir y transformar cada campo.

Los datos se envían mediante la red móvil, por esta razón la pérdida de datos se enlaza con el uso de operadora ya que estas cuentan con diferente cobertura, esta se puede elegir indistintamente, se debe tomar en cuenta que para la configuración se necesita el ingreso de APN que es más sencillo si la operadora no tiene usuario ni contraseña.

El módulo GSM realiza la conexión a internet del prototipo, mediante comandos AT. Para validar cada envío por parte de los sensores es importante realizar la conexión para cada envío de datos.

En el caso de que exista una pérdida de datos, se implementó un backup con una MicroSD, en la que se guardan los datos que no se envían cuando existen un fallo en la conexión y cuando se retoma la señal estos se envían para que ingresen a la base de datos mediante la hora.

La capacidad de la MicroSD depende del usuario ya que se puede colocar una memoria de cualquier tamaño y los datos ingresados se guardan en un archivo “.txt”, donde se muestran las coordenadas.

Se concluyó el uso de una unidad inercial porque cuenta con un giroscopio que mediante la velocidad angular da una alerta al usuario de un accidente ocurrido al ciclista, se configura mediante un intervalo para que el dato que se obtiene sea preciso.

Los pulsadores integrados con los leds permiten realizar acciones como: comprobación de conexión, inicio y fin de ruta. Es importante contar con estos elementos para comprobar que el envío de datos está realizándose, así como también para validar que no se envíe información.

Mediante la realización de pruebas se puede concluir que el porcentaje de pérdida de datos en una hora es de 8%, en treinta minutos es de 2% y en quince minutos es de 7%. Estos porcentajes comprueban que la pérdida de datos no representa un problema para la gráfica de rutas.

El perfil ciclista de la página web permite realizar una administración del prototipo para conocer estadísticas, tiempo real y rutas definidas por fechas. Además, cuenta con una autenticación de usuario para la seguridad del ciclista.

## **5.2. Recomendaciones**

Se recomienda el uso de una tarjeta SIM que el operador de red cuente con amplia cobertura para evitar la pérdida de datos, aunque exista un backup.

Se recomienda analizar el tiempo de conexión del GSM con la red móvil para conocer cuánto se demora el prototipo en cada envío de datos.

Para detectar la caída de la bicicleta es necesario implementar un sensor, se recomienda el uso de una unidad inercial debido a que tiene integrado un giroscopio y acelerómetro, el giroscopio permite medir el cambio de posición mediante grados mientras que el acelerómetro es la variación de la aceleración.

Se recomienda realizar un análisis del amperaje total del circuito para usar una batería adecuada.

Se recomienda realizar pruebas del prototipo en un tiempo prolongado para comprobar la robustez del mismo y conocer las fallas que tiene para mejorarlas.

## REFERENCIAS

- Arduino. (2018). *Arduino Mega 2560 REV3*. Recuperado el 16 de noviembre del 2018, de <https://store.arduino.cc/usa/arduino-mega-2560-rev3>
- Arias, M. (2017). *Aprende Programación Web con PHP y MySQL: 2ª Edición*. IT Campus Academy.
- Barranco de Areba, J. (2001). *Metodología del análisis estructurado de sistemas Volumen 20 de Libros de Texto*. Univ Pontifica Comillas.
- Bikelec. (2014). *SpyBike GPS*. Recuperado el 28 de octubre del 2018, de <https://www.bikelec.es/blog/spybike-gps/>
- Briceño Murillo, Y. F. (2016). *Diseño de un sistema de seguridad y monitoreo satelital para una bicicleta por medio redes GSM/GPRS*. Recuperado el 22 de noviembre del 2018, de <http://repository.udistrital.edu.co/bitstream/11349/5969/1/Brice%c3%b1oMurilloYesidFabian2017.pdf>
- Britannica, E. (2018). *Batería*. Recuperado el 15 de diciembre del 2018, de <https://www.britannica.com/technology/battery-electronics>
- Caicedo Pedrera, A. (2017). *Arduino para Principiantes: 2ª Edición*. IT Campus Academy.
- Combaudon, S. (2018). *MySQL 5.7 Administración y optimización*. Ediciones ENI.
- Comercio, E. (2017). *Tres bicicletas de alta gama fueron robadas en Quito*. Recuperado el 1 de noviembre del 2018, de <https://www.elcomercio.com/actualidad/bicicletas-robadas-detenidos-policia-parques.html>
- Date, C. J. (2001). *Introducción a los sistemas de bases de datos*. Pearson Educación.
- Digital Guide. (2019). *¿Qué es un servidor web y qué soluciones de software existen?*. Recuperado el 9 enero del 2019, de <https://www.ionos.es/digitalguide/servidores/know-how/servidor-web-definicion-historia-y-programas/>

