



FACULTAD DE INGENIERÍA Y CIENCIAS AGROPECUARIAS

DISEÑO E IMPLEMENTACIÓN DEL PROTOTIPO DE UN SISTEMA DE
MONITOREO, DETECCIÓN DE ACCIDENTES Y ALERTA INMEDIATA PARA
MOTOCICLISTAS

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

Profesora Guía
MSc. Carmen Alicia Carabalí Carabalí

Autores
Daniela Viviana Castillo Martínez
Diego Andrés Bonilla Alcázar

Año
2016

DECLARACIÓN DEL PROFESOR GUÍA

“Declaro haber dirigido este trabajo a través de reuniones periódicas con los estudiantes, 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”.

Carmen Alicia Carabalí Carabalí
Máster en Ingeniería Biomédica
CI. 1716369713

DECLARACIÓN DE AUTORÍA DE LOS ESTUDIANTES

“Declaramos que este trabajo es original, de nuestra 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”.

Daniela Viviana Castillo Martínez

CI. 1721896049

Diego Andrés Bonilla Alcázar

CI. 0503143745

AGRADECIMIENTO

Agradezco principalmente a Dios, a mis padres por ser un pilar fundamental en mi vida y por haberme forjado como una gran persona, todos mis logros es gracias al apoyo incondicional que me han brindado, a mis hermanos por su gran apoyo diario.

Daniela Castillo Martínez.

AGRADECIMIENTO

Agradezco a mis padres por el apoyo incondicional que me han dado durante todos mis estudios, a mis hermanos que siempre han estado apoyándome en todo momento.

Diego Bonilla Alcázar.

DEDICATORIA

A mis padres y hermanos por siempre apoyarme y estar conmigo en todo momento, con mucho amor y cariño les dedico todo mi esfuerzo, trabajo puesto para la realización de esta tesis.

Daniela Castillo Martínez.

DEDICATORIA

Dedico todo mi trabajo a mi familia, en especial a mis hermanos a los cuales mis esfuerzos se encuentra enfocados, a mi papá por apoyarme incondicionalmente, y a mi madre la luz y el eje de mi vida.

Diego Bonilla Alcázar.

RESUMEN

Los accidentes motociclísticos se incrementan continuamente, según la Agencia Nacional de Tránsito y estadísticas comparadas entre meses del año 2014 y 2015 tomando la media entre tres meses de cada año, se ha incrementado en un 0,065% los accidentes involucrados motocicletas (INEC, 2014), por lo que se puede decir que actualmente el riesgo de un accidente tiene una alta probabilidad. Adicionalmente, la baja eficiencia en el contacto a una entidad de auxilio es una gran desventaja para el usuario afectado, según el Diario La Hora de Ecuador, el tiempo óptimo de una ambulancia para llegar a su destino es de 10 minutos además de esperar 7 minutos el cual se demora el llamado de auxilio de una persona a la entidad correspondiente (Zambrano, J., 2013), por lo cual es un tiempo muy elevado en cuanto a una respuesta de auxilio.

Con el problema encontrado, este proyecto se enfocó en una solución para minimizar el tiempo de respuesta enviando un mensaje de alerta cuando un accidente ha tenido lugar, reduciendo de 7 minutos a 1 minuto el llamado de auxilio a un contacto del accidentado.

La solución consiste en un prototipo que se creó colocando sensores en un casco para determinar; si existe un impacto y aceleración en el casco del motociclista, y el pulso del este mismo individuo; además se desarrolló una aplicación móvil, la cual permite gestionar y almacenar datos del usuario. La aplicación móvil se comunica con los sensores colocados en el casco por vía Bluetooth.

En caso de que llegue a existir un accidente, la aplicación envía un mensaje de alerta a un contacto predeterminado creado por el usuario y/o a una central, en este mensaje se incluye la ubicación GPS, para que se pueda ubicar de mejor manera el lugar donde ocurrió el accidente. Con este prototipo se presenta una solución al problema expuesto, la cual tras una serie de pruebas demostró que

el tiempo de envío de un mensaje de alerta a la central establecida en una red LAN es menor a un minuto, y se determina como satisfactorio el prototipo establecido.

ABSTRACT

The vehicle motorcycle accidents continually increase, according to the National Traffic Agency and comparative statistics between months of 2014 and 2015 by taking the average of three months of each year, it has increased by 0,065% accidents involved motorcycles (INEC, 2014) so it can be said that currently the risk of an accident has a high probability. In addition, low efficiency in the contact to an entity relief is a big disadvantage for the affected user, according to the newspaper La Hora of Ecuador, the optimum time for an ambulance to get to your destination is 10 minutes plus wait 7 minutes which delay the call for help from one person to the appropriate entity (Zambrano, J., 2013), so it is very high as a relief response time.

Found the problem, this project focused on a solution to minimize the response time by sending an alert message when an accident has occurred, reducing 7 min to 1 minute distress call to a contact of the injured.

The solution is a prototype that was created by placing sensors in a helmet to determine; if there is an impact and acceleration on the helmet of the motorcyclist and the pulse of the same individual; also it developed a mobile application, which allows you to manage and store user data. The mobile application communicates with sensors placed on the helmet via Bluetooth.

If it gets to be an accident, the application sends an alert message to a contact default created by the user and / or a central, in this message GPS location is included, so you can locate better place where the accident occurred. With this prototype a solution to the aforementioned problem is presented, which after a series of tests showed that the time of sending an alert message to the central established on a LAN is less than a minute, and is determined as satisfying the prototype settled down.

ÍNDICE

INTRODUCCIÓN	1
Antecedente	1
Alcance	2
Justificación	3
Objetivo	3
Objetivos específicos	3
1. CAPÍTULO I. Marco Teórico	5
1.1 SENSORES.....	5
1.1.1 Clasificación de los Sensores	5
1.1.2 Sensores a implementar en el prototipo.....	7
1.2 MICROCONTROLADORES.....	9
1.2.1 Arduino.....	10
1.2.2 Arduino Micro	11
1.3 COMUNICACIONES INALÁMBRICAS.....	12
1.3.1 Estándar IEEE 802.....	12
1.3.2 Bluetooth	13
1.3.3 Funcionamiento de la tecnología Bluetooth	14
1.4 SISTEMAS OPERATIVOS	15
1.4.1 Sistema operativo fijo	15
1.4.2 Sistema operativo móvil	15
1.5 LENGUAJES DE PROGRAMACIÓN	16
1.5.1 Programación Orientada a Objetos	17
1.5.2 JAVA	17
1.6 BASE DE DATOS	18
1.6.1 Base de Datos Móviles.....	19
1.6.2 SQLite	19
1.7 SISTEMA CIRCULATORIO Y RITMO CARDÍACO	20
1.7.1 Sistema Circulatorio	20

1.7.2 Signos Vitales	22
1.7.3 Ritmo Cardíaco	23
1.8 FUNDAMENTOS TEÓRICOS RELACIONADOS CON ACCIDENTES DE TRÁNSITO.....	24
1.8.1 Colisión	24
1.8.2 Accidente	25
1.8.3 Casco	25
2. CAPÍTULO II. Análisis de Requerimientos del Sistema..	28
2.1 APLICACIÓN MÓVIL	28
2.2 IDENTIFICACIÓN DEL PROBLEMA, OBJETIVOS Y OPORTUNIDADES	29
2.2.1 Descripción del producto.....	29
2.2.2 Identificación del problema.....	29
2.2.3 Identificación de los objetivos.....	30
2.2.4 Análisis de oportunidad de mercado.	30
2.2.5 Análisis de riesgos.	31
2.3 Determinación de los requerimientos.....	32
2.3.1 Análisis del contexto.....	32
2.3.2 Objetos del contexto.....	33
2.3.3 Captura de contexto y protocolo.....	34
2.4 ANÁLISIS DE LAS NECESIDADES DEL SISTEMA	34
2.4.1 Casos de uso	34
2.5 SENSORES A IMPLEMENTAR	36
2.5.1 Pulso	36
2.5.2 Sonido	40
2.5.3 Aceleración	42
2.5.4 Módulo Bluetooth para Arduino	44
2.6 Diagrama de bloques del sistema.....	46
3. CAPÍTULO III. Diseño del Prototipo	48
3.1 Diseño de Hardware.....	48

3.2 DISEÑO DE HARDWARE EN EL CASCO	52
3.3 Diseño de Software	54
3.3.1 Software Arduino.....	54
3.3.2 Software Android Aplicación	58
3.3.3 Diseño de Base de Datos.....	80
4. CAPÍTULO IV. Implementación del Prototipo.....	84
4.1 Implementación del Sistema	84
4.1.1 Sensor de Pulso.....	84
4.1.2 Sensor de Sonido.....	85
4.1.3 Sensor de Aceleración	86
4.1.4 Transmisión de señal Bluetooth	86
4.1.5 Arduino micro	87
4.2 Implementación Aplicación Móvil	90
5. CAPÍTULO V. Pruebas de funcionamiento y	
Resultado	98
5.1 PULSO.....	98
5.2 SONIDO	101
5.3 ACELERACIÓN.....	103
5.4 Prueba conjunta del sistema integrado.....	105
6. Conclusiones y Recomendaciones.....	108
6.1 Conclusiones.....	108
6.2 Recomendaciones	110
Referencias.....	111
ANEXOS	115

ÍNDICE DE FIGURAS

Figura 1. Funcionamiento general de un sensor.	6
Figura 2. Clasificación general de los sensores.	7
Figura 3. Representación esquemática de un microcontrolador.	10
Figura 4. Placa Arduino.	11
Figura 5. Placa Micro Arduino.	12
Figura 6. Frecuencia de Bluetooth.	14
Figura 7. Sistema circulatorio del ser humano.	21
Figura 8. El corazón del ser humano.	21
Figura 9. Ritmo cardíaco	23
Figura 10. Ciclo de vida del desarrollo de sistemas.	28
Figura 11. Diagrama de contexto global del sistema “Casco Inteligente”.....	32
Figura 12. Diagrama de contexto del sistema “Casco Inteligente”.	33
Figura 13. Diagrama de captura de contexto y protocolo del sistema “Casco Inteligente”.....	34
Figura 14. Diagrama de casos de uso.....	35
Figura 15. Sensor KY-039.....	37
Figura 16. Sensor SEN-11574.	38
Figura 17. Diagrama sensor SEN-11574.	38
Figura 18. Sensor de Sonido HXJ-17.....	41
Figura 19. Sensor de Aceleración ADXL345.....	43
Figura 20. Sensor de Aceleración GY-88.....	43
Figura 21. Bluetooth HC-05.....	45
Figura 22. Bluetooth HC-06.....	45
Figura 23. Sistema a implementar.....	47
Figura 24. Diagrama de bloques del sistema.	47
Figura 25. Diagrama esquemático de Arduino Micro.	49
Figura 26. Diagrama de sensor de sonido conjunto a Arduino Micro.....	49
Figura 27. Diagrama de sensor de pulso conjunto a Arduino Micro.....	50
Figura 28. Diagrama de sensor de aceleración en conjunto a Arduino Micro. .	51

Figura 29. Diagrama de transmisor / receptor bluetooth en conjunto con Arduino Micro.....	51
Figura 30. Diagrama del circuito completo.....	52
Figura 31. Elementos de un casco.....	53
Figura 32. Sensor Pulso.....	55
Figura 33. Sensor Pulso Interrupt.....	56
Figura 34. Sensor Sonido.....	57
Figura 35. Sensor Aceleración.....	58
Figura 36. Activity Main.....	59
Figura 37. Subrutina Ingresar.....	60
Figura 38. Activity Usuario.....	61
Figura 39. Aceptar Nuevo Usuario.....	62
Figura 40. Activity Contacto.....	63
Figura 41. Subrutina Aceptar Contacto.....	64
Figura 42. Activity Motocicleta.....	65
Figura 43. Subrutina Botón Aceptar.....	66
Figura 44. Subrutina Botón Foto.....	67
Figura 45. Activity Foto Moto.....	68
Figura 46. Subrutina Botón Foto Nueva.....	69
Figura 47. Activity Recuperar Password.....	70
Figura 48. Subrutina Botón Aceptar Recuperar Password.....	71
Figura 49. Activity Bluetooth.....	72
Figura 50. Subrutina Enlazar.....	73
Figura 51. Activity Estado.....	74
Figura 52. Subrutina Guardar Datos Estado.....	75
Figura 53. Subrutina Comprobación Datos Estado.....	76
Figura 54. Subrutina Botón Terminar.....	77
Figura 55. Subrutina Envío Mensajes.....	78
Figura 56. Activity Principal.....	79
Figura 57. Central.....	80
Figura 58. Figura de tabla Usuario y campos respectivos.....	81
Figura 59. Figura de tabla Contacto y campos respectivos.....	82

Figura 60. Figura de tabla Moto y campos respectivos.	82
Figura 61. Figura de tabla Casco y campos respectivos.	83
Figura 62. Diagrama de base de datos móvil.	83
Figura 63. Sensor de pulso.	84
Figura 64. Sensor de sonido.	85
Figura 65. Sensor de aceleración.	86
Figura 66. Transmisor de señal Bluetooth.	87
Figura 67. Arduino Micro.	88
Figura 68. Circuito completo implementado en el casco.	90
Figura 69. Página principal de la aplicación.	91
Figura 70. Página “Usuario” de la aplicación “Casco Inteligente”.	92
Figura 71. Página “Contacto” de la aplicación “Casco Inteligente”.	93
Figura 72. Página “Motocicleta” de la aplicación “Casco Inteligente”.	93
Figura 73. Página “Foto Motocicleta” de la aplicación “Casco Inteligente”.	94
Figura 74. Página “Página Principal” de la aplicación “Casco Inteligente”.	95
Figura 75. Página “Bluetooth” de la aplicación “Casco Inteligente”.	96
Figura 76. Página “Estado” de la aplicación “Casco Inteligente”.	96
Figura 77. Central de la aplicación “Casco Inteligente”.	97
Figura 78. Prueba con la Aplicación de Sonido.	103
Figura 79. Prueba con la Aplicación de Aceleración.	105
Figura 80. Prueba de interfaz de la Central desplegando mensaje de alerta.	106
Figura 81. Prueba de Datos Almacenados.	107
Figura 82. Prueba mensaje de alerta.	107

ÍNDICE DE TABLAS

Tabla 1. Tabla descripción de estándar IEEE 802.	12
Tabla 2. Objetos del contexto del sistema "Casco Inteligente".....	33
Tabla 3. Variables útiles del sensor de Pulso.....	39
Tabla 4. Conexión del sensor de Pulso.....	85
Tabla 5. Conexión del sensor de Sonido.....	85
Tabla 6. Conexión del sensor de Aceleración.	86
Tabla 7. Conexión Bluetooth.	87
Tabla 8. Conexión Arduino Micro.	88
Tabla 9. Corriente del circuito a implementar.....	89
Tabla 10. Primera prueba sensor Pulso.....	98
Tabla 11. Segunda prueba sensor Pulso.....	99
Tabla 12. Tercera prueba sensor Pulso.....	99
Tabla 13. Cuarta prueba sensor Pulso.....	100
Tabla 14. Quinta prueba sensor Pulso.....	101
Tabla 15. Primera prueba sensor Sonido, calibrado con -40 dB.....	102
Tabla 16. Segunda prueba sensor Sonido, calibrado con -40 dB.....	102
Tabla 17. Tercera prueba sensor Sonido, valores externos al casco y sensor en el casco, calibrado con -40 dB.....	103
Tabla 18. Prueba sensor Aceleración.....	104

INTRODUCCIÓN

Antecedente

Durante los últimos años el potencial crecimiento del mercado de motocicletas en Ecuador, responde a la necesidad de los ecuatorianos por obtener soluciones de transporte eficientes y económicas. Actualmente, existen 362.357 motocicletas que se encuentran circulando (INEC, s.f., p. 51), lo que representa ventajas para la comunidad, como un menor volumen de emisiones de carbono perjudiciales a la atmósfera, descongestión vehicular y una solución eficiente para las empresas que requieren de servicios de mensajería.

Lamentablemente esta forma de transporte también presenta ciertas desventajas, una de las más representativas, el nivel de accidentes de tránsito motociclísticos que se presenta en el país, los cuales entre otras cosas, se producen debido a que la motocicleta no ofrece ninguna protección de barrera al conductor. Esto, sumado con el hecho de que los afectados no reciben atención oportuna, por no existir un sistema de alerta de que se suscitó un accidente y/o registro de la locación donde ocurrió, produce como consecuencia que muchos usuarios de esta forma de transporte pierdan sus vidas o sufran lesiones permanentes.

Adicionalmente no se tiene registros de las condiciones en las que se producen este tipo de accidentes que permitan, identificar patrones de comportamiento o de la forma de manejo del conductor, que sirvan a futuro para pensar en medidas de prevención de los mismos.

Diferentes proyectos se han especificado y desarrollado, tratando de relacionar seguridad del conductor con tecnología Arduino, en India, receptando información de casco y enviando por un sistema inalámbrico información de ubicación (Krishna, V., Praveen, K., 2013). Para sistemas de impacto, ha sido creados prototipos enfocados a deportes, al igual que sistemas para control y

monitoreo de signos vitales (Gherin, G., 2013). Otro proyecto para cascos, realizado por empresas multinacionales, no se tiene un proyecto que integre en un solo sistema los diferentes casos a efectuarse en un posible accidente (Bayly, M., Regan, M., Hosking, S., 2006).

Alcance

Este trabajo de titulación pretende crear el prototipo de un sistema que cuente con un casco capaz de detectar cuando un motociclista ha sufrido un accidente, registrar las condiciones que este ha ocurrido y alertar sobre el suceso con la finalidad de conseguir asistencia inmediata. Está pensado para la seguridad física de personas naturales o motociclistas de empresas.

El casco contará con comunicación inalámbrica a un dispositivo celular en el que habrá una aplicación móvil, la cual almacenará, controlará y recopilará información que más adelante será enviada desde el casco a una central de monitoreo.

En el casco se instalará sensores, los cuales se controlarán desde la aplicación móvil. Para esto se necesitará implementar circuitos electrónicos que hagan posible la recolección de la información, su acondicionamiento y envío inalámbrico a través de Bluetooth al dispositivo móvil, que hará posible el almacenamiento de la información en una base de datos.

Solo en caso de que exista algún tipo de accidente, la aplicación lo detectará y enviará la alerta a la central y también por medio de un mensaje a un número predeterminado. Con ello y con la ayuda de la central se podrá asistir de inmediato a dicho evento.

Para lograr el proyecto mencionado, se requiere implementar conocimientos aprendidos a lo largo de la carrera como: microprocesadores, aplicaciones móviles, electrotecnia, diseño de bases de datos, entre otras.

Justificación

Previamente se mencionó el alto número de accidentes en motociclistas. En la actualidad el aumento de los accidentes de tránsito, especialmente de los motociclistas en los que el mínimo impacto se vuelven críticos y las consecuencias de dichos accidentes son graves.