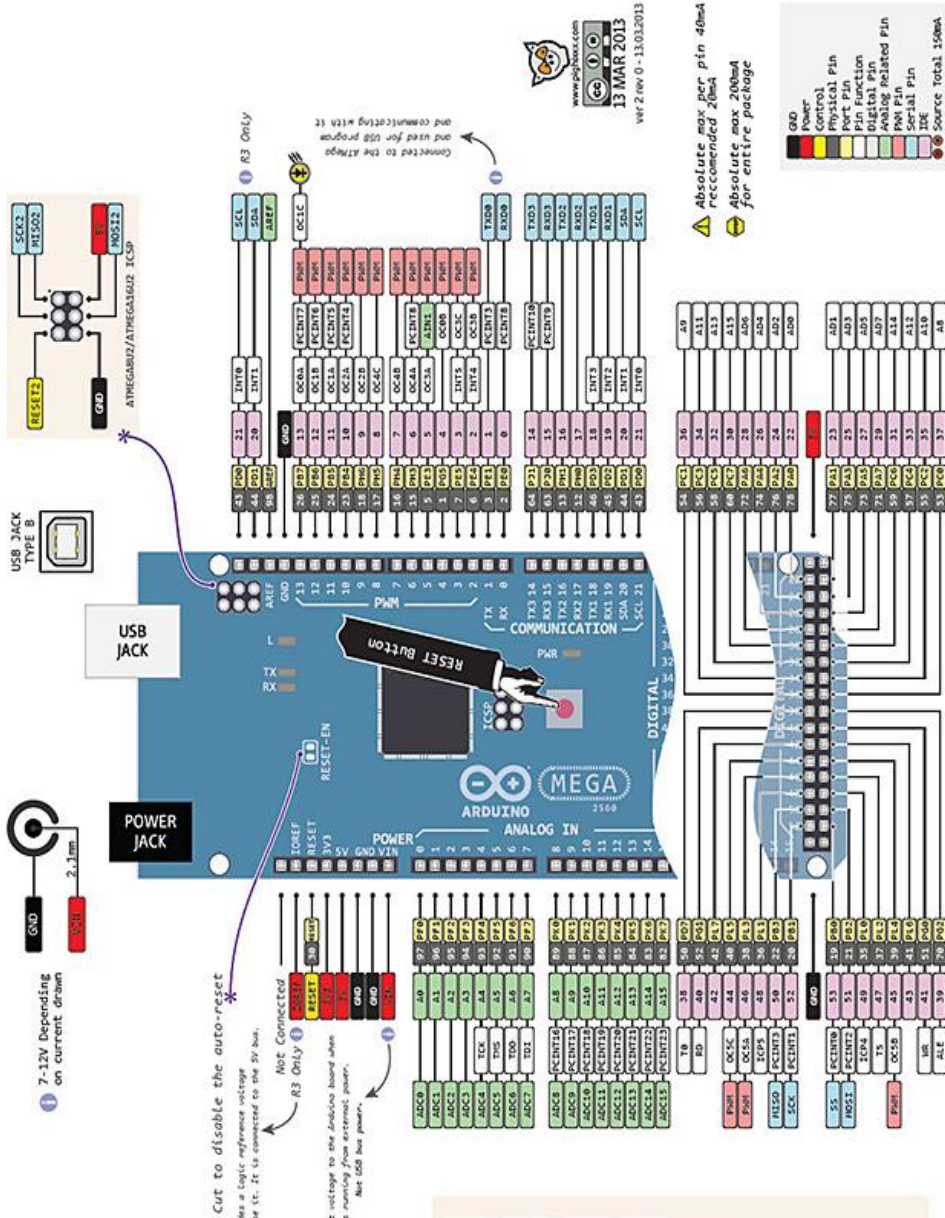


- EcuRed. (2019). *Servidor Web*. Recuperado el 10 de enero del 2019, de [https://www.ecured.cu/Servidor\\_Web](https://www.ecured.cu/Servidor_Web)
- El Telégrafo. (2016). *El uso de la bicicleta es una gran idea que ayuda a la naturaleza y a nuestra salud*. Recuperado el 20 de noviembre del 2018, de <https://www.eltelegrafo.com.ec/noticias/quito/1/el-uso-de-la-bicicleta-es-una-gran-idea-que-ayuda-a-la-naturaleza-y-a-nuestra-salud>
- Fernández, A. (2018). *La Aventura de Aprender*. Recuperado el 11 de noviembre del 2018, de <http://www.laaventuradeaprender.com/entornos-de-desarrollo-integrados/>
- Gregersen, E. (2018). *CONTROL DE CALIDAD DEL VUELO LiDAR UTILIZADO PARA LA MODELIZACIÓN 3D DE LAS FALLAS DE ALHAMA (MURCIA) Y*. Recuperado el 12 de enero del 2019, de [http://oa.upm.es/33673/1/PFC\\_MIGUEL\\_CRESPO\\_MAZO.pdf](http://oa.upm.es/33673/1/PFC_MIGUEL_CRESPO_MAZO.pdf)
- Mackenzie, D., & Sharkey, K. (2003). *Aprendiendo Visual Basic.Net en 21 Lecciones Avanzadas*. Pearson Educación.
- Pozo Ruz, Ribeiro, García Alegre, García, Guinea, & Sandoval. (2018). *SISTEMA DE POSICIONAMIENTO GLOBAL (GPS): DESCRIPCIÓN, ANÁLISIS*. Recuperado el 5 de enero del 2019 de [http://www.oocities.org/es/foro\\_gps/infografia/gps5.pdf](http://www.oocities.org/es/foro_gps/infografia/gps5.pdf)
- Rivero, E. (2004). *Bases de datos relacionales: diseño físico Volumen 19 de Ingeniería (Universidad Pontificia Comillas)*. Univ Pontificia Comillas.
- Thibaud, C. (2006). *MySQL 5: instalación, implementación, administración, programación*. Ediciones ENI.
- Torrente Artero, Ó. (2013). *Arduino : curso práctico de formación*. RC Libros.
- Torres Remon, M. Á. (2014). *Desarrollo de aplicaciones web con PHP*. Editorial Macro.
- Valdés Pérez, F., & Pallàs Areny, R. (2007). *Microcontroladores: fundamentos y aplicaciones con PIC*. Marcombo.
- Villalobos, R. (2014). *Fundamentos de programación con JAVA (100 algoritmos codificados)*. Editorial Macro.