Se evidencia la necesidad de un sistema capaz de detectar cuando ocurre un accidente y alertar para prevenir la pérdida de vidas.

Por lo que se plantea el diseño e implementación de este sistema es para disminuir el tiempo de respuesta a una alerta de accidente y también para dar más seguridad a los motociclistas. Se ve la necesidad imperiosa de implementar un casco inteligente que nos brinde mayor seguridad, nos ayude a prevenir y el cual permita monitorear y comunicar al instante posibles accidentes. Al lograr tener una comunicación inmediata, se disminuiría el tiempo de acción en caso de un accidente, siendo el principal objetivo del prototipo a diseñar.

Objetivo General

Implementar un prototipo de un sistema que cuente con un casco inteligente capaz de recolectar información sobre condiciones de manejo de motociclistas y una aplicación móvil capaz de almacenar los datos recolectados, detectar accidentes y emitir alertas en caso de que estos sucedan.

Objetivos específicos

- ❖ Analizar las características técnicas de los diferentes sensores a utilizar, aplicación móvil y conexión mediante red inalámbrica para el sistema propuesto.

- ❖ Diseñar el prototipo de un casco que cuente con sensores para tener información real en caso de accidente, además de una base de datos para la recolección de la información de los parámetros de los sensores y de la información del usuario.

- ❖ Crear una aplicación móvil la cual permitirá dar comienzo a la aplicación cuando el motorista emprenda su viaje y de igual manera cuando se termine el mencionado viaje, adicionalmente constará de una recolección de datos que a su vez será actualizada cada cierto tiempo para tener en caso de un accidente una información acertada de lo que sucedió y enviará a un contacto o central predeterminada.

- ❖ Implementar la comunicación inalámbrica entre el casco y el dispositivo móvil por medio de bluetooth.

- ❖ Recolectar datos y resultados de las pruebas correspondientes del prototipo.

1. CAPÍTULO I. Marco Teórico

Este proyecto pretende implementar un prototipo de un sistema que cuente con un casco para detectar cuando un motociclista sufre un accidente con la finalidad de que sea asistido inmediatamente. El sistema consistirá en un casco con sensores para identificar diferentes variables y comunicación inalámbrica a un dispositivo celular, en el cual existirá una aplicación móvil. En esa aplicación móvil se guardarán los datos y en caso de emergencia se enviará una señal de alerta.

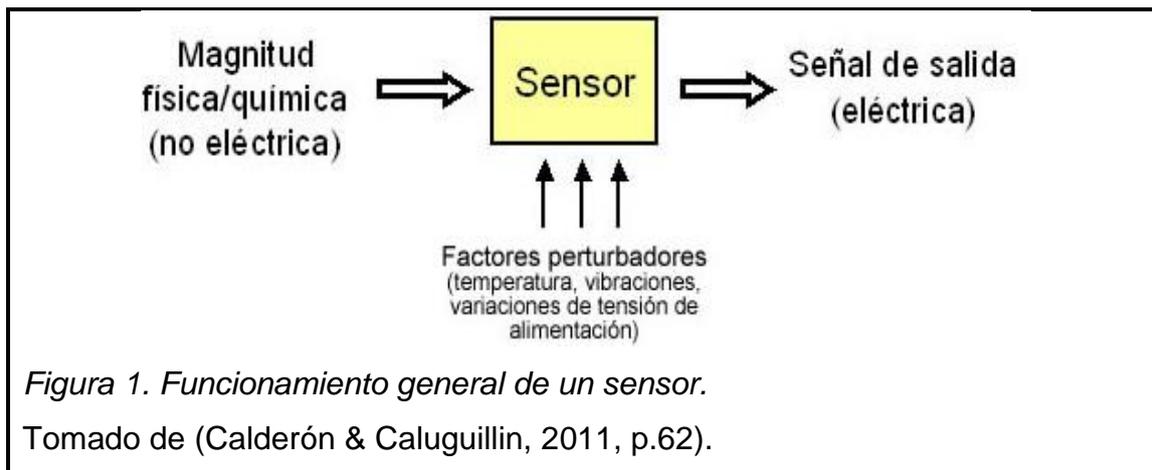
1.1 SENSORES

Un sensor es un dispositivo que está diseñado para recibir información de una magnitud y convertirla en otra, es decir, transforma una señal ya sea aceleración, temperatura, presión, pulso, etc. en una señal eléctrica, aprovechan las propiedades de algunos materiales de los cuales se convierten en generadores de señal en presencia de determinadas excitaciones, (Avendaño, 2010, p.10), la cual es detectada por un sistema de control, como se muestra en la Figura 1.

“Un sensor es un tipo de transductor que transforma la magnitud que se quiere medir o controlar, en otra, que facilita se medida. Pueden ser de indicación directa (un termómetro de mercurio) o estar conectados a un indicador (a través de un convertidor analógico a digital, un computador y un visualizador) de modo que los valores detectados puedan ser leídos por un humano”. (Cordero, 2014, p. 12).

1.1.1 Clasificación de los Sensores

En la actualidad existen varios tipos de sensores, en la Figura 2 se detalla una clasificación general. (Olivia, 2010, p. 1-12.).



Según el principio de funcionamiento

- ❖ **Activos:** Funcionan como generador, los principios básicos que se basan son los efectos termoeléctrico, fotoeléctrico, peltier, etc.
- ❖ **Pasivos:** Funcionan como modulador, la impedancia de dichos sensores es sensible a variaciones de la magnitud de medida. La característica eléctrica sensible es la resistividad, constante dieléctrica, permeabilidad magnética.

Según el tipo de señal eléctrica que generan

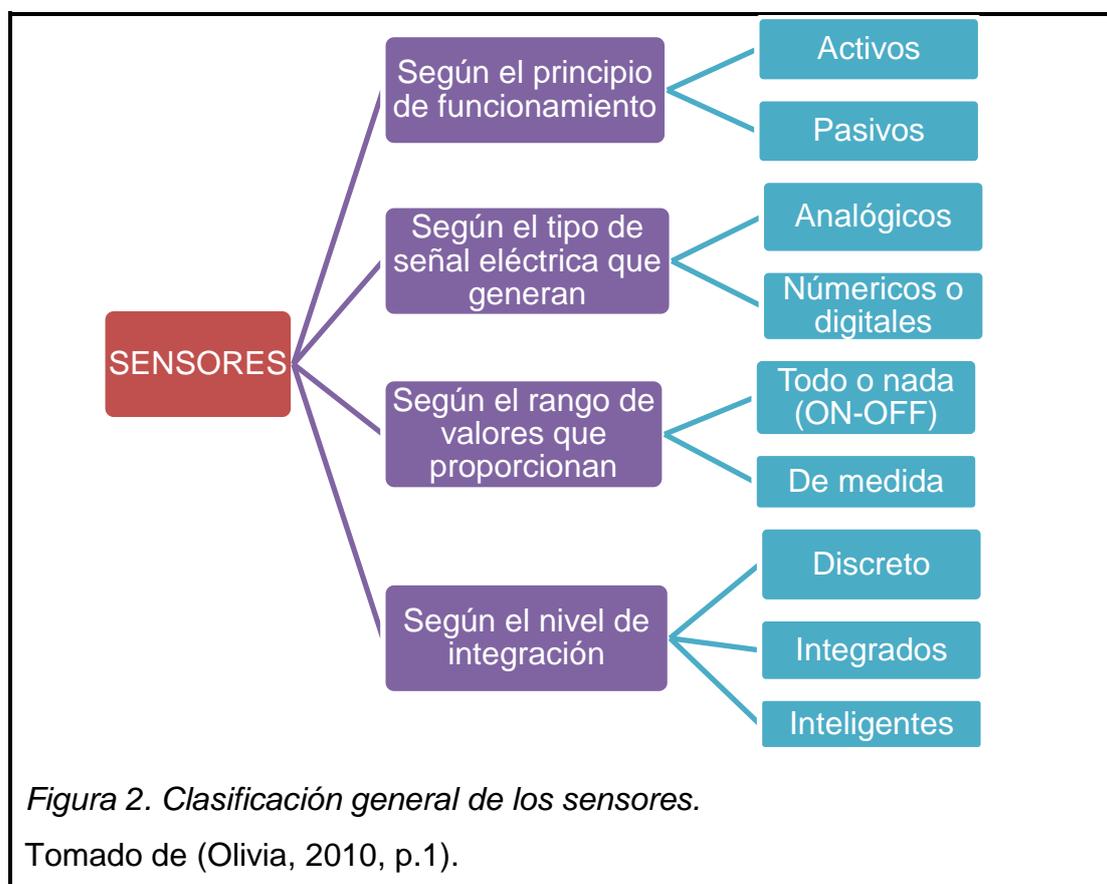
- ❖ **Sensores analógicos:** La salida varía de forma continua. La información está en la amplitud. Se denomina cuasi-digitales cuando la información está en la frecuencia.
- ❖ **Sensores numéricos o digitales:** La salida varía en pasos discretos.

Según el rango de valores que proporcionan

- ❖ **Sensor todo o nada:** Detecta la magnitud de entrada, si está por encima o debajo de un determinado valor.
- ❖ **Sensor de medida:** Proporciona la salida de los valores correspondientes a cada valor de la variable.

Según el nivel de integración

- ❖ **Sensor discreto:** El circuito de acondicionamiento se realiza mediante componentes electrónicos separados e interconectados entre sí.
- ❖ **Sensor integrado:** Elemento sensor y circuito acondicionador.
- ❖ **Sensor inteligente:** Realiza una de las funciones de cálculos numéricos, comunicación en red, etc.



1.1.2 Sensores a implementar en el prototipo

SENSOR PULSO

El sensor de pulso o más conocido como sensor de frecuencia cardíaca o detector de latido del corazón. Se utiliza para monitorear el número de latidos

por minuto, es capaz de monitorear el tejido vascular a través de varios sitios en el cuerpo humano, como por ejemplo: dedo, carotideo, oreja, etc., permitiendo medir de manera precisa las pulsaciones a través de la detección de la cantidad de oxígeno en la sangre. (Ventura, 2015, p.2).

“Los leds emiten dos longitudes de onda específicos, rojos 660nm e infrarrojo de 905nm; estas ondas de luz son emitidas a través del tejido del paciente. Se encienden y apagan a través de pulsos secuenciales, así el fotodetector es sincronizado para hacer una lectura simultánea de la transmisión de las longitudes de onda”. (Ronquillo y Salgado, 2013, p.10).

SENSOR SONIDO

El sensor de sonido es un dispositivo que detecta decibeles (dB), un decibel es una medida de presión del sonido. Dicho sensor trabaja con un pequeño micrófono que está diseñado para demostrar las formas de onda de audio, proporcionando una manera fácil de detectar el sonido. Puede ser utilizado para aplicaciones de seguridad y monitoreo, ya que su precisión es regulable. (Jiankai, 2015, p. 5).

SENSOR DE ACELERACIÓN

Es un dispositivo que mide la aceleración, que es la tasa de cambio de velocidad de un objeto. Se mide en metros por segundo al cuadrado ($\frac{m}{s^2}$). Acelerómetro de tres ejes con medición de alta resolución (13 bits) en un máximo de ± 16 g. de datos de salida digital. Con formato de dos en dos de 16 bits que se complementan y a su vez es accesible a través de interfaz digital I2C o un SPI (3 o 4 hilos). Dicho sensor es muy útil para aplicaciones móviles. La actividad y la inactividad de detección detectan la presencia o ausencia de movimiento mediante la comparación de la aceleración en cualquier eje con los umbrales establecidos por el usuario. (Earl, s.f., p. 4-8).

1.2 MICROCONTROLADORES

Un microcontrolador es un circuito integrado programable que está adaptado en un chip. Dispone de las siguientes características: unidades de memoria para almacenar datos del programa a ejecutarse; unidad aritmética lógica (ALU) encargada de realizar operaciones aritméticas (suma, resta, multiplicación) y lógicas (and, or, or exclusivo); las unidades de entrada y salida para la comunicación que dispone el microcontrolador para recibir o enviar datos en forma de serie o paralelo; temporizadores que sirven para las operaciones del programa con un reloj interno; y con controlador de interrupciones. Se representa esquemáticamente en la Figura 3. (Ceduvirt, s.f., p. 2-3).

“Los microcontroladores son computadores digitales integrados en un chip que cuentan con un microprocesador o unidad de procesamiento central (CPU), una memoria para almacenar el programa, una memoria para almacenar datos y puertos de entrada salida”. (Torres, 2007, p.11).

El funcionamiento de un microcontrolador dispone de un conjunto de instrucciones las cuales son traducidas a un lenguaje de máquina (0 y 1), por un programa llamado ensamblador. Un microcontrolador puede manipular un sistema específico, mediante lenguajes de programación y tiene unidades funcionales de una computadora como: Periféricos de entrada y salida, Memoria y Unidad central de procesamiento.



1.2.1 Arduino

Según Astudillo (2014, p. 1-5), ya lo dijo Arduino es una plataforma de prototipos electrónicos con desarrollo de código abierto, en hardware y software flexibles y de fácil uso, siendo una herramienta para que los ordenadores puedan controlar un entorno físico a través de un computador personal. Arduino puede trabajar mediante la recepción de entradas de cualquier tipo de sensor y lo convierte en una salida. Los proyectos en Arduino son autónomos o pueden comunicarse con un programa (software) que se ejecute en un ordenador. Se utiliza el lenguaje de programación de Arduino (basado en el cableado), y el software de Arduino (IDE). El software se puede descargar gratuitamente de la página oficial Arduino.

Usos para Arduino:

- ❖ Crear objetos interactivos.
- ❖ Leer datos de una gran variedad de interruptores y sensores.
- ❖ Controlar tipos de luces, motores y otros actuadores físicos.

Arduino simplifica el proceso de trabajo con microcontroladores, además que ofrece ventajas con respecto a otros sistemas. Puede ser alimentado a través

de la conexión USB o con una fuente de alimentación externa. La placa Arduino Uno está compuesta como se demuestra en la Figura 4 y es una de las más usadas para los proyectos grandes o que requiera de gran capacidad.

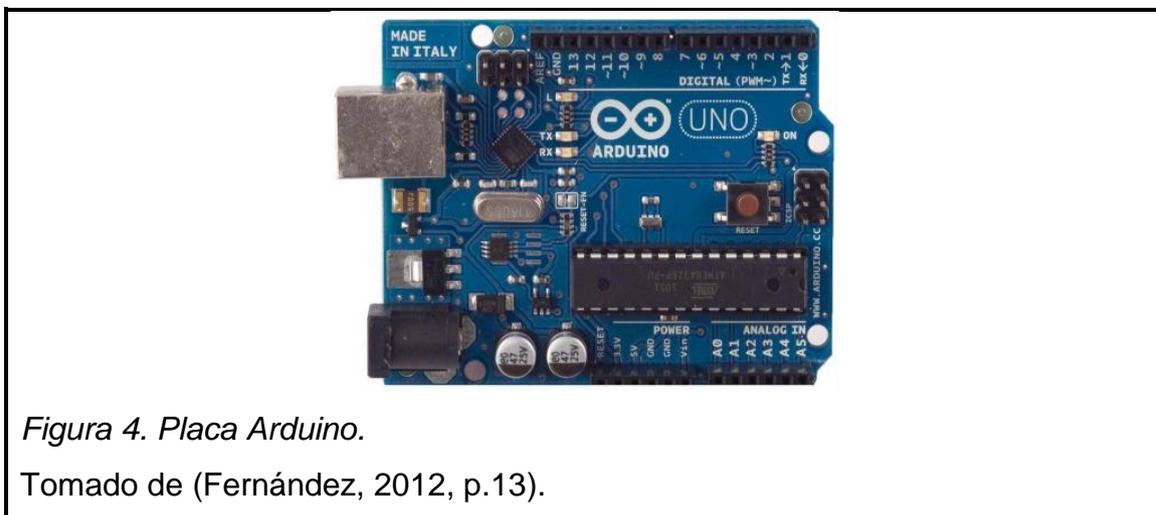


Figura 4. Placa Arduino.

Tomado de (Fernández, 2012, p.13).

1.2.2 Arduino Micro

Un microarduino es una placa electrónica basada en el ATmega32U4, desarrollado en conjunto con Adafruit. Cuenta con 20 pines digitales de entrada / salida (de los cuales 7 se pueden utilizar como salidas PWM y 12 entradas analógicas), como un oscilador de cristal de 16 MHz, una conexión micro USB, una cabecera ICSP, y un botón de reinicio. Contiene todo lo necesario para apoyar el micro controlador; basta con conectarlo a un ordenador con un cable micro USB para empezar. Tiene un factor de forma que le permite ser fácilmente colocado en una placa.

Una de las mayores ventajas de este chip es que dispone de un puerto USB nativo que permite entre otras cosas evitar tener un conversor serie/USB y además de poder programar la placa como un dispositivo USB cliente como un teclado o un ratón.

La placa incluye un conector micro USB, un puerto ICSP, un botón de reset y algunos diodos LED de estado, como se muestra en Figura 5.

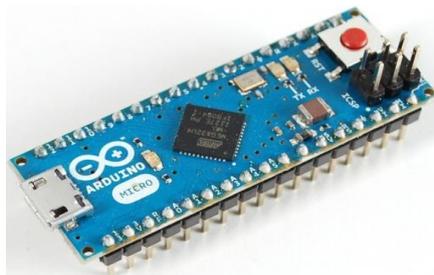


Figura 5. Placa Micro Arduino.

Tomado de (Arduino, s.f).

1.3 COMUNICACIONES INALÁMBRICAS

La comunicación inalámbrica es aquella comunicación entre dispositivos que intercambian información sin el uso de cables de interconexión. Una ventaja de dicha tecnología es por su movilidad.

1.3.1 Estándar IEEE 802

El estándar IEEE por sus siglas en inglés “Institute of Electrical and Electronics Engineers”, ha desarrollado varias normas para redes locales (LANs) y metropolitanas (MANs). Se desarrolló con el fin de crear estándares para que diferentes tipos de tecnologías trabajen juntas y a su vez integrarse. El estándar IEEE 802 se relaciona con la transmisión de datos y el cableado físico. Los grupos de trabajo del estándar IEEE 802 son como se muestra en la Tabla 1.

Tabla 1. Tabla descripción de estándar IEEE 802.

ESTÁNDAR	NOMBRE DEL ESTÁNDAR	DEFINICIÓN
802.1	Protocolos superiores de redes de área local.	Autenticación de dispositivos conectados a un puerto LAN.
802.2	Control de enlace lógico.	La forma en que los datos son transmitidos por un medio físico.
802.3	Ethernet	No transfiere más de una información a la vez, es decir no pierde dicha información y es conocido como CSMA/CD.
802.4	Token Bus	Esquemas de red de gran ancho de banda.
802.5	Token Ring	Acceso de paso de tokens.