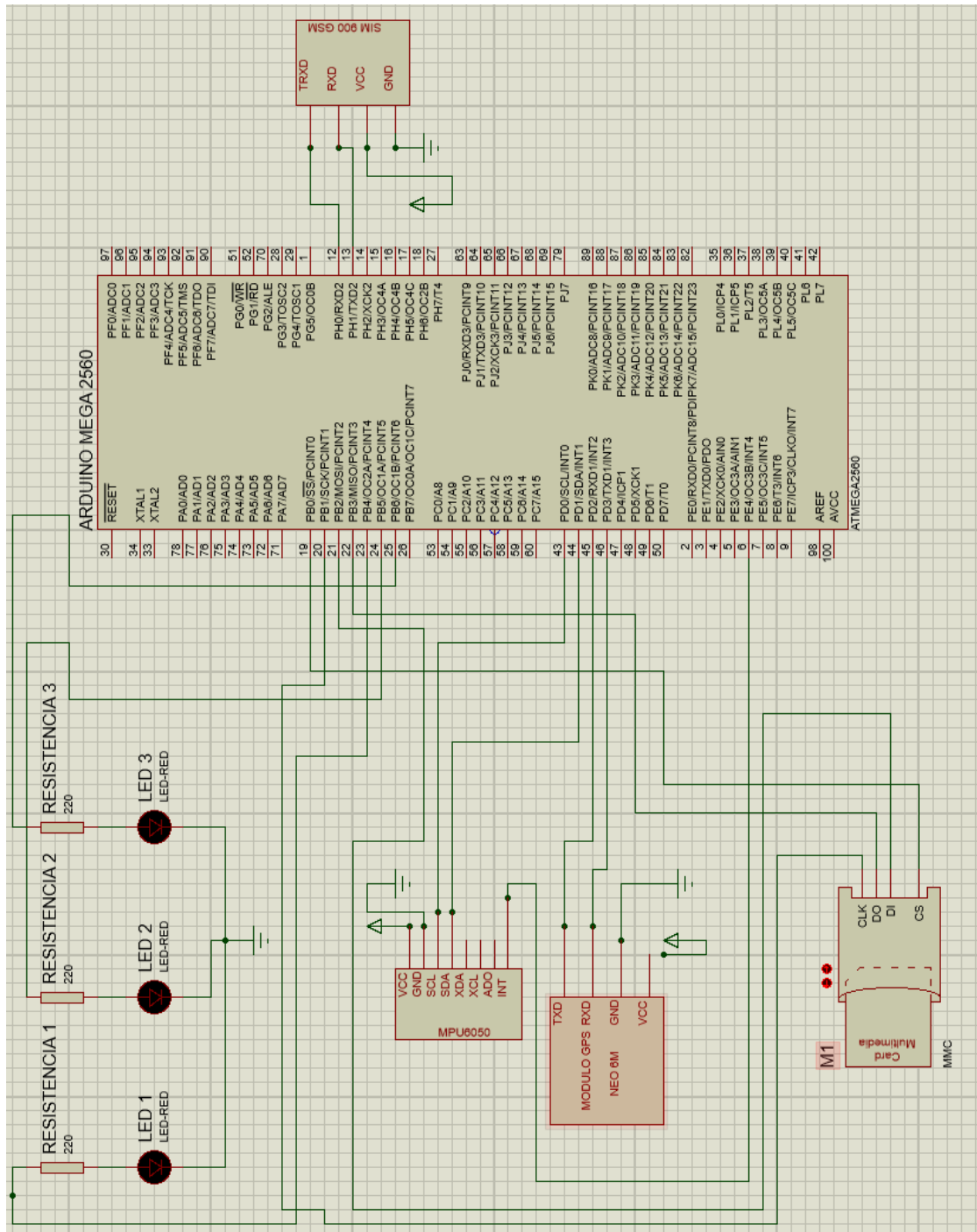
## **ANEXOS**

ANEXO 1

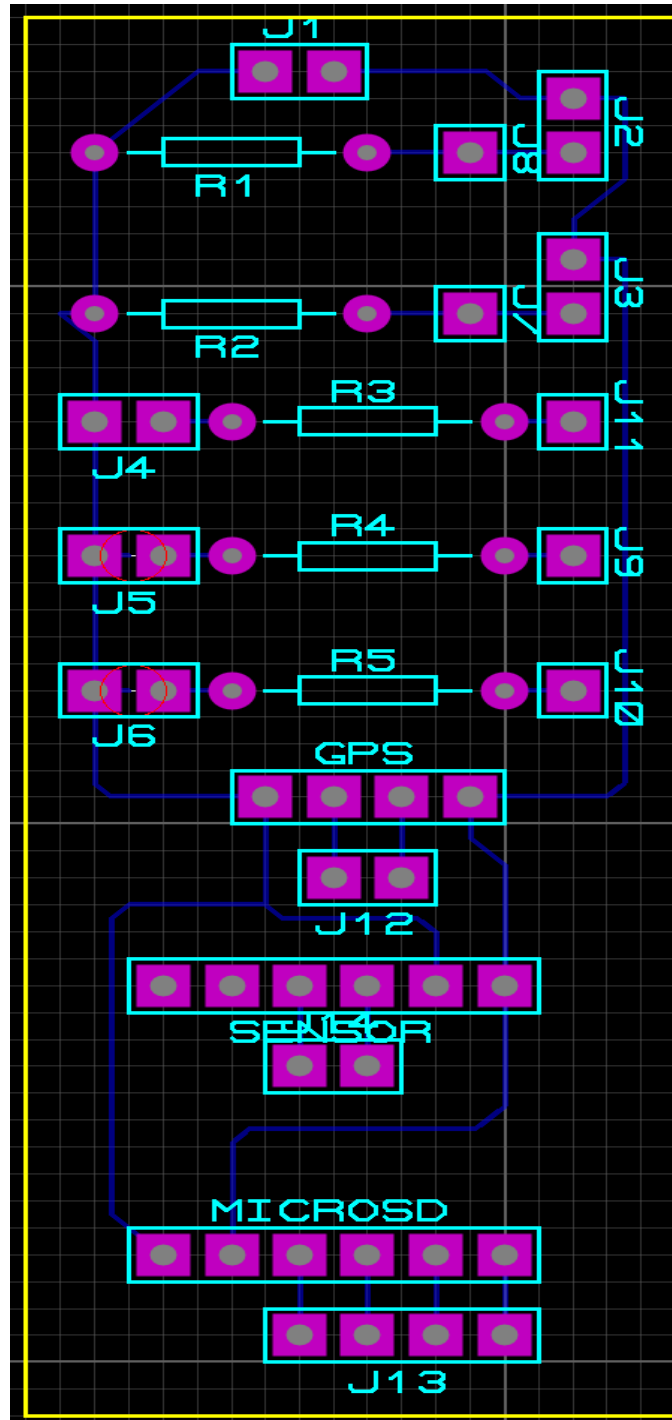