802.6	Red de área metropolitana (MAN)	Provee servicios de datos, voz y video en una área de 50km.
802.7	Grupo de asesoría técnica sobre Banda ancha	Redes con mayor ancho de banda usando cable coaxial.
802.8	Grupo de asesoría técnica sobre Fibra óptica	Provee consejo a otros subcomités en redes por fibra óptica como alternativa a las redes basadas en cable de cobre.
802.9	RAL o LAN de servicios integrados	Provee flujo multiplexado que lleva canales de información de datos y voz conectado a un cable de cobre.
802.10	Seguridad inter operable en RAL o LAN	Modelo de seguridad que opera sobre varias redes incorporando métodos de autenticación y encriptamiento.
802.11	Red local inalámbrica (WIFI)	En el enfoque distribuido, cada estación de trabajo controla su acceso a la red.
802.12	Prioridad de demanda	Define el estándar Ethernet de 100 Mbits/seg.
802.14	Cable módems	Transporta información sobre cable Coaxial/ Fibra Óptica de TV.
802.15	Red de área personal inalámbrica (Bluetooth)	Permiten que los dispositivos inalámbricos puedan comunicarse e inter operar uno con el otro en corta distancia.
802.16	Acceso inalámbrico de banda ancha (WiMAX)	Provee conectividad fija en redes MAN y velocidad hasta 75MB/seg.

1.3.2 Bluetooth

Define un estándar de la norma global de comunicación inalámbrica IEEE 802.15, que está enfocada básicamente al desarrollo de estándares para redes WPAN o redes LAN. Ayuda a la transmisión entre diferentes equipos mediante un enlace de radiofrecuencia.

Está basado en la tecnología inalámbrica como spread spectrum (espectro esparcido), en concreto en frequency hopping (salto de frecuencia): estos sistemas de frequency hopping dividen la banda de frecuencia en varios canales de salto (hopping); en el transcurso de la conexión se produce una transición brusca (salto o hopping) de un canal a otro de forma pseudo aleatoria. (Garin y Hazard, 2013, p. 3).

Objetivos de Bluetooth

- ❖ Facilitar las comunicaciones entre equipos móviles y fijos.
- ❖ Eliminar cables y conectores.
- ❖ Reducir la potencia consumida.

- ❖ Ofrecer la posibilidad de crear pequeñas redes inalámbricas.

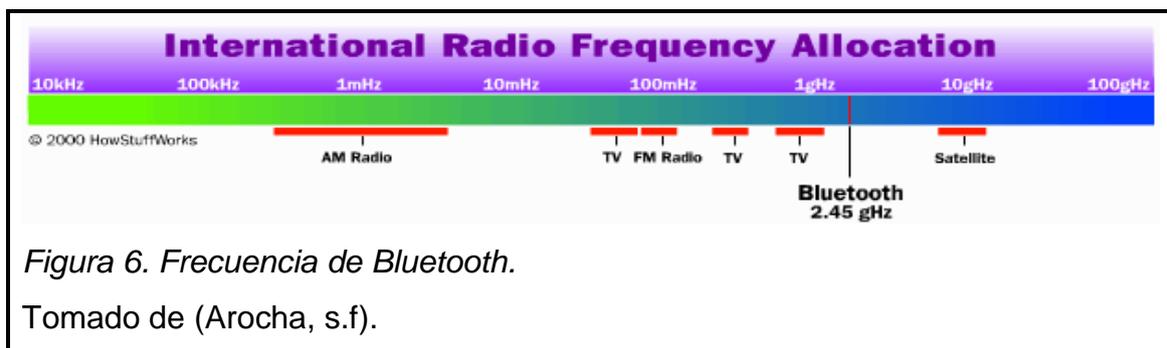
Bluetooth provee una conexión inalámbrica y de fácil uso. Es un estándar empleado para enlaces de corto alcance. Dicha tecnología permite a los usuarios conexiones instantáneas de voz y datos entre varios dispositivos y lo más importante en tiempo real. Este modo de transmisión tiene una protección contra cualquier tipo de interferencia y seguridad en los envíos de datos.

1.3.3 Funcionamiento de la tecnología Bluetooth

Los dispositivos están compuestos por dos partes:

- ❖ Dispositivo de radio encargado de modular y transmitir la señal.
- ❖ Un controlador digital

Es una tecnología de radio frecuencia que utiliza la banda ISA de 2.5 GHz. Como se muestra en la Figura 6.



Según (IEEE, 2005), la estructuración de los paquetes en una comunicación Bluetooth debe cumplir con las características que se menciona a continuación.

- ❖ Máximo de 5 espacios de tiempo.
- ❖ Los datos en un paquete tiene un máximo de 2745 bits.
- ❖ Existen dos tipos de transferencia de datos entre dispositivos SCO (conexión orientada síncrona) son los paquetes que se envían a intervalos regulares y nunca se vuelve a transmitir.

ACL (conexión mínima asíncrona) los paquetes se retransmiten en caso de error.

1.4 SISTEMAS OPERATIVOS

Un sistema operativo es un software cuya labor es administrar todos los dispositivos de una computadora, y a su vez proporcionar una interfaz sencilla para los programas de usuario, con ello poder comunicarse con el hardware. (Maya, p. 13).

1.4.1 Sistema operativo fijo

Un sistema operativo fijo está diseñado para trabajar en equipos pequeños, como por ejemplo dispositivos electrónicos autónomos con autonomía reducida o los PDA (Asistentes Personales Digitales). Una de las características principales y más importantes, es la avanzada administración de energía y la capacidad de funcionar con recursos limitados. (CCM, s.f.).

Los principales sistemas fijos para los PDA son:

- ❖ PalmOS
- ❖ Windows CE/ Windows Mobile/ Windows Smartphone
- ❖ Windows, Linux, MacOS.

1.4.2 Sistema operativo móvil

Un sistema operativo móvil es la unión de varios elementos que permiten operar un objeto. Controla un dispositivo móvil, computadoras que ocupan Windows, Mac Os, Linux, etc., dichos sistemas son simples y están enfocados a la conectividad y transmisión de datos inalámbricamente. (Pedrozo, p. 3).

Características principales:

- ❖ Núcleo.- El núcleo es el que permite el acceso a los elementos físicos del dispositivo, se lo puede relacionar como el procesador principal, y el que permite gestionar procesos, ingresar al sistema de archivos, gestionar la memoria, entre otras opciones. También es conocido como Kernel.
- ❖ Middleware.- Es una plataforma lógica la cual permite establecer comunicación o interacción entre elementos físicos o lógicos que se encuentran establecidos en el sistema o dispositivo, además de ser transparente para los usuarios. Su función es lograr establecer servicios para que las aplicaciones puedan relacionarse con el hardware.
- ❖ Interfaz de usuario.- La interfaz de usuario es la que facilita la interacción de las aplicaciones con el usuario, la cual presenta visualmente la aplicación, el cual contiene los componentes gráficos y la presentación de interacción.
- ❖ Entorno de ejecución.- El entorno de ejecución está relacionado con un gestor de aplicaciones e interfaces para permitir al desarrollador probar y programar el software, y que pueda ejecutarse la aplicación correctamente.

1.5 LENGUAJES DE PROGRAMACIÓN

Lenguaje de programación, se refiere a idiomas creados y diseñados para realizar procesos o cálculos lo cuales pueden ser entendidos por máquinas como computadores. Al ser un pseudo-idioma, está conformado por un conjunto de palabras, símbolos y reglas de sintaxis y semántica que se deben acoger para que tengan estructura y significado, como cualquier otro lenguaje. (Barber y Ferrís, p.9).

Los lenguajes de programación se pueden clasificar en diferentes macro grupos.

- ❖ **Lenguajes de bajo nivel.** Son los lenguajes que dependen de la máquina la cual está siendo programada, por ende no se puede utilizar en otros dispositivos. Este lenguaje es diseñado a medida de programación hacia el hardware, y es utilizado para aprovechar al máximo las características de los elementos.

Lenguaje máquina. El lenguaje máquina fue creado para ordenar a los dispositivos como realizar las funciones fundamentales en un lenguaje que se pueda acoplar con los procesos necesarios para la máquina y humanos, la cual consiste en la combinación de dígitos binarios.

Lenguaje ensamblador. El lenguaje ensamblador es derivado del lenguaje máquina, pero con una mejora la que permite acoplar abreviaturas en letras y números para poder facilitar la utilización de estos mismos a los programadores. Se utiliza una plataforma para convertir el código ensamblador a código máquina para que puedan ser procesadas las instrucciones programadas.

1.5.1 Programación Orientada a Objetos

“La programación Orientada a Objetos es una metodología que basa la estructura de los programas en torno a los objetos.

Los lenguajes de POO ofrecen medios y herramientas para describir los objetos manipulados por un programa. Más que describir cada objeto individualmente, estos lenguajes proveen una construcción (Clase) que describe a un conjunto de objetos que poseen las mismas propiedades”. (Carballo, 2007, p. 1).

1.5.2 JAVA

JAVA es un lenguaje compilado e interpretado. Además es un lenguaje orientado a objetos que puede realizar diferentes proyectos. Tiene una

semejanza en la sintaxis del lenguaje de programación C y C++. El lenguaje JAVA comienza con paquetes, que es el mecanismo de espacio de nombres del lenguaje Java, dentro de estos paquetes se encuentran las clases y dentro las clases se encuentran métodos, variables, constantes, etc. (Steven, 2012, p. 1-3).

La seguridad en JAVA es lo más primordial, tanto en el programador como en la ejecución de la máquina virtual. Los criterios fundamentales de Java son: independencia de la máquina, seguridad de trabajo en red y potencia para sustituir código nativo.

La herramienta de desarrollo se conoce como JDK, por sus siglas en inglés (Java Development Kit). La herramienta es gratuita y se puede descargar de la página oficial de JAVA. (Steven, 2012, p. 3-5). Dicha herramienta cuenta con:

- ❖ Compilador de línea de comandos “javac”
- ❖ Máquina virtual de JAVA para ejecutar aplicaciones “java”
- ❖ Herramienta de documentación “javadoc”
- ❖ Herramienta para empaquetar proyectos “jar”

1.6 BASE DE DATOS

Es un sistema informático que sirve como almacén, dicho almacén podrá guardar varios volúmenes de información. Son recursos que recopilan información para atender las necesidades de los usuarios. En la actualidad se necesita una base de datos para automatizar todo el acceso de la información y así poder acceder más rápido, con mayor facilidad de encontrar información relevante y realizar cambios de forma eficiente. (Sierra, 2012, p. 1-2).

Toda base de datos debe constar con las siguientes características:

- ❖ Seguridad y Privacidad
- ❖ Integridad
- ❖ Independencia

- ❖ Consistente
- ❖ Redundancia
- ❖ Flexibilidad
- ❖ Monitoreo

La clasificación de la base de datos son estáticas y dinámicas. Las estáticas solo servirán de lectura y almacenamiento, las dinámicas permiten que la información se pueda modificar o actualizar.

1.6.1 Base de Datos Móviles

Durante los últimos años el uso de teléfonos móviles, PDA y computadoras portátiles han aumentado considerablemente, por ello se requirió el uso de la base de datos móvil. Una base de datos móvil se puede instalar en un dispositivo móvil a través de una red del mismo tipo.

“El cliente y el servidor tienen conexiones inalámbricas, la memoria caché se mantiene para almacenar los datos frecuentes y transacciones de manera que no se pierdan debido a un fallo de conexión.

Una base de datos es una forma estructurada de organizar la información, lo cual podría ser una lista de contactos, información de precios o de la distancia recorrida”. (Rivero, Pérez, Vila, 2013, p. 3).

1.6.2 SQLite

“SQLite es un motor de base de datos SQL incorporado. A diferencia de la mayoría de las otras bases de datos SQL, SQLite no tiene un proceso de servidor independiente, lee y escribe directamente en archivos de disco ordinarios”. (D. Richard Hipp).

Es de dominio público e implementa una librería de 500KB programada en lenguaje, a su vez funciona como sistema de gestión de base de datos relacionales. Una de las funciones de SQLite es que no permite concurrencia de conexiones, por lo tanto es monousuario. (SQLite, s.f.).

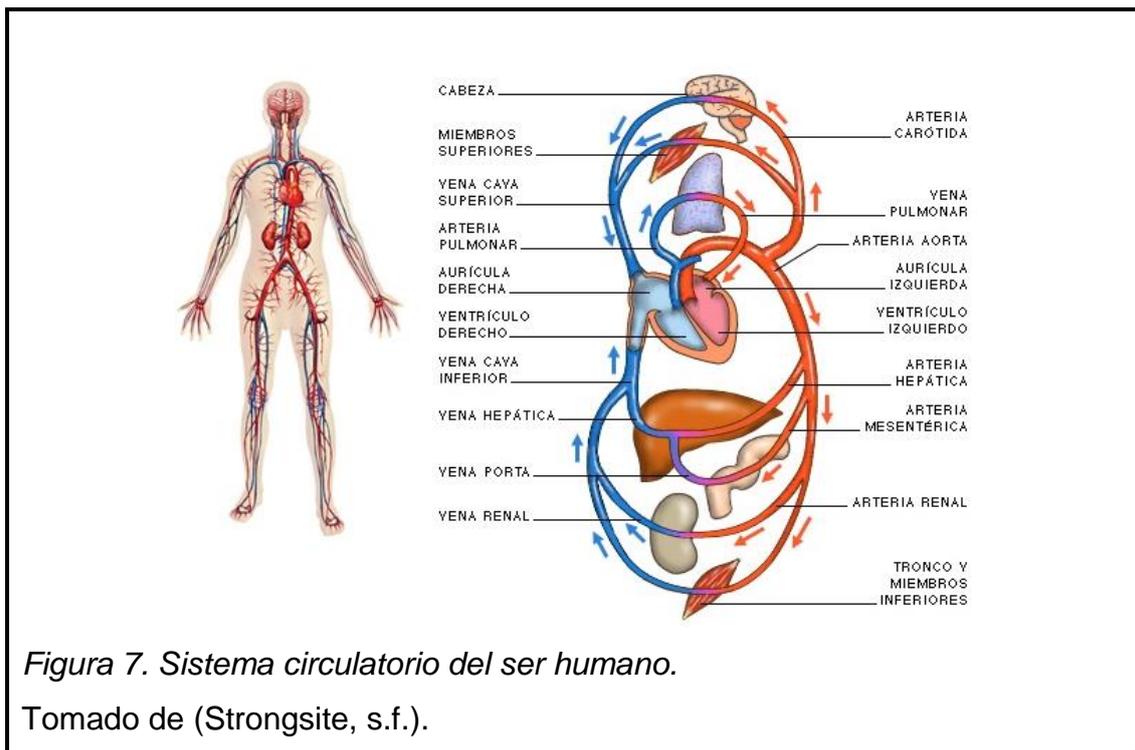
1.7 SISTEMA CIRCULATORIO Y RITMO CARDÍACO

1.7.1 Sistema Circulatorio

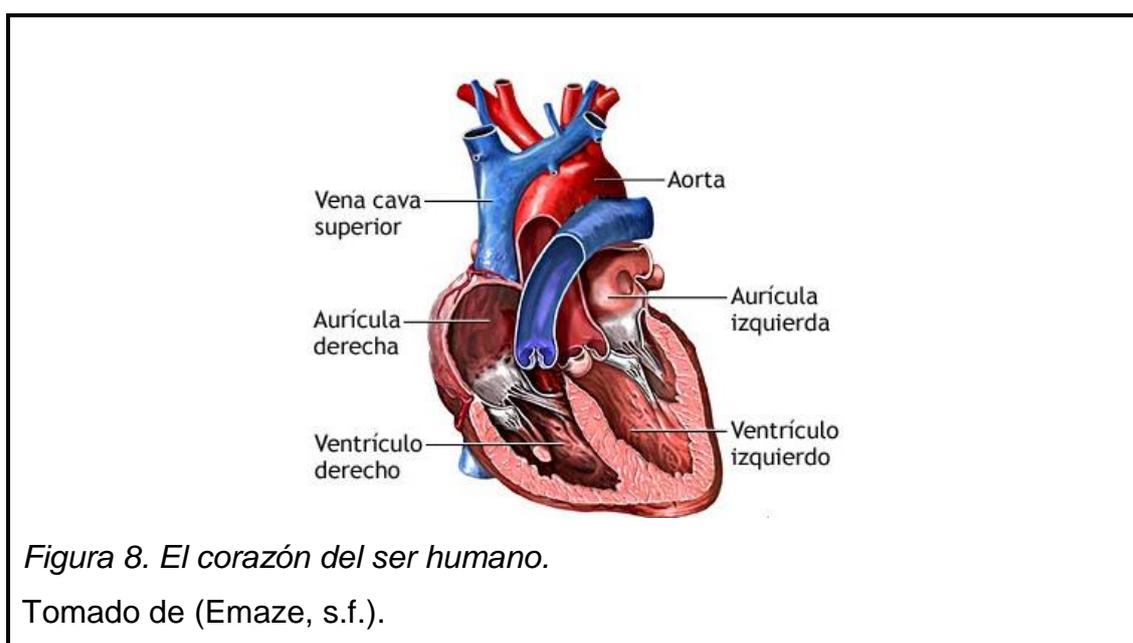
El sistema circulatorio en el ser humano es unidireccional, su función principal es la de transportar diferentes sustancias tales como nutrientes, oxígeno, dióxido de carbono y residuos. Este sistema está compuesto por varios elementos como se muestra en la Figura 7.

El corazón es el principal órgano del sistema circulatorio, y el eje principal del cuerpo. Tiene forma de un cono truncado con un peso aproximado de 342 gramos, como se muestra en la Figura 8.

El corazón se encuentra entre los pulmones, detrás del esternón, y se puede encontrar una membrana de dos capas llamadas pericardio, la cual envuelve y protege al corazón.



El corazón late más de diez mil veces diarias para bombear 3874 litros de sangre por el cuerpo. Al ser la sangre un tejido líquido, se compone de agua, sustancias disueltas y células sanguíneas. Los glóbulos rojos se encargan de la distribución del oxígeno, los glóbulos blancos realizan trabajos de limpieza y defensa, y las plaquetas permiten la coagulación de la sangre.



El corazón está formado por dos bombas separadas por una pared muscular denominada tabique. La bomba derecha, formado por una aurícula derecha y ventrículo derecho, recibe la sangre desoxigenada y envía a los pulmones. La bomba izquierda, formado por una aurícula izquierdo y ventrículo izquierdo, reciben la sangre oxigenada de los pulmones y la envía a todo el cuerpo.

La aurícula, también denominada atrium, designa la cavidad superior del músculo cardíaco que recoge la sangre. El corazón tiene habitualmente dos aurículas cuya misión principal es la de favorecer el transporte de la circulación sanguínea desde las venas cavas y pulmonares hacia los ventrículos. (Aragoncillo, p. 35-37).

1.7.2 Signos Vitales

Se denominan signos vitales a las reacciones que presenta el ser humano vivo a los siguientes puntos: Ritmo cardíaco presente, presión arterial, temperatura, frecuencia respiratoria y reflejo pupilar.

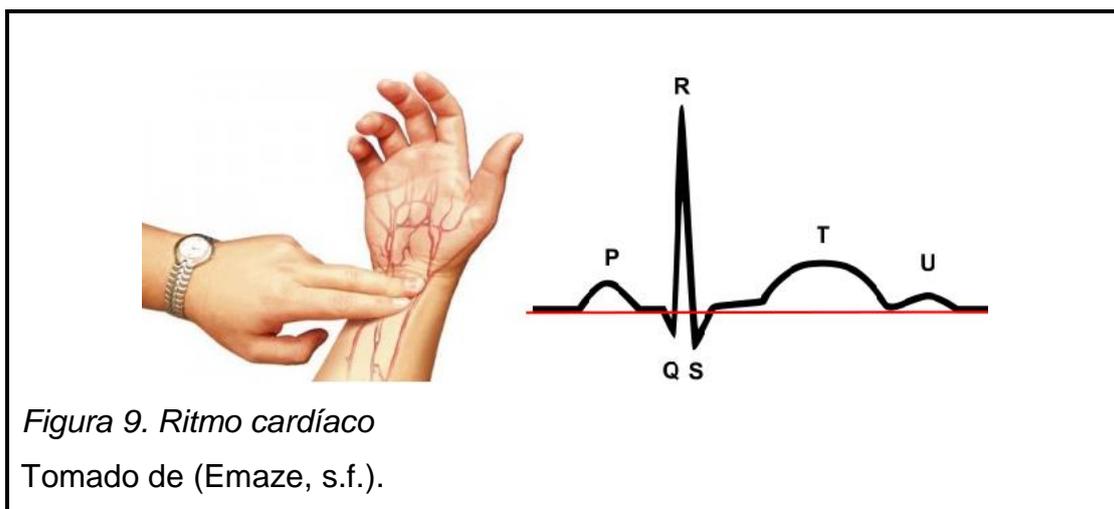
Los rangos normales para un ser humano adulto son:

- ❖ 36.0 a 37.5 grados centígrados de temperatura del cuerpo.
- ❖ 15 a 20 respiraciones por minuto.
- ❖ 60 a 80 latidos por minuto.
- ❖ 90/60 mmHg hasta 120/80 mmHg de presión arterial.

La medición de los signos vitales permiten llevar un monitoreo de su estado al momento de sufrir accidentes, y permiten determinar a la persona tratante que métodos poder utilizar para poder realizar tratamientos óptimos según su diagnóstico de salud. (Aguayo y Lagos, p. 1).

1.7.3 Ritmo Cardíaco

El ritmo cardíaco es el periodo armónico de los latidos cardíacos generados por el bombeo de sangre del corazón. El pulso se siente a través de los vasos sanguíneos ubicados en la superficie de la piel como en el brazo, cuello, muñeca, etc., como se muestra en la Figura 9.



Componentes del ritmo cardíaco

El corazón late durante sístole, que representa a la contracción de los músculos del corazón para bombear la sangre, y durante la diástole cuando se relajan los músculos del corazón para volver a recuperar sangre y contraerse nuevamente. (Villalón y López, s.f., p. 597)

Al generarse estas etapas, se puede determinar el ritmo cardíaco según la frecuencia de contracciones que genera el corazón durante un periodo de tiempo y así determinar si existen signos vitales.

Factores que alteran el ritmo cardíaco

- ❖ Edad.
- ❖ Ejercicio físico.
- ❖ Temperatura

- ❖ Estado emocional
- ❖ Posición del cuerpo
- ❖ Uso de medicamentos

Tipos de medición

Existen varias formas de medir el ritmo cardíaco, se puede realizar de forma manual o con aparatos electrónicos. De forma manual se necesita colocar el dedo índice y dedo medio en ciertas áreas del cuerpo en donde una arteria pase cerca de la piel con la finalidad de sentir su movimiento debido a la circulación de la sangre, existen varios puntos donde se puede tomar el pulso y encontrar la arteria, entre ellos:

- ❖ Pulso carotídeo (cuello)
- ❖ Pulso radial (muñeca)
- ❖ Pulso Femoral (ingle)
- ❖ Pulso temporal (sien)
- ❖ Pulso cubital (parte interna del codo)
- ❖ Pulso poplíteo (parte posterior de la rodilla)
- ❖ Pulso medio (cara interna del pie)
- ❖ Ritmo cardíaco del oído

1.8 FUNDAMENTOS TEÓRICOS RELACIONADOS CON ACCIDENTES DE TRÁNSITO

1.8.1 Colisión

Según la RAE (s.f.), “es el choque de dos cuerpos”. Según Wilson (1996), Es la aceleración o desaceleración de un cuerpo, causado por un impacto, que generara un impulso creando una fuerza, según la segunda ley de Newton, el cual se transforma a energía cinética.

Choque elástico, es la cual al golpear a una masa estacionaria, la transferencia de energía es total, pero la masa que recibe el impulso continuará la trayectoria a tomar.

Choque inelástico, es el cual al golpear una masa estacionaria, la transferencia de energía es total, y la masa que recibe el impulso queda pegado con la masa que generó el choque después del impacto.

1.8.2 Accidente

Según RAE (s.f.), “Suceso eventual que altera el orden regular de las cosas”, “Suceso eventual o acción de que resulta daño involuntario para las personas o cosas”. En contexto, un suceso eventual, que alteran o dañan a objetos o personas.

Accidente de tránsito

Un accidente de tránsito son aquellos sucesos no esperados ocurridos en una vía pública, en el cual se encuentra involucrado uno o más vehículos que transitan por aquella vía, en la cual se pueden ver afectados peatones, vehículos y otros elementos.

Accidente de tránsito de motocicletas

Los accidentes de tránsito de motocicletas, al igual que un accidente de tránsito, es el que involucra a uno o más vehículos que transitan por la vía pública, pero se incluye al menos una motocicleta en el incidente.

1.8.3 Casco

El casco es un objeto utilizado por motociclistas para protegerse de accidentes o eventos, diseñado para cubrir la zona craneal y facial. Estos cascos se

encuentran elaborados de materiales resistentes como plástico, fibra de vidrio, entre otros. El objetivo del mismo es disminuir los daños por colisiones.

Tipos de Cascos

Los diferentes cascos creados, permiten al usuario tener mayor protección en caso de algún incidente, y proporcionar mayor seguridad. En la actualidad, existen varios diseños de cascos, dividiendo en dos diferentes grupos.

Casco abiertos

Calimero: Es un casco que permite la protección únicamente de la zona craneal, pero deja expuesto el rostro, la nuca y los laterales de la cabeza.

Semi Jet: Es un casco que tiene mayores protecciones, ya que protege los laterales de la cabeza, y generalmente encuentran con una pantalla pequeña de fibra de carbono que protege el rostro.

Modulares: Es muy similar a los cascos integrales, con la opción de remover la parte frontal. El riesgo de este tipo de cascos se presenta al extraer la parte frontal ya que el rostro se encuentra descubierto.

Casco cerrados

Integrales: Son cascos con el diseño más seguro, ya que cubren toda la cabeza, y cuentan con pantalla de fibra de carbono, que protege a todo el rostro, además de su diseño de protección en la quijada y nuca.

Homologación del casco

El uso del casco es obligatorio cuando se circula en motocicleta por la vía pública, es muy importante para la seguridad del motociclista; por lo tanto, el casco debe cumplir con una serie de normas para poder circular sin ningún

inconveniente. La homologación es diferente en cada país, a continuación se detallará las homologaciones internacionales más exigentes. Se trata de la homologación Snell USA y DOT Certified que pertenecen a Estados Unidos, homologación ECE/ONU 22.05 a Europa, en Ecuador sirve la homologación DOT y ECE/ONU. (Euromoto, s.f.).

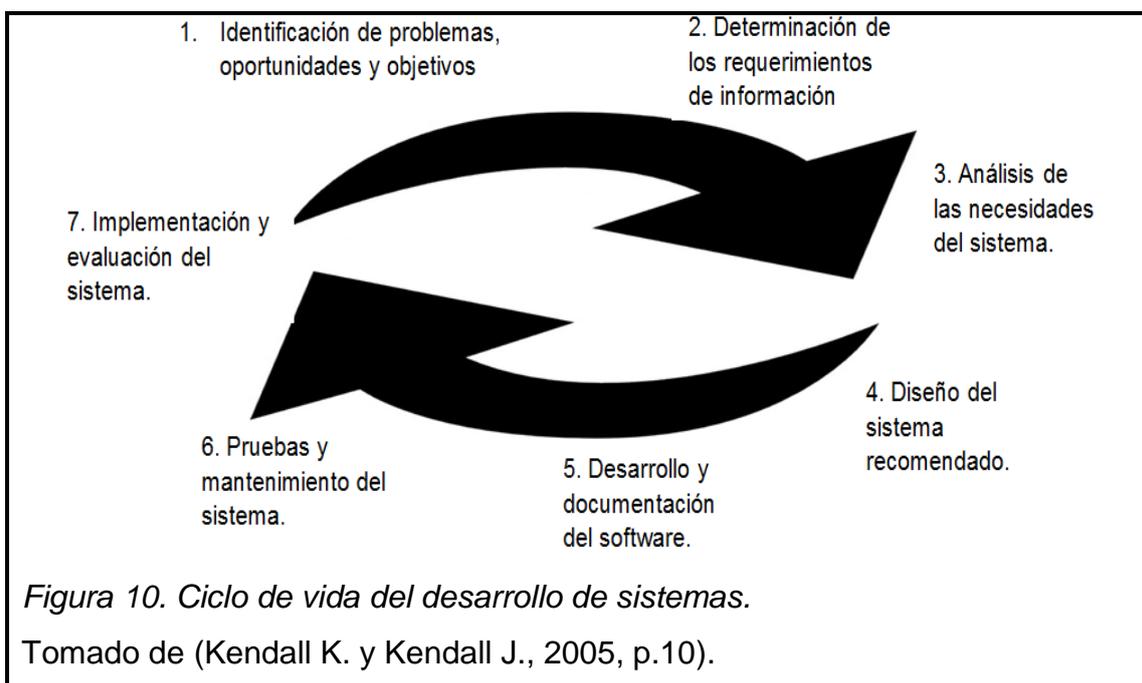
- ❖ **Snell:** Es emitida por una fundación privada sin fines de lucro, es una de las más exigentes en el mundo, realizaron pruebas que consisten en test de choques con deceleraciones en golpes entre 250 y 300 G, considerando como lesiones críticas para el cerebro. En el año 2013 se estandarizó y bajo la medición a los 250 G, por lo que se equiparó a la DOT como a la ECE/ONU. (Motopasion, s.f.).
- ❖ **DOT:** Una de las homologaciones más popular en Estados Unidos, protege hasta el 90% de los impactos, realizaron pruebas que consisten en test de choques hasta los 250 G donde prima la absorción de la resistencia de los impactos. (Motopasion, s.f.).
- ❖ **ECE/ONU:** Homologación válida y requerida en más de cincuenta países. Es muy similar a la homologación DOT. Minimiza el peso, comodidad durante el día y aumenta al máximo la protección. Son aprobados para los eventos de competición. (Motopasion, s.f.).

2. CAPÍTULO II. Análisis de Requerimientos del Sistema

2.1 APLICACIÓN MÓVIL

Previo a la implementación de la aplicación móvil es necesario seguir una serie de pasos para tener claro los aspectos más importantes de la aplicación a desarrollar. Utilizar el análisis de requerimientos con métodos establecidos que permite enfocar al sistema propuesto.

Los sistemas y aplicaciones se basan en ciclos de vida, que permiten analizar, diseñar y crear el sistema a plantearse, dividiendo por fases el análisis y el diseño del sistema para lograr un óptimo desarrollo y ejecución del sistema a realizarse. En la Figura 10 se muestra las fases de ciclo de vida del desarrollo de los sistemas.



Siendo el primer paso la identificación del problema, que incluye definir los objetivos y oportunidades asociadas a este.

El segundo paso en el análisis de requerimientos de la aplicación será entender la información recopilada del usuario que solicita la aplicación, la cual será utilizada para establecer los objetivos propuestos anteriormente.

El tercer paso el análisis de requerimientos de la aplicación será el análisis de las necesidades del sistema. Se establecen las herramientas, y técnicas que permitirán encaminar al sistema a un diseño total de la aplicación, utilizando diagramas de datos, estableciendo decisiones estructuradas, que establecen las condiciones, acciones y reglas de acción de la aplicación.

Al concluir los pasos de análisis de requerimientos del sistema, se prosigue con el diseño de la aplicación, no contemplado en el actual capítulo de desarrollo para el prototipo del sistema de casco inteligente.

2.2 IDENTIFICACIÓN DEL PROBLEMA, OBJETIVOS Y OPORTUNIDADES

2.2.1 Descripción del producto.

La aplicación del prototipo del casco inteligente pretende disminuir la respuesta de acción a un caso de emergencia en un accidente motorizado, mediante el uso del casco con el usuario que utiliza conjunto a la aplicación controladora.

2.2.2 Identificación del problema.

El problema existente es determinado según la Agencia Nacional de Tránsito (s.f.), por estadísticas de accidentes en motociclistas en la República del Ecuador hasta Mayo 2016 se registraron 636 siniestros con un porcentaje del 18%, también se registra el 31% de fallecimientos a comparación de Diciembre 2014 que se registraron 875 siniestros con un porcentaje del 18% y de fallecimientos con un 18%, el cual es un número alto y necesario de controlarlo. Además de la baja eficacia en respuesta de emergencia en estos casos, que

ocasiona serios problemas en el accidentado y hasta la muerte por no tener una rápida acción de respuesta.

2.2.3 Identificación de los objetivos.

El objetivo principal del sistema es reducir el tiempo de aviso de un accidente que se encuentra involucrado el usuario, a segundos después de lo ocurrido. Para lograrlo es necesario utilizar la aplicación a crear en conjunto con el hardware a implementar en el casco.

El segundo objetivo de la aplicación será almacenar los eventos ocurridos durante el transcurso de viaje del usuario en un lapso de tiempo, realizando muestras cada dos segundos, el cual permitirá saber que ocurrió previo al accidente.

El tercer objetivo de la aplicación será implementar el envío de mensaje de auxilio por medio de la red móvil al contacto predeterminado que es el principal actor en caso de una emergencia para pedir auxilio al usuario afectado.

2.2.4 Análisis de oportunidad de mercado.

Las respuestas a conseguir de las preguntas planteadas en este párrafo y complementadas en los siguientes, determinará la oportunidad de mercado para el prototipo a realizarse. ¿Quién o qué compite con el producto que se desarrollará? ¿Qué tecnologías se implementará para desarrollar el producto? ¿Cuál es la necesidad del mercado para realizar el prototipo propuesto? ¿Cuál es el plazo de salida de mercado del producto a desarrollarse?

Las oportunidades del sistema implicarán la satisfacción del usuario, el cual utilizará el prototipo, que en un caso de emergencia será vital. El sistema en caso de emergencia es útil en conjunto con el prototipo, permitirá una rápida alerta, el cual es la principal oportunidad del sistema en función a los otros

sistemas, sin encontrar competencia directa en sistemas relacionados en este caso.

Se implementará en el sistema móvil un conjunto de base de datos móvil el cual es necesario para satisfacer los objetivos planteados previamente. Es necesario para el usuario en un caso de emergencia, indicando la solución del problema principal. Finalmente, se establecerá como prototipo para realizar pruebas del sistema y funcionalidad, y posteriormente se podrá utilizar para mejorar y concluir estableciendo como necesario el prototipo propuesto.

2.2.5 Análisis de riesgos.

Los riesgos que se presentará durante el proceso de diseño e implementación, puede inducir a fallos relativos a personas en cuanto a la falta de experiencia del personal, fallos en implementación del hardware a utilizar en conjunto, entre otros.

Los riesgos en cuestión de software que se podrán inducir a los sistemas que dependen de la aplicación, como la base de datos a implementar, la comunicación entre el hardware y software a juntar, posibles fallos en nuevas sistemas y permisos para los dispositivos móviles, entre otros.

Los riesgos en cuestión de recursos que se podrían encontrar son de acuerdo a fallos en la comunicación con el hardware, el cual suministrará los datos a la aplicación, el fallo de obtención de datos desde la base de datos, entre otros.

El riesgo en cuestión de tecnología que se podrían encontrar es de acuerdo a modificaciones de permisos y accesos en los sistemas coordinadores del dispositivo móvil.

2.3 Determinación de los requerimientos.

2.3.1 Análisis del contexto.

Se divide en dos capas, la primera es el entorno en el que se desenvolverá la aplicación que está determinado por el casco, el cual determinará una percepción de cualquier tipo mediante los sensores. Puede ser nulo cuando el usuario navega en su motocicleta sin sufrir ningún accidente, o con datos cuando el usuario navega y sufre un accidente.

La siguiente capa se despliega por la interfaz que utiliza el sistema, que será por medio de conexión inalámbrica hasta la aplicación móvil, donde los datos se desplegarán. Y finalmente el sistema es encapsulado por el entorno global el cual gestionará y almacenará todos los datos, y procesará la información en caso de un accidente. El diagrama de contexto se muestra en la Figura 11 y Figura 12.

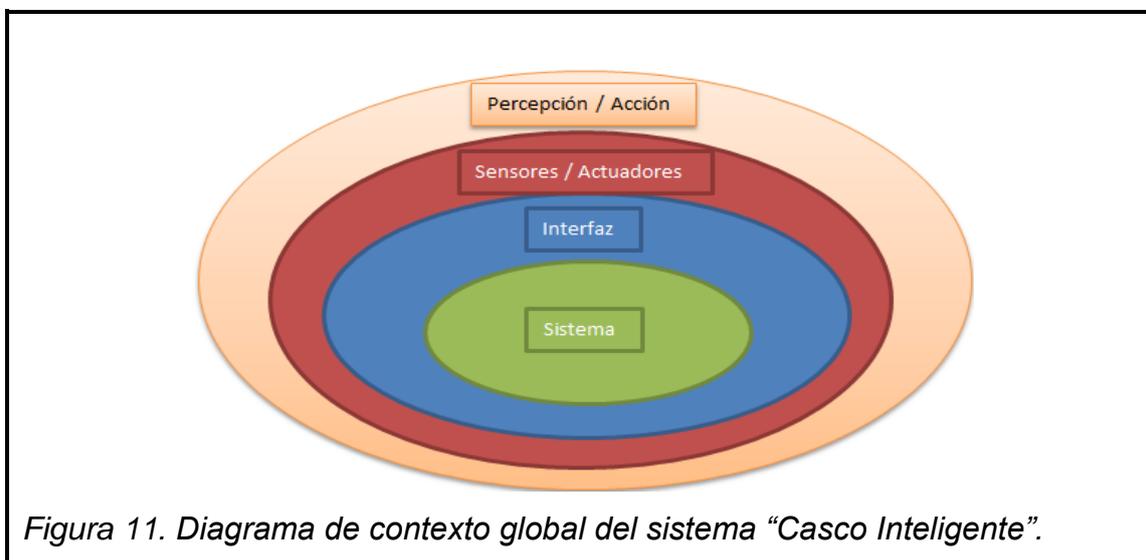


Figura 11. Diagrama de contexto global del sistema "Casco Inteligente".