THE DEFINITIVE ARDUINO MEGA PINOUT DIAGRAM



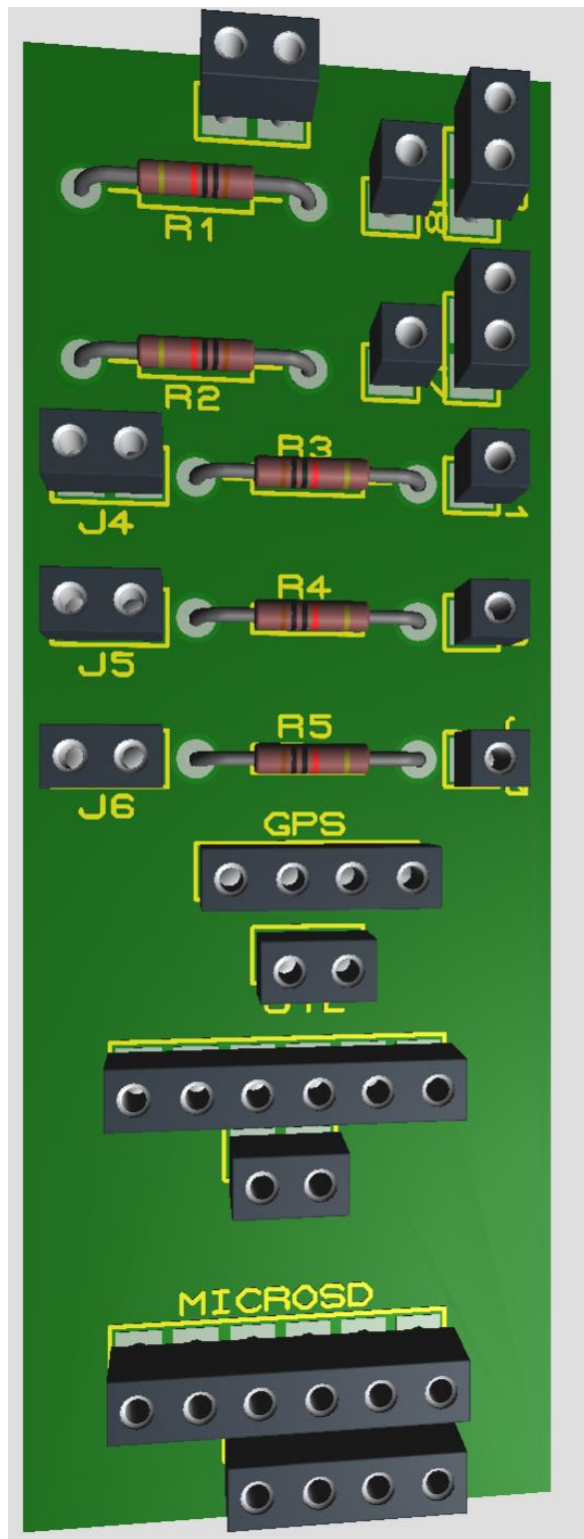