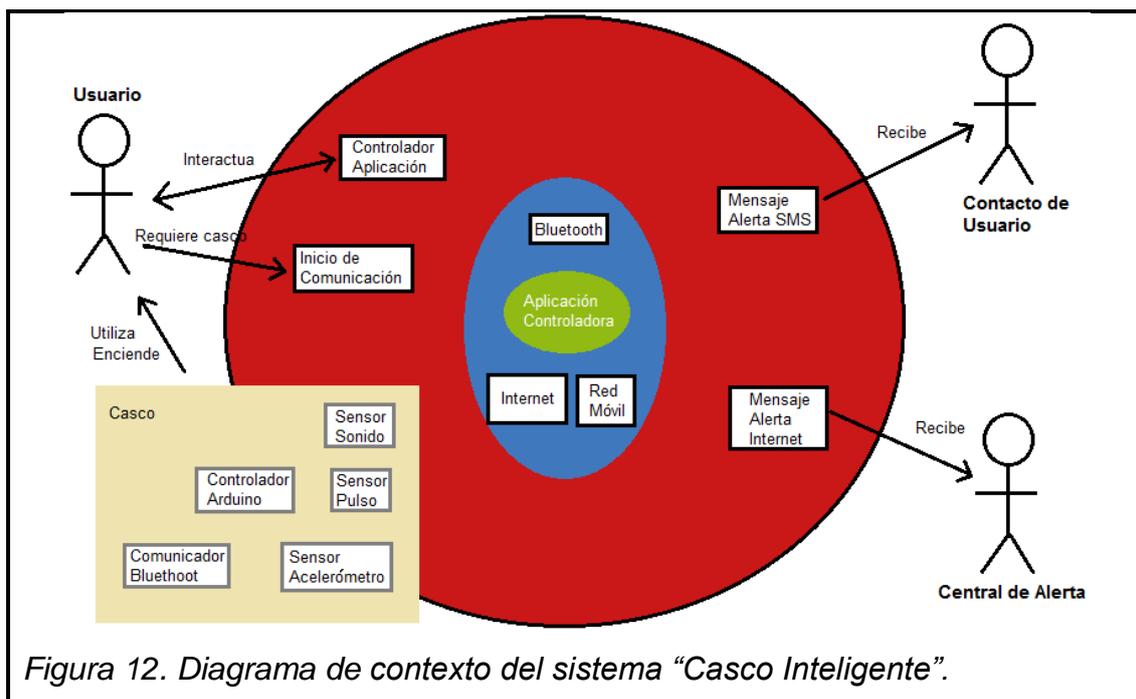


Figura 12. Diagrama de contexto del sistema "Casco Inteligente".

2.3.2 Objetos del contexto.

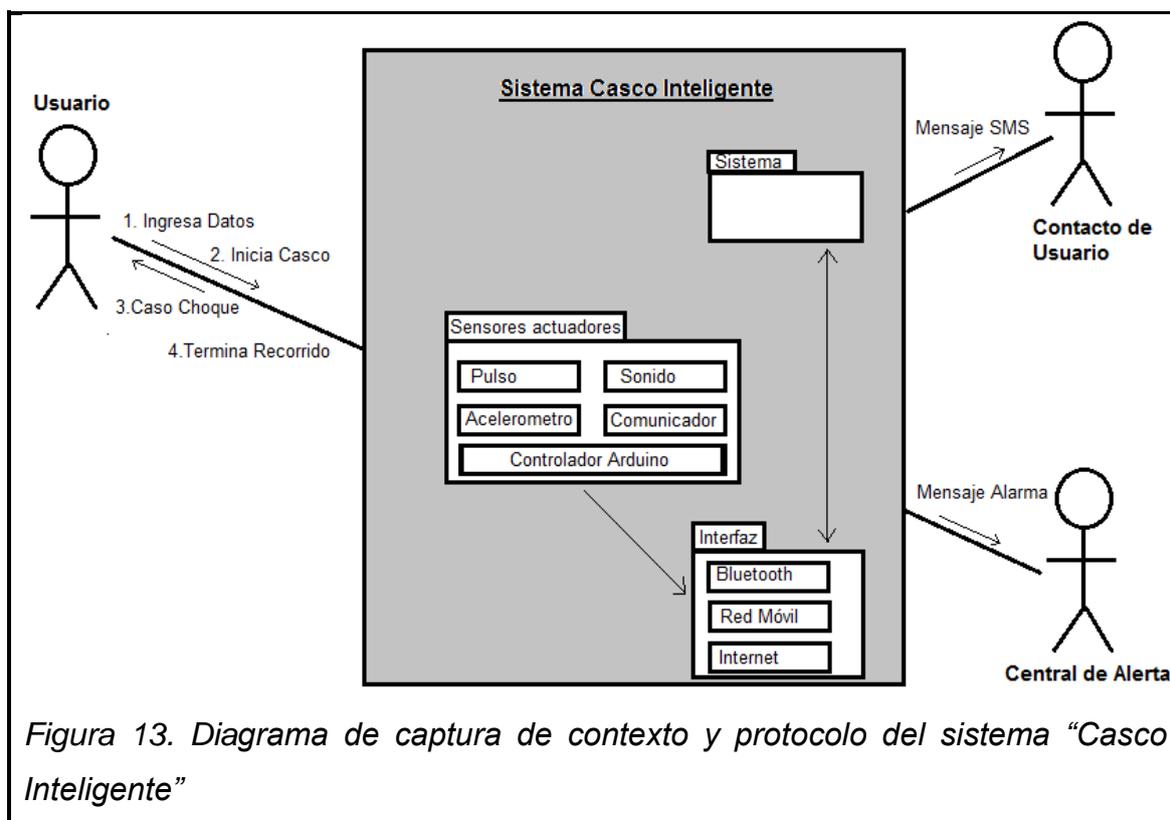
En el presente sistema se encuentra como objetos a los entes que entregan o reciben información al sistema, y los principales actores. En la tabla 2 se presenta los objetos del contexto para el sistema a diseñar e implementar.

Tabla 2. Objetos del contexto del sistema "Casco Inteligente".

Percepción / Acción	Sensores / Actuadores	Interfaz	Sistema
El usuario se coloca el casco para iniciar la conexión con el sistema.	Casco encendido. Transmisor Bluetooth.	Bluetooth 802.15.1	Aplicación móvil
El usuario navega con el casco colocado durante tiempo indefinido.	Casco encendido Transmisor Bluetooth. Sensor de sonido. Sensor de pulso. Sensor de aceleración.	Bluetooth 802.15.1	Aplicación móvil
El usuario sufre un accidente con el casco colocado durante la navegación.	Transmisor Bluetooth. Sensor de sonido. Sensor de pulso. Sensor de aceleración.	Bluetooth 802.15.1	Aplicación móvil
El usuario se saca el casco sin finalizar la navegación.	Transmisor Bluetooth. Sensor de pulso.	Bluetooth 802.15.1	Aplicación móvil

2.3.3 Captura de contexto y protocolo.

El diagrama final de presentación del análisis de contexto relaciona los objetos y el diagrama principal permite visualizar al sistema en un entorno global completo de lo que se implementará. Creará un claro enfoque de lo que realiza los diferentes elementos y actores con el sistema. En la Figura 13, se muestra el diagrama de captura de contexto y protocolo de la aplicación.



2.4 ANÁLISIS DE LAS NECESIDADES DEL SISTEMA.

2.4.1 Casos de uso

De acuerdo al análisis de requerimientos se planteó el diseño para la interacción del usuario del sistema y la base de datos para el correcto funcionamiento, previo a la implementación. En la Figura 14 se muestra el diagrama de casos de uso del sistema propuesto.

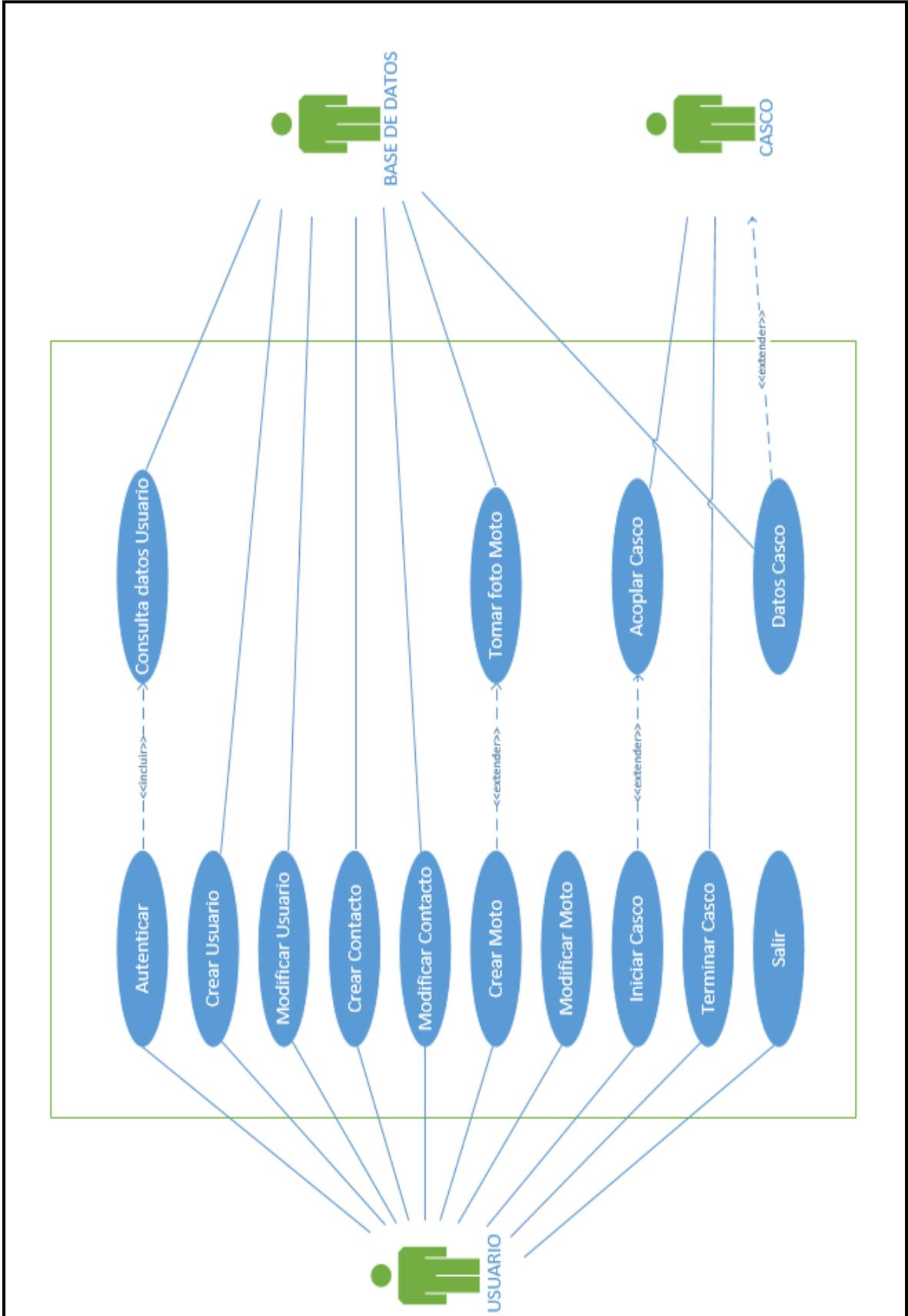


Figura 14. Diagrama de casos de uso.

2.5 SENSORES A IMPLEMENTAR

Para alcanzar los objetivos de diseño e implementación del sistema propuesto se requiere de una aplicación electrónica para la cual se necesitará usar diferentes tipos de sensores para adquirir las diferentes variables que es importante monitorear para esta aplicación, por tanto, se realizará un análisis y comparación de los sensores existentes en el mercado y se escogerá el que mejor se adapte a las necesidades del prototipo. Ver ANEXO 1.

2.5.1 Pulso

ARDUINO KY-039 – Módulo Detector de Latidos

KY-039 es un sensor de frecuencia cardíaca que trabaja para Arduino, el cual sirve para detectar el pulso del dedo, dicho sensor utiliza un LED infrarrojo brillante que se enciende con cada pulso, el LED genera la luz por un lado del dedo, mientras que el fototransistor detecta el pulso en la otra parte del dedo, si la presión arterial varía la resistencia del fototransistor tiene un cambio ligero. El fototransistor es utilizado para obtener el flujo emitido. Obteniendo las lecturas de pulso un poco confiables. (Sensores y Actuadores, s.f., p. 47).

Conexión al Arduino

El sensor KY-039 consta de tres pines como se muestra en la Figura 15, los cuales son:

- ❖ Sensor de pasador S se conectará al Arduino en el pin Analógico 0 (A0)
- ❖ Pin de sensor Positivo se conectará al Arduino en el pin 5V
- ❖ Pin de sensor Negativo se conectará al Arduino en el pin GND

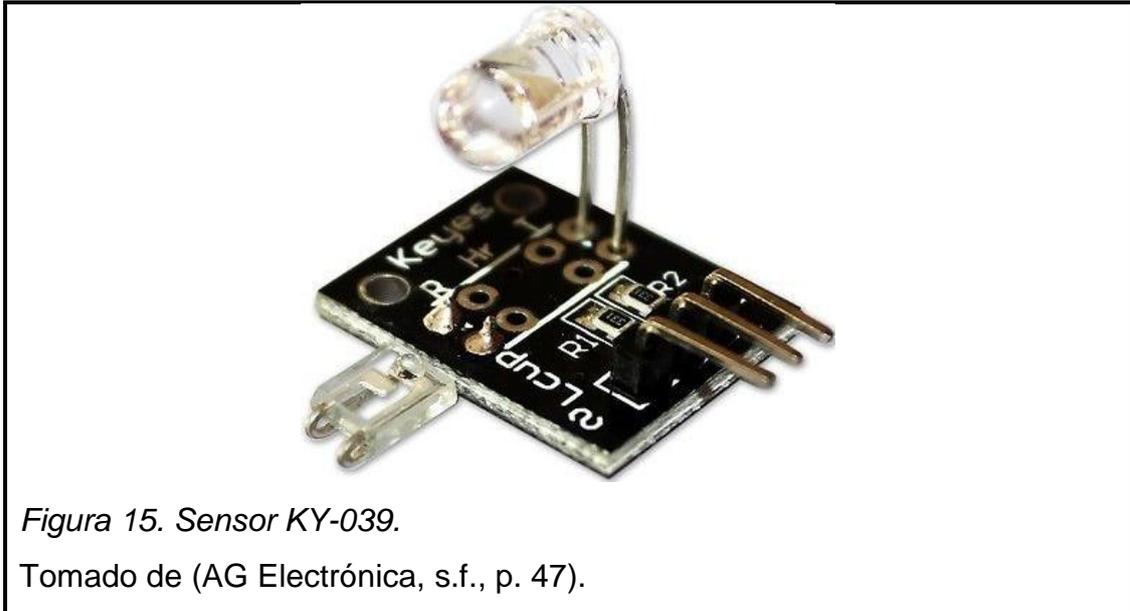


Figura 15. Sensor KY-039.

Tomado de (AG Electrónica, s.f., p. 47).

SENSOR DE PULSO SEN-11574

Es un dispositivo médico que es usado para monitorear la frecuencia cardíaca plug and play. Tiene una aplicación de código abierto que representa el pulso en tiempo real.

El corazón bombea sangre a través del cuerpo, con cada latido hay una onda de pulso, es decir, es como una onda de choque, que se desplazará a lo largo de las arterias de las extremidades del tejido capilar donde se colocará el sensor de pulso dependiendo del proyecto. La sangre real circula en el cuerpo mucho más lento de que la onda de pulso viaja.

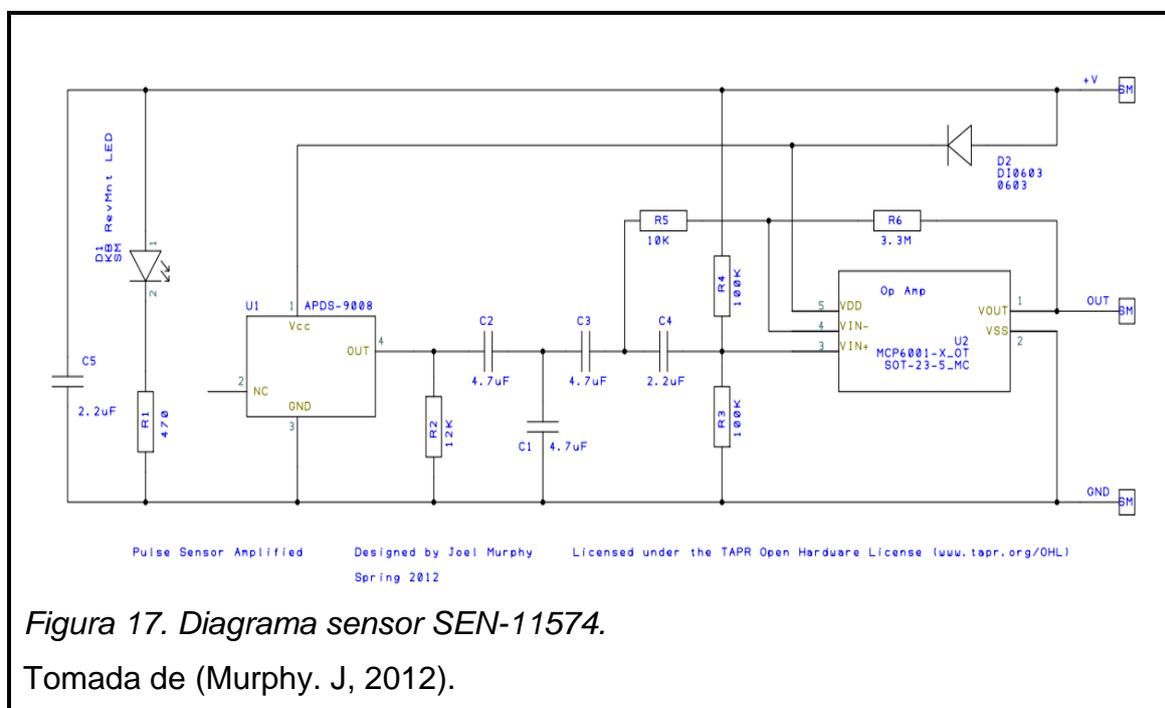
Es ideal para aplicaciones móviles porque trabaja con 5V. De fácil uso que puede ser utilizado por cualquier persona que deseen trabajar con datos de frecuencia cardíaca. Añade la amplificación de ruido y la cancelación de circuitos para el hardware.

La señal de pulso sale de una foto pletismógrafo que es una fluctuación del voltaje analógico y obtiene una forma de onda predecible. Amplifica la señal sin procesar el pulso anterior y normaliza la onda del pulso del punto medio de la

tensión ($\frac{V}{2}$). Dicho sensor está diseñado para colocar en el dedo o en el lóbulo de la oreja que está completamente dimensionado por su diámetro y su espesor, como se muestra en la Figura 16. (SparkFun, s.f.).



Diagrama sensor de pulso SEN-11574



En la Tabla 3 se muestra las variables útiles que se encuentran en el programa y con la frecuencia con la que se actualiza.

Tabla 3. Variables útiles del sensor de Pulso.

VARIABLE	FRECUENCIA DE ACTUALIZACIÓN	DESCRIPCIÓN
Señal	2mS	Señal del sensor de pulso en bruto.
IBI	Cada latido	El tiempo entre los latidos del corazón en EM.
BPM	Cada latido	Latidos por minuto.
QS	Establecer cada golpe del latido	Modificaciones por el usuario (programador).
Pulso	Establecer cada golpe del latido	Aprobado por ISR.

CARACTERÍSTICAS TÉCNICAS

Tabla 4. Características técnicas del sensor de Pulso.

Características	Sensores de pulso	
	KY-039	Sen-11574
Costo Estados Unidos (Dólares) / Costo Ecuador (Dólares)	7,98 / -	24,95 / 40,00
Disponibilidad en mercado (Ecuador)	No	Si
Tipo de salida	Analógica	Analógica
Tipo de transmisión	Onda freq: 38Khz	Onda freq: 38Khz
Voltaje (DC)	5 V.	5 V.
Lugar de detección	Dedo Índice	Lóbulo de Oído, Dedo Índice
Dimensiones (cm)	2.5 x 2.0 x 1.8	1.59 (Diámetro) x 0.32(Grosor)
Peso (kg)	0,0005	0,0003

Para el proyecto a realizar se analizó los sensores de pulso propuestos anteriormente y de acuerdo a las características, necesidades que se va a utilizar en el prototipo se eligió el sensor de pulso SEN-11574.

Necesitamos conocer los signos vitales del usuario al momento de un accidente de tránsito, los valores obtenidos del sensor se enviará a la aplicación móvil. Además dicho sensor es un modelo muy pequeño, se puede encontrar fácilmente en el mercado, de fácil uso y programación, sirve para aplicaciones móviles por su voltaje y además que nos permitirá colocar en el lóbulo de la oreja y se podría adaptar en el casco.

2.5.2 Sonido

LM386

Según National Semiconductor (2000). El módulo LM386 detecta rápidamente el sonido, usualmente se lo utiliza para la detección de la intensidad del sonido. Puede utilizarse con varios fines como por ejemplo para aplicaciones de seguridad, interruptor o monitoreo. De acuerdo al uso al que se le dé, se puede regular al sensor.

Es un amplificador de potencia de audio con características de bajo consumo de energía. Es capaz de configurar la ganancia de tensión dentro del rango de 0–200. (National Semiconductor, 2000, p. 3). La tensión de funcionamiento es de 3.3V a 5V, tiene la salida de conmutación digital (0 y 1, nivel alto o bajo) y el tamaño del PCB de 3.4cm x 1.6cm. Consta de cuatro pines:

- ❖ Poder (VCC)
- ❖ GND (Ground)
- ❖ Señal de salida digital (D0)
- ❖ Señal de salida analógica (A0)

Sensor HXJ-17

Se usa para detectar la intensidad de sonido del ambiente, la tensión de alimentación es de 3.3V a 5V. La interfaz de dicho sensor es Analógico, se

puede lograr la percepción del sonido con el efecto de interacción, como se muestra en la Figura 18.

La estructura de este sensor es muy liviana, es decir, de montaje estándar. Las interfaces de los sensores son de fácil reconocimiento. La codificación de este sensor en Arduino es muy simple y fácil de interpretar. El conector que disponen es de la más alta calidad. Consta de tres pines:

- ❖ Poder (VCC)
- ❖ GND (Ground)
- ❖ Señal de salida



CARACTERÍSTICAS TÉCNICAS

Tabla 5. Características técnicas del sensor de Sonido.

Características	Sensores de sonido	
	LM386	HXJ-17
Costo Estados Unidos (Dólares) / Costo Ecuador (Dólares)	1,15 / -	2,80 / 7,50
Disponibilidad en mercado (Ecuador)	No	Si
Tipo de salida	Analógica / Variación de voltaje	Analógica / Variación de voltaje
Voltaje (DC)	5 V.	5 V.
Recepción de señal	Micrófono	Micrófono
Dimensiones (cm)	N/A	4.3 x 1.6 x 1.3
Peso (kg)	0,0004	0,0005

Según las necesidades del proyecto se utilizará el sensor de sonido HXJ-17. Ya que necesitamos saber el impacto al momento de un accidente de moto, por lo cual se eligió el mencionado sensor ya que cuenta con las características requeridas para el proyecto por su tamaño por lo que es fácil de colocar en la parte interna del casco, utilizaremos la salida analógica del sensor. La codificación es muy simple y fácil de encontrar en el mercado.

2.5.3 Aceleración

Sensor ADXL345

Es un dispositivo muy pequeño, de baja potencia, consta con un acelerómetro de 3 ejes, con la medición de alta resolución (13 bits) en un máximo de ± 16 g. de datos de salida digital tiene el formato de dos en dos de 16 bits se complementan y es accesible a través de ya sea un SPI (3- o 4 hilos) o interfaz digital I2C. Como se muestra en la Figura 19.

Es adecuado para medir la aceleración de la gravedad estática en aplicaciones de inclinación de detección, así como la aceleración dinámica resultante del movimiento o shock. La actividad y la inactividad de detección, detecta la presencia o ausencia de movimiento y si la aceleración en cualquier eje supera un nivel establecido por el usuario. Un sistema integrado, pendiente de patente de 32 niveles primero en entrar, primero en salir (FIFO) se puede utilizar para almacenar datos para minimizar la intervención del procesador central. Modos de bajo consumo permiten la administración de energía basado en movimiento inteligente con sensor de medición de la aceleración umbral y activa a extremadamente baja disipación de potencia. (Earl, s.f., p. 4-6).

Características del sensor ADXL-345:

- ❖ Tensión de alimentación 2.0-3.6VDC
- ❖ Ultra Low Power: 40uA en el modo de medición, 0.1uA en modo de espera 2.5V

- ❖ Toque / Detección Double Tap
- ❖ La detección de caída libre
- ❖ SPI y las interfaces I2C

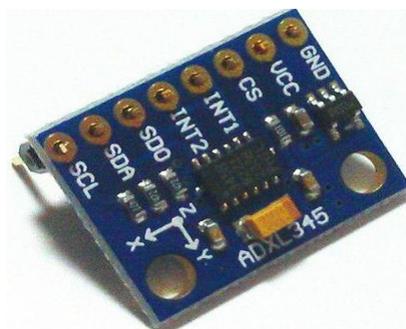


Figura 19. Sensor de Aceleración ADXL345.

Tomado de (Earl, s.f., p.1).

Sensor GY-88

Incluye un acelerómetro, brújula digital y sensor de presión barométrica/temperatura. Los sensores son accesibles a través de I2C lo cual necesita cuatro conexiones para acceder: SDA, SCL, VCC y tierra. Se utiliza a 3,3V – 5V. Acelerómetro de 3 ejes, giroscopio de 3 ejes MPU 6050. Módulo sensor de campo magnético de 3 ejes HMC5883L, como se muestra en la Figura 20. (Bernal, s.f., p. 23)

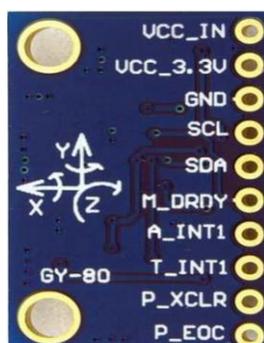


Figura 20. Sensor de Aceleración GY-88.

Tomado de (Arduino, s.f.).

CARACTERÍSTICAS TÉCNICAS

Tabla 6. Características técnicas del sensor de Aceleración.

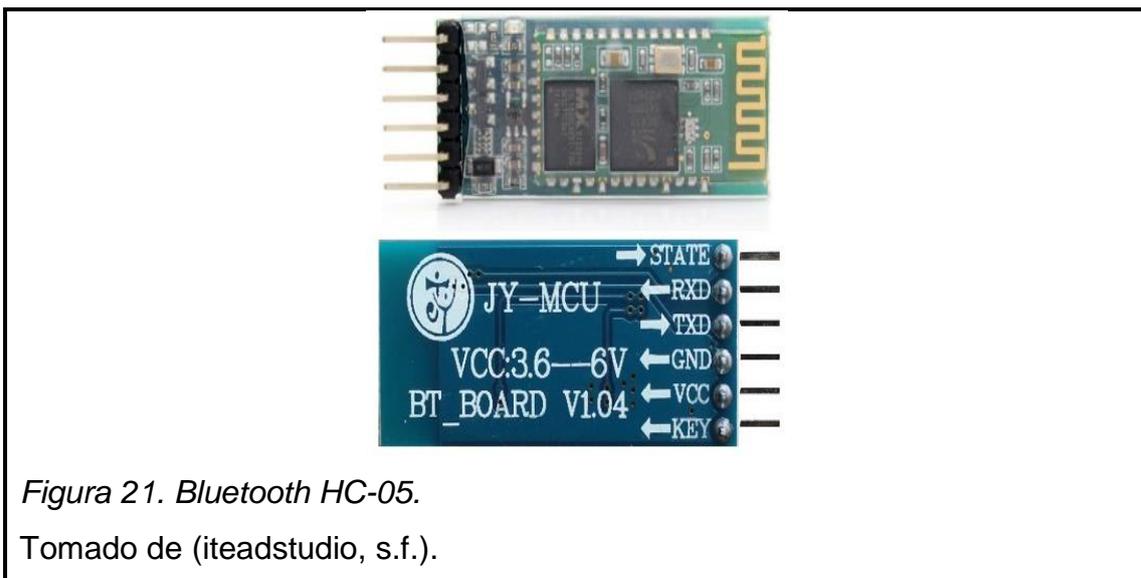
Características	Sensores de aceleración	
	ADXL345	GY-88
Costo Estados Unidos (Dólares) / Costo Ecuador (Dólares)	7,59/18,00	16,00/ -
Disponibilidad en mercado (Ecuador)	Si	No
Tipo de salida	Digital	Digital
Tipo de transmisión	SPI-I2C	I2C
Voltaje (DC)	3,3 V. - 5 V.	3,3 V. - 5 V.
Sensibilidad (G)	2, 4, 8, 16	2, 4, 8, 16
Librerías Soportadas	ADXL345	-
Lugar de detección	Superficie	Superficie
Dimensiones (cm)	1.5 x 2.0 x 0.2	2.2 x 1.6 x 0.2
Peso (kg)	0,0005	0,0005

Según el requerimiento del sistema propuesto se necesita el sensor ADXL345, ya que su tamaño es pequeño, la precisión es casi exacta, de baja potencia, consta con un acelerómetro de 3 ejes, la codificación no es compleja ya que en Arduino tiene su propia librería, existe en el mercado y de precio económico.

2.5.4 Módulo Bluetooth para Arduino

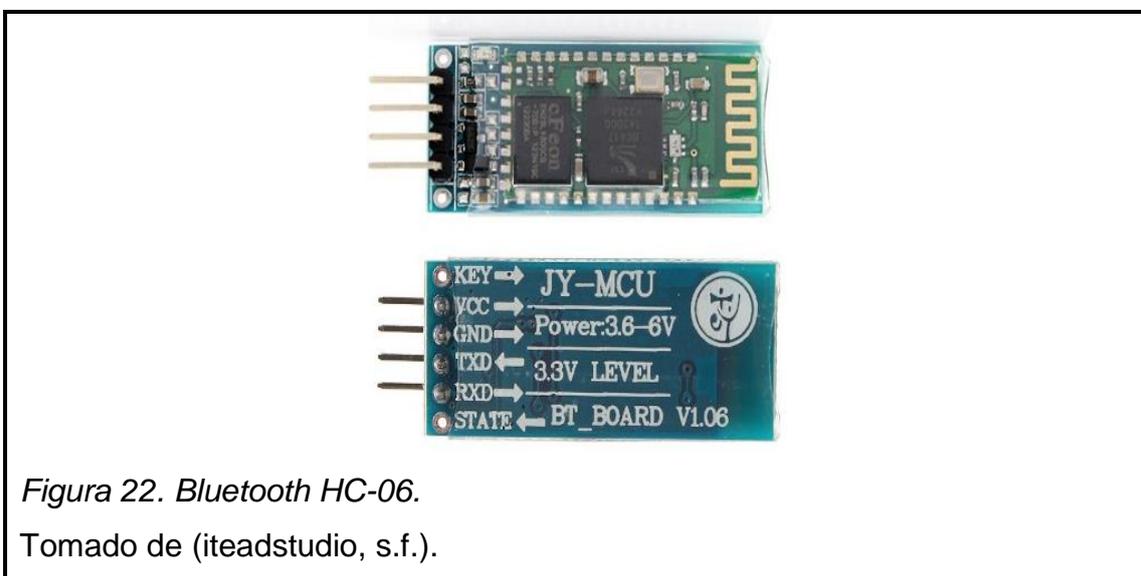
Bluetooth HC-05

Puede ser maestro/esclavo. Además de recibir conexiones desde una PC o equipo, también es capaz de generar conexiones hacia otros dispositivos Bluetooth. Lo cual nos permitirá conectar dos módulos de y formar una conexión punto a punto para transmitir datos entre dos micro controladores o dispositivos, como se muestra en la Figura 21.



Bluetooth HC-06

Puede actuar como esclavo. Tiene un firmware distinto y funcionamiento distinto a HC-05. Acepta un set muy básico de comandos que permite de muy pocas configuraciones, como se muestra en la Figura 22.



La diferencia entre maestro y esclavo es básicamente, el maestro se conecta con un dispositivo mientras que el esclavo es el dispositivo que se conecta al módulo.

Físicamente estos módulos son muy parecidos con la diferencia que varían de algunas conexiones.

Los pins son los siguientes:

- ❖ Vcc: Alimentación del módulo entre 3,6V y 6V.
- ❖ GND: La masa del módulo.
- ❖ TXD: Transmisión de datos.
- ❖ RXD: Recepción de datos a un voltaje de 3,3V
- ❖ KEY: Poner a nivel alto para entrar en modo configuración del módulo (únicamente en el modelo HC-05)
- ❖ STATE: Para conectar un led de salida para visualizar cuando se comuniquen datos.

Para la conexión inalámbrica del casco a la aplicación móvil se requiere del módulo HC-05, ya que es maestro/esclavo, lo más importante es que puede recibir conexiones desde una PC, equipo o dispositivo móvil, y genera conexiones hacia otros dispositivos Bluetooth y cumple con los requerimientos solicitados.

2.6 Diagrama de bloques del sistema.

El sistema a implementar estará enfocado con sensores que recepte información desde el casco, que serán la principal fuente de información. Se alimentarán de energía por medio de una batería, la cual brindará energía a todo el circuito. El principal controlador será el dispositivo Arduino, el que controlará la información de ingreso y enviará la información por medio de un transmisor de datos, en este caso por medio de un dispositivo Bluetooth. El receptor de los datos estará en el celular mediante una aplicación móvil que permitirá la conexión con el transmisor Bluetooth. Finalmente, la salida del mensaje de auxilio será desde un dispositivo celular por medio de la red móvil o de datos. En la Figura 23 se muestra visualmente como se implementará el sistema, y en la Figura 24 se muestra el diagrama de bloques de entrada y salida de datos para el sistema completo propuesto.

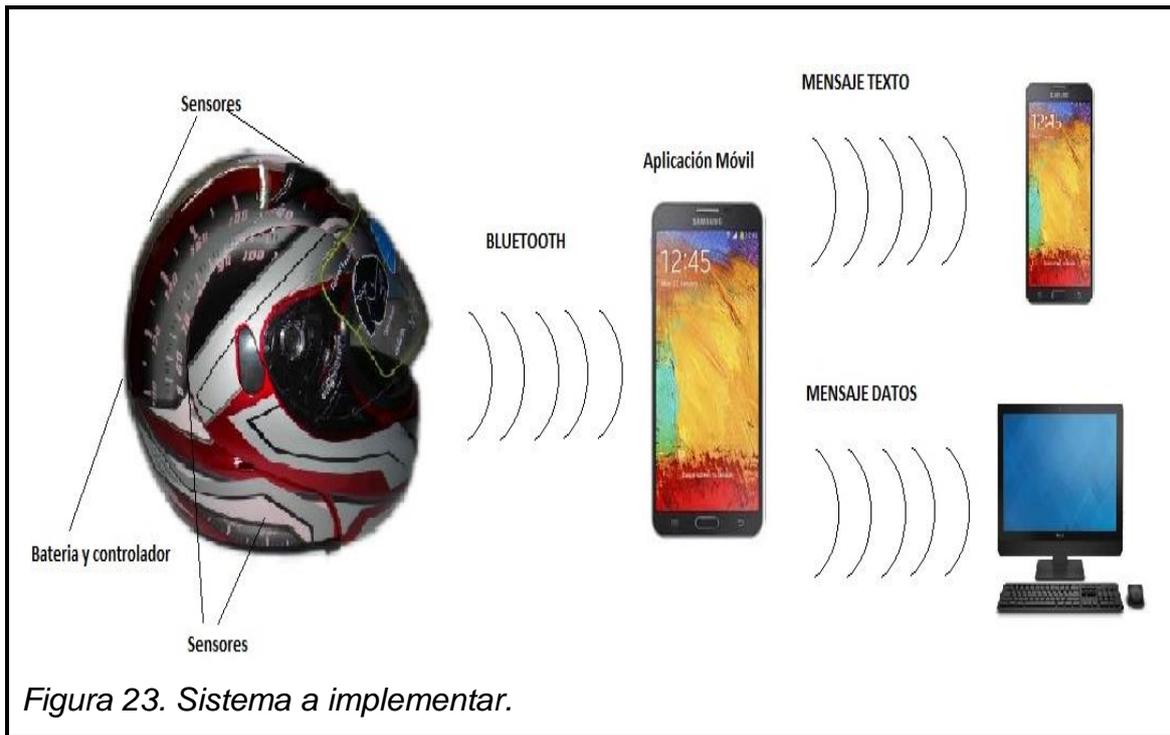


Figura 23. Sistema a implementar.