# ANEXO 2



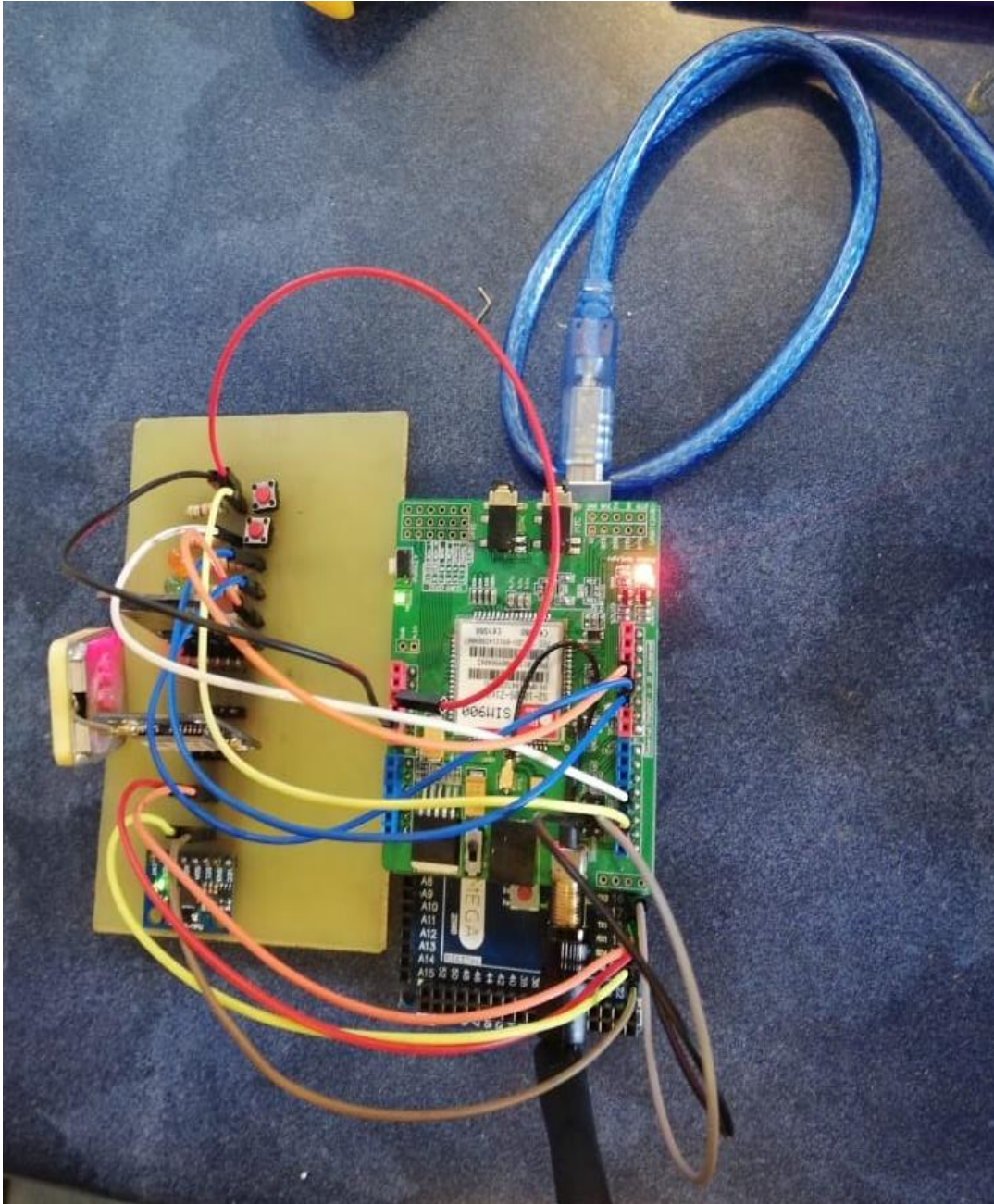
ANEXO 3