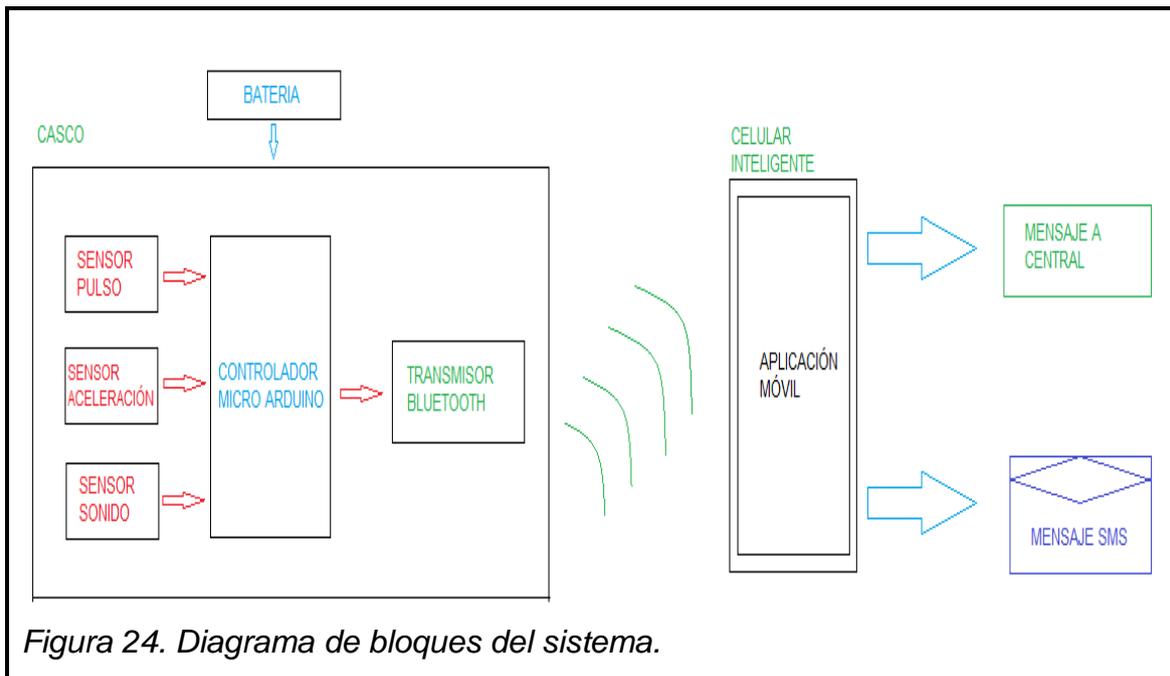


Figura 24. Diagrama de bloques del sistema.

3. CAPÍTULO III. Diseño del Prototipo

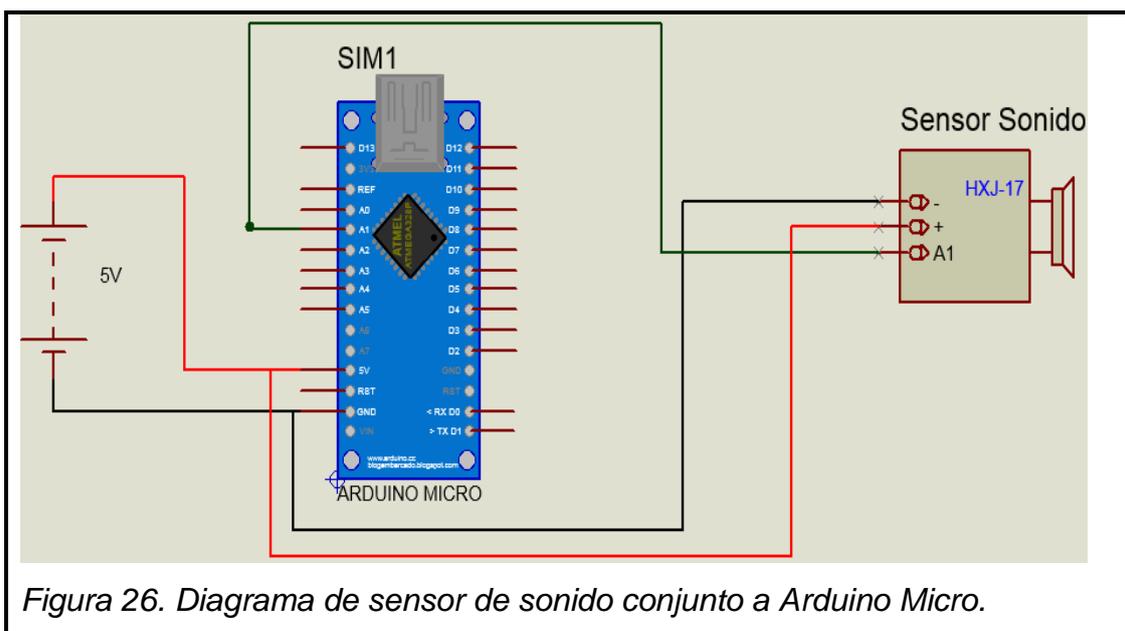
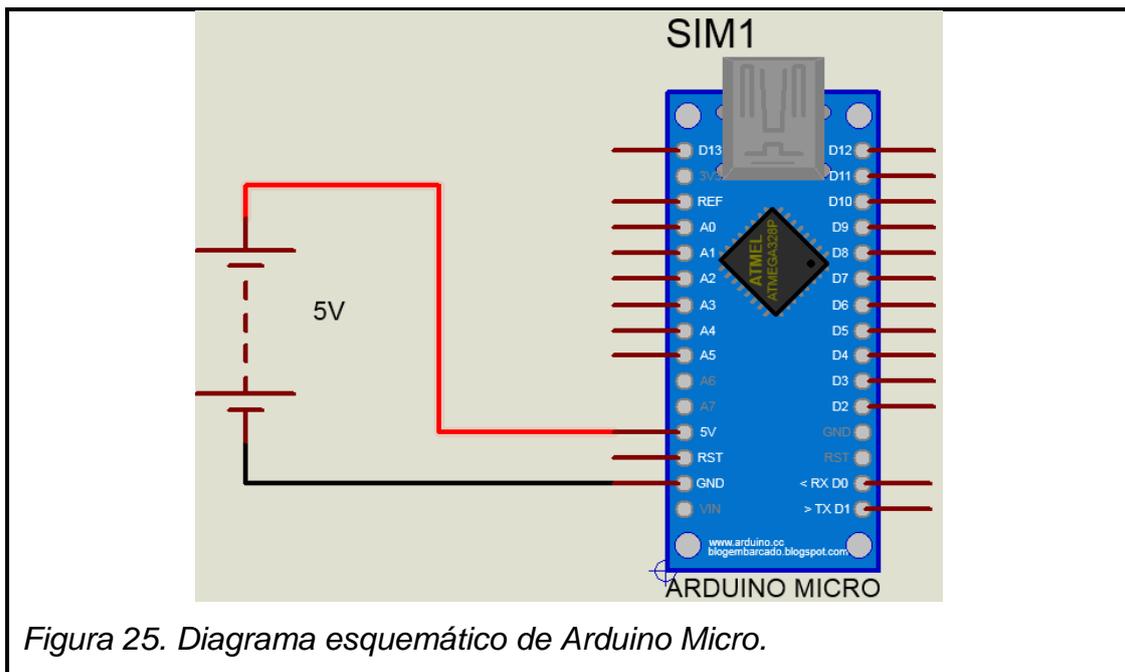
3.1 Diseño de Hardware

Para el diseño del hardware se utilizará los sensores elegidos y estudiados anteriormente, con el fin de poder realizar un mejor diseño para el prototipo. Se detallará los objetos a utilizarse:

Batería.- Es la fuente de energía, su función es entregar energía en voltaje continuo, se utilizará un voltaje de 5 voltios, ya que el consumo de voltaje del Arduino es de 5V, el cual entregará la misma cantidad de voltaje a los sensores propuestos y bluetooth.

Arduino micro.- Permitirá ser el centro de concentración, obtención y procesamiento de datos para el hardware el cual permitirá el almacenamiento de datos, almacenamiento de codificación de sensores y objetos, y alimentación de energía a los dispositivos, se eligió dicho Arduino por su diseño, el tamaño, compatibilidad con los sensores y tiene los suficientes puertos para el proyecto propuesto. En la Figura 25 se muestra las compuertas y diagramación del dispositivo.

Sensor de sonido.- Recibirá datos de vibración que genera el sonido que determinará si el casco recibe un impacto. El sensor analógico HXJ-17 transmitirá valores enteros desde 0, y se regulará de acuerdo al mínimo necesario. En la Figura 26 se muestra la conexión del sensor de sonido conjunto al centro principal Arduino.



Sensor de pulso.- Recibirá datos de los pulsos del latido de corazón, tomado desde el lóbulo de la oreja del usuario, el cual por medio de sensores de luz y la ayuda de capilares, que al llenarse de sangre se expanden, se determinará la existencia de contracción ventricular y del corazón. El sensor analógico transmitirá estos valores y procederá a realizar funciones específicas para determinar el ritmo cardíaco en el centro principal Arduino. En la Figura 27 se

muestra la conexión del sensor de pulso en conjunto al centro principal Arduino.

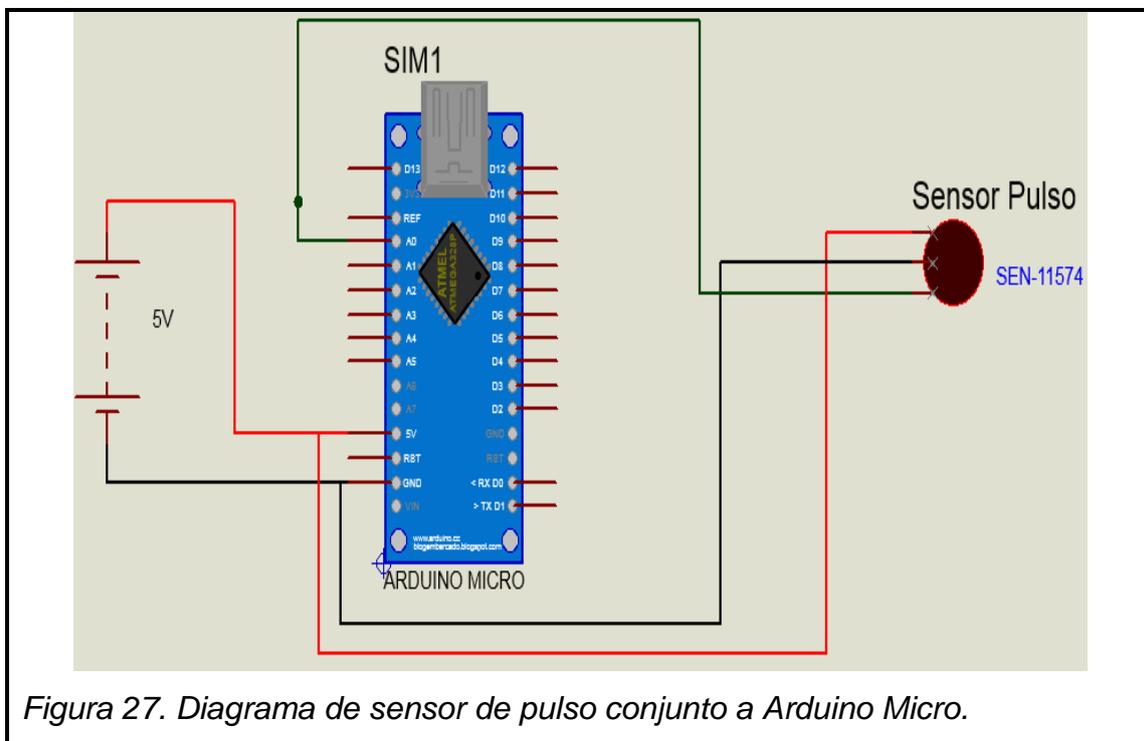


Figura 27. Diagrama de sensor de pulso conjunto a Arduino Micro.

Sensor de aceleración.- Recibirá datos del cambio de movimiento de la cabeza del usuario, tomado desde el casco, por medio de un arreglo de capacitores los cuales determinarán el movimiento de masa, de acuerdo a los planos. El sensor digital transmitirá valores predeterminados en los ejes del plano cartesiano, representando en el eje del casco. En la Figura 28 se muestra la conexión del sensor de aceleración en conjunto al centro principal Arduino.

Transmisor de señal bluetooth.- Transformará la señal física transmitida desde el centro principal Arduino, para ser enviado por medio de señal no guiada en estándar 802.15.1, y receptorá la señal de ingreso desde un dispositivo móvil. En la Figura 29 se muestra la conexión del transmisor en conjunto al centro principal Arduino.

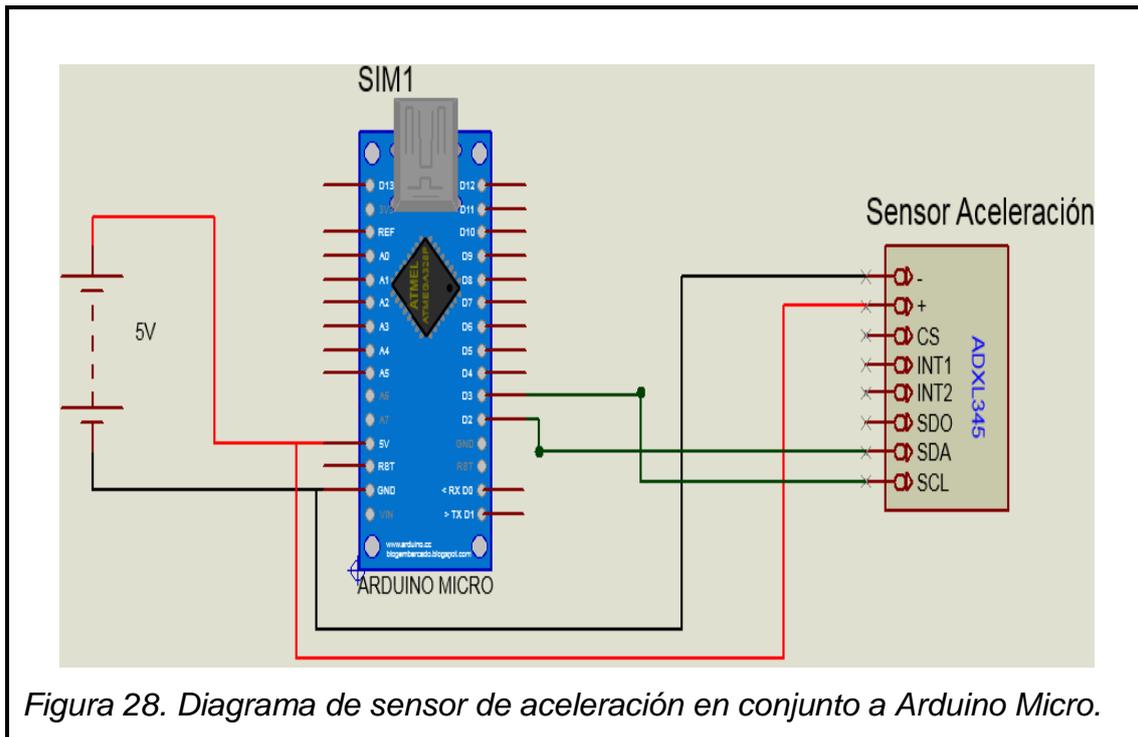


Figura 28. Diagrama de sensor de aceleración en conjunto a Arduino Micro.

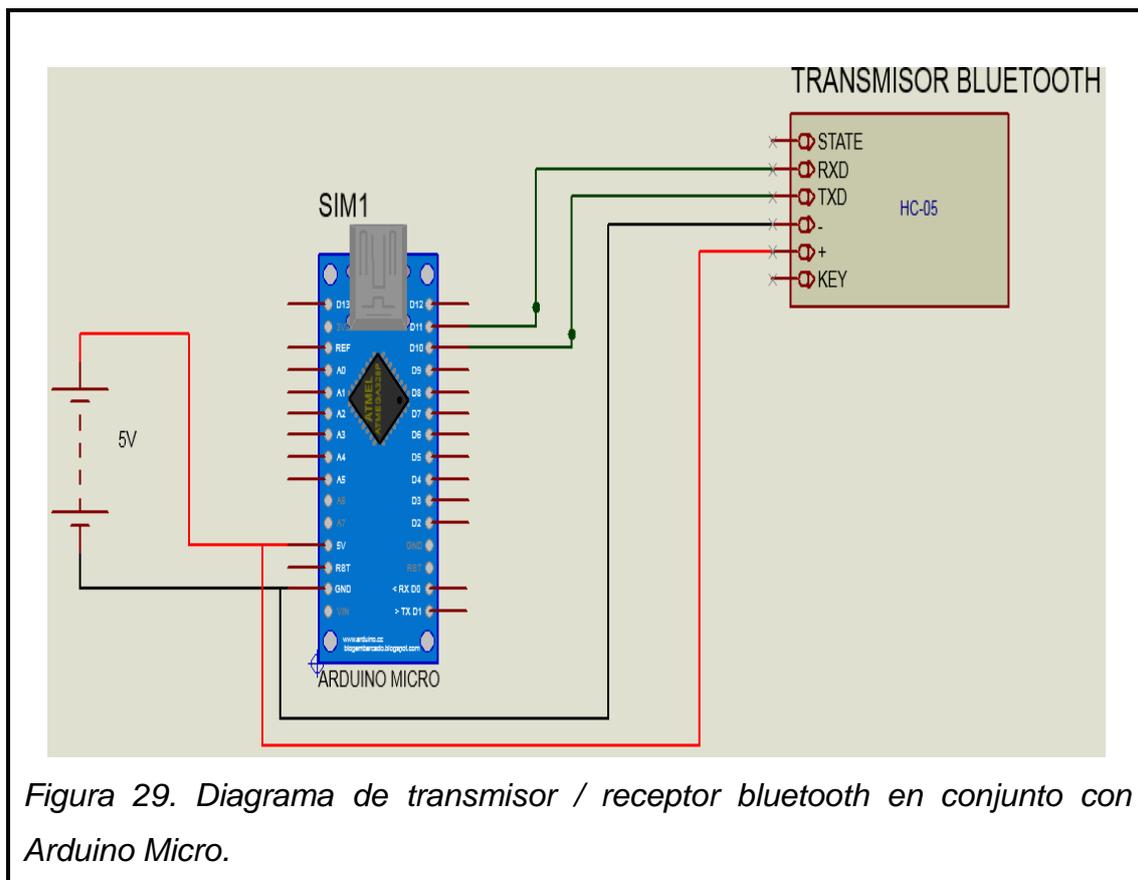


Figura 29. Diagrama de transmisor / receptor bluetooth en conjunto con Arduino Micro.

Circuito completo.- Arduino es el principal objeto que obtendrá y enviará datos a la aplicación, junto con los sensores crearán el hardware responsable en el casco principal para el funcionamiento. En la Figura 30 se muestra la conexión de todo el circuito.