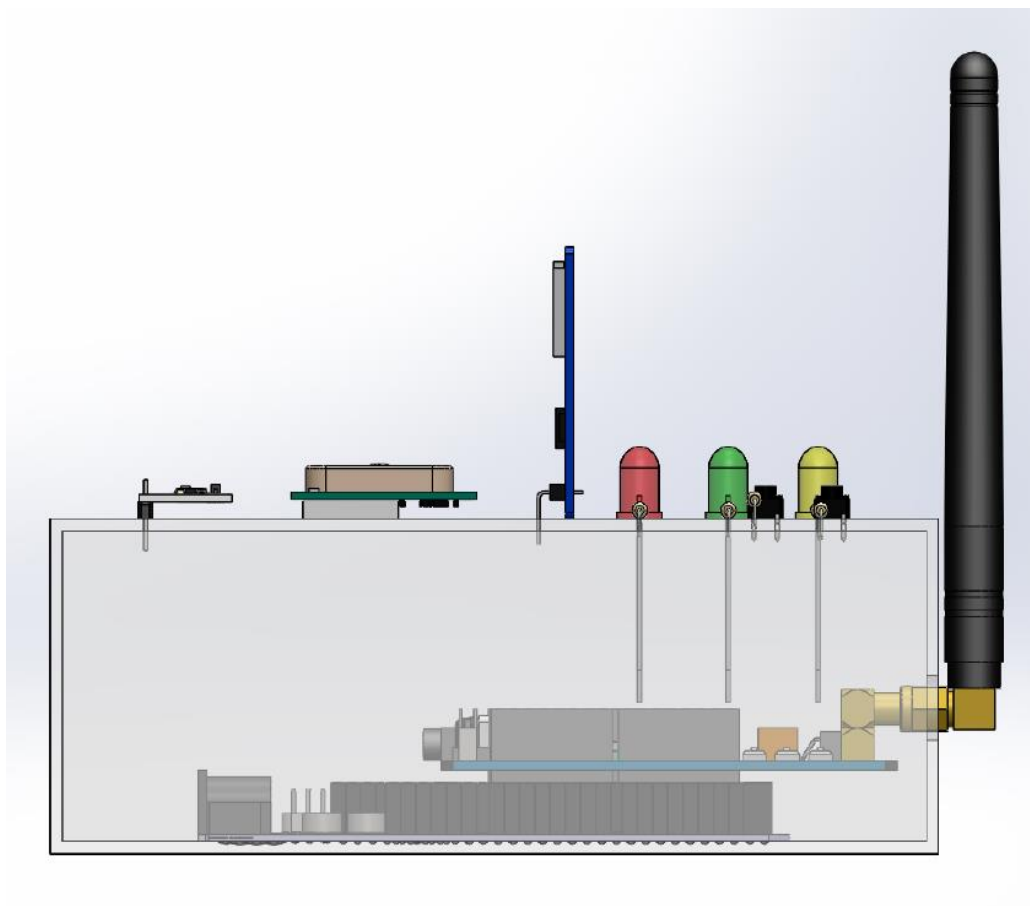
ANEXO 4



ANEXO 5



## ANEXO 6





## ANEXO 7