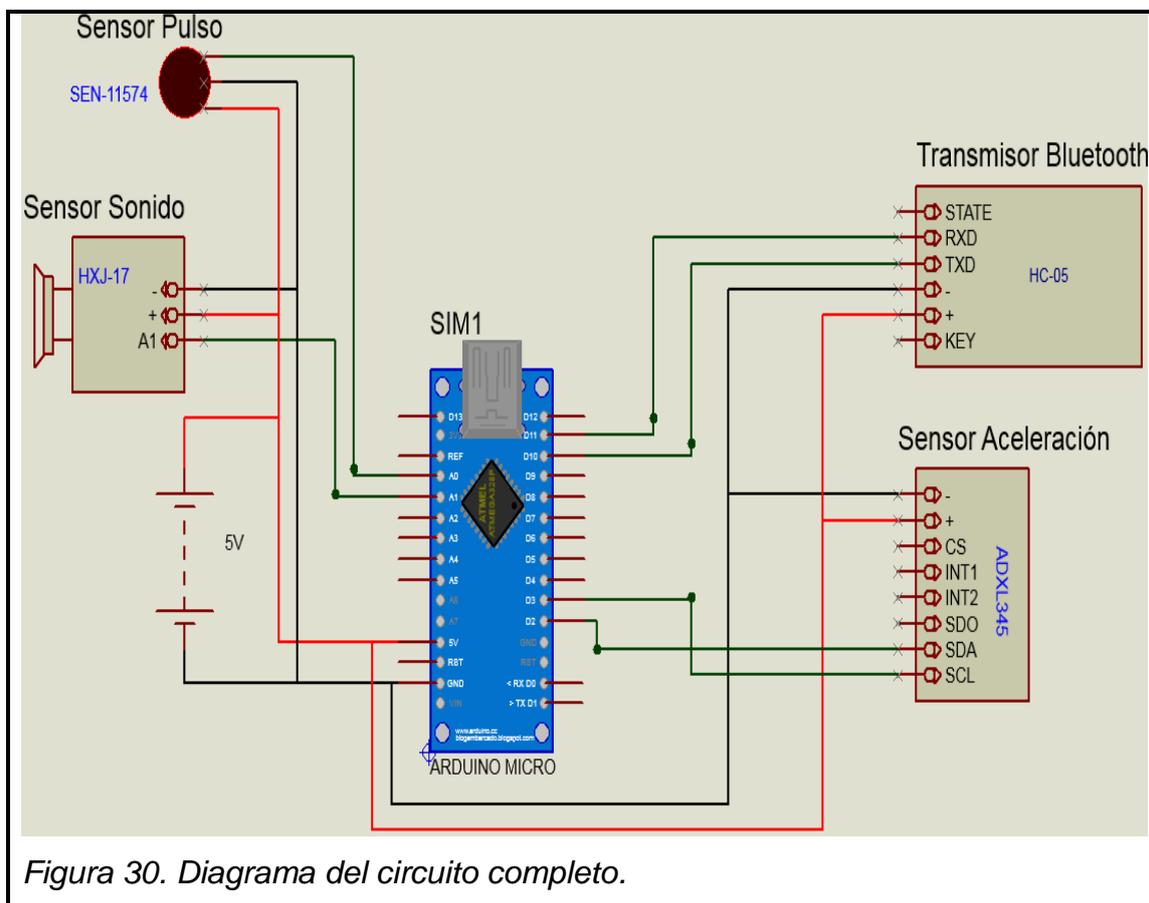


Figura 30. Diagrama del circuito completo.

3.2 DISEÑO DE HARDWARE EN EL CASCO

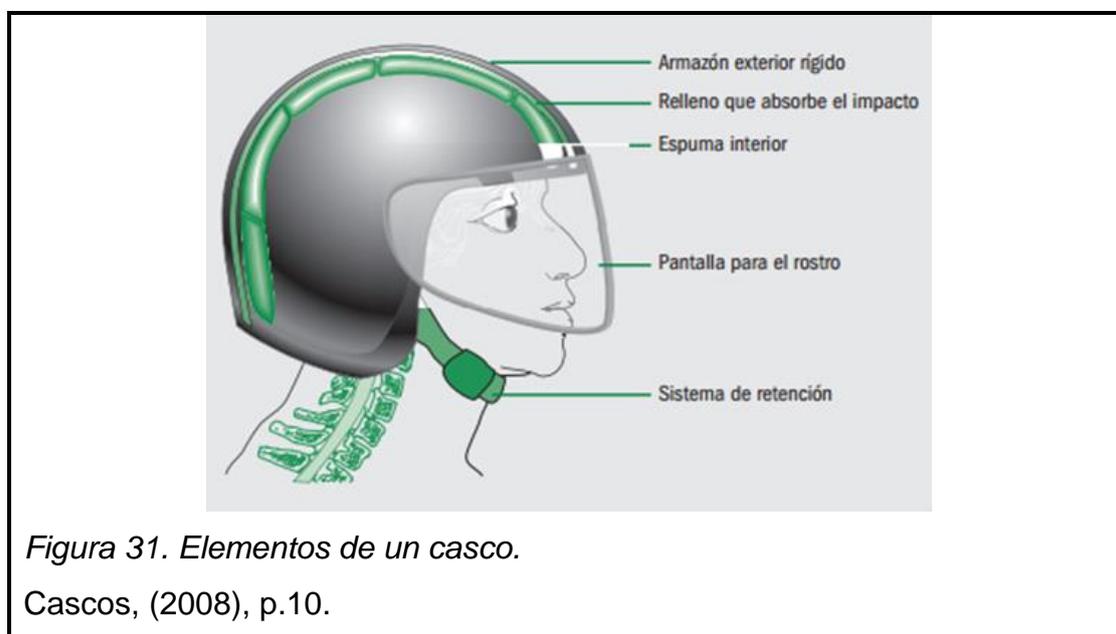
Definición del Casco

“El casco sirve para reducir el riesgo de traumatismos craneoencefálicos graves al aminorar el impacto de una fuerza o colisión en la cabeza.” (Cascos, 2008, p. 10).

Funciones principales del casco

Las funciones específicas de los cascos son para reducir la aceleración del cráneo y absorber el impacto de un accidente. Dispersa la fuerza del impacto en todo el casco, y previene el contacto entre el cráneo y la superficie con la que se generó el impacto. (Cascos, 2008, p. 10).

El casco tiene cuatro diferentes componentes básicos para cumplir los objetivos anteriormente descritos, en la Figura 31 se muestra gráficamente.



Descripción de elementos del casco

Armazón rígido exterior. Es la capa externa resistente del casco, este permitirá reducir la fuerza de impacto, ya que distribuye dicho impacto en una superficie mayor que corresponde al armazón.

Relleno absorbente de impacto. El material utilizado principalmente es polietileno expandido, el cual permitirá amortiguar el golpe y absorbe el impacto.

Espuma interior. Conformado con tela y espuma blanda, que entregará comodidad al usuario.

Sistema de retención. Permitirá mantener el casco en el cráneo del usuario mediante correas que sujetan por la parte inferior de la barbilla.

Los sensores se colocarán entre la espuma interior y el relleno absorbente del casco, por motivos de seguridad física y correcto funcionamiento. Para la colocación de los dispositivos se acoplarán según su medida, para que no afecte la comodidad o seguridad del usuario. No se realizará agujeros o alteraciones en el armazón rígido exterior, ya que afectaría las condiciones del casco en un posible accidente llegando a ocasionar serias heridas.

3.3 Diseño de Software

3.3.1 Software Arduino

Recibirá datos de pulso de los latidos del corazón, lo realizará cada 2ms, si determina el latido que es llamado BPM restablece el indicador cuantificado para la próxima vez, caso contrario toma una pausa de 20ms para buscar de nuevo otro latido. Llamará a una subrutina llamada "Interrupt", la cual se encargará de determinar el pulso, tiempo de pulso, etc., como se muestra en la Figura 32 y Figura 33.

Recibirá los datos de vibración que genera el sonido y los valores que se obtendrá serán enviados a un array para posteriormente serán visualizados en la aplicación, como se muestra en la Figura 34.

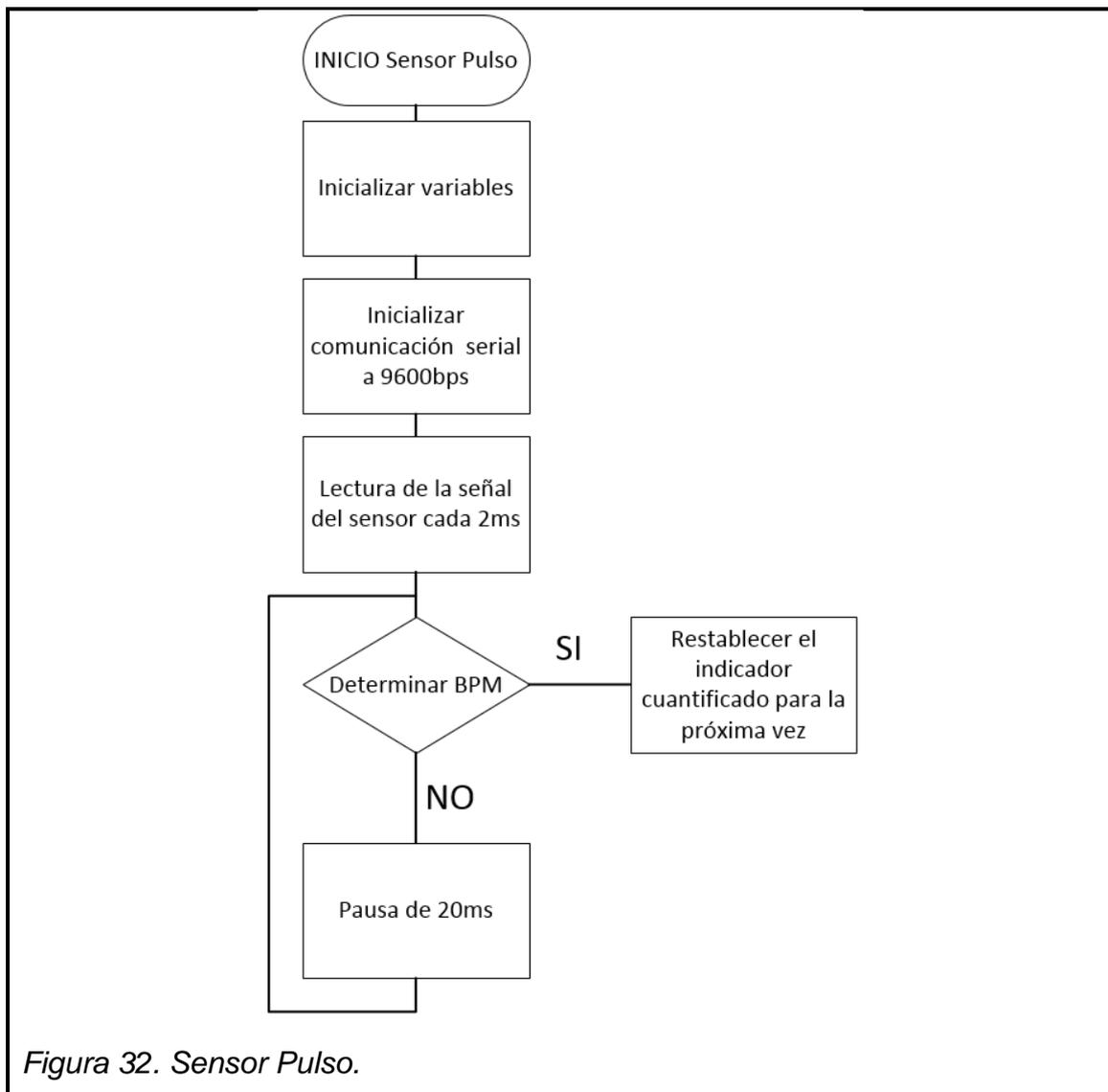
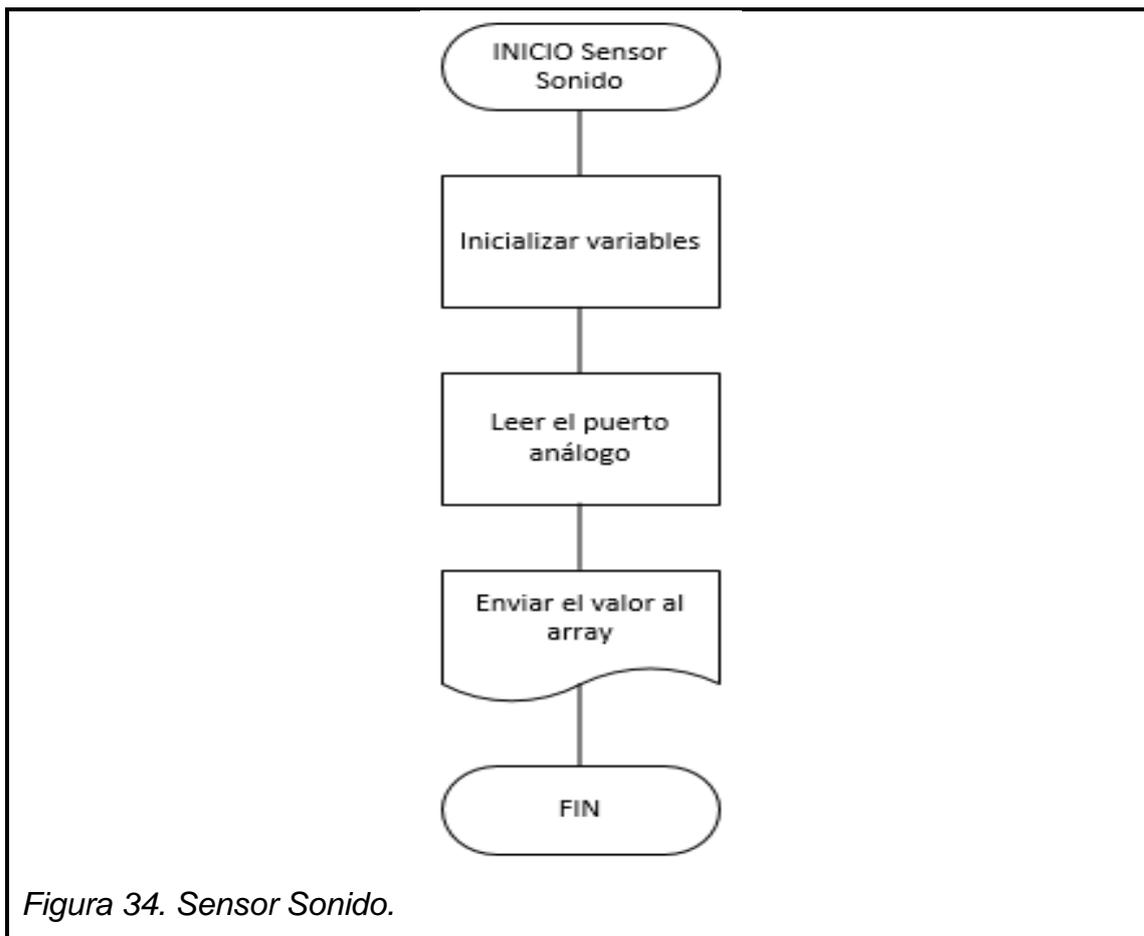
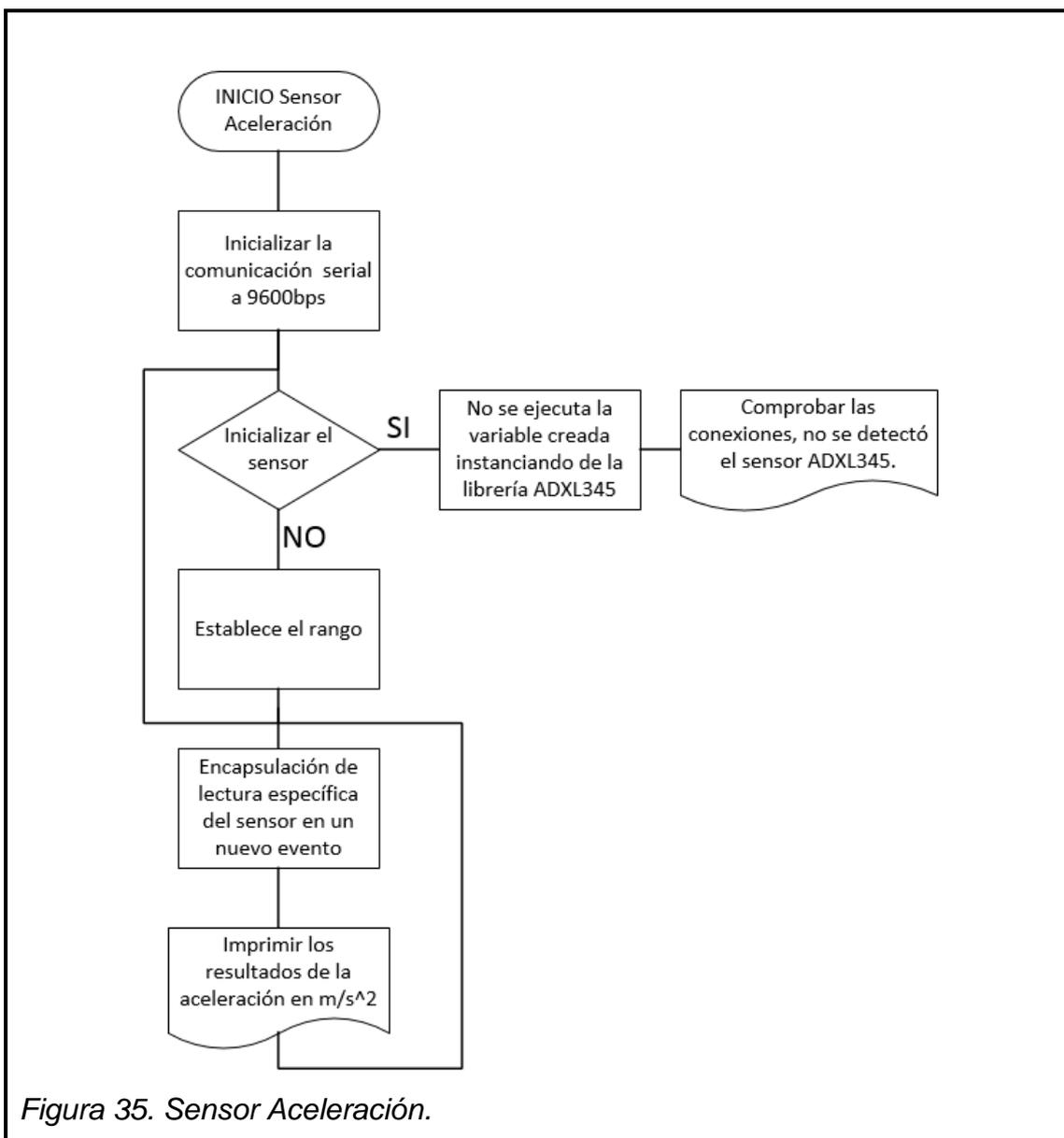


Figura 32. Sensor Pulso.

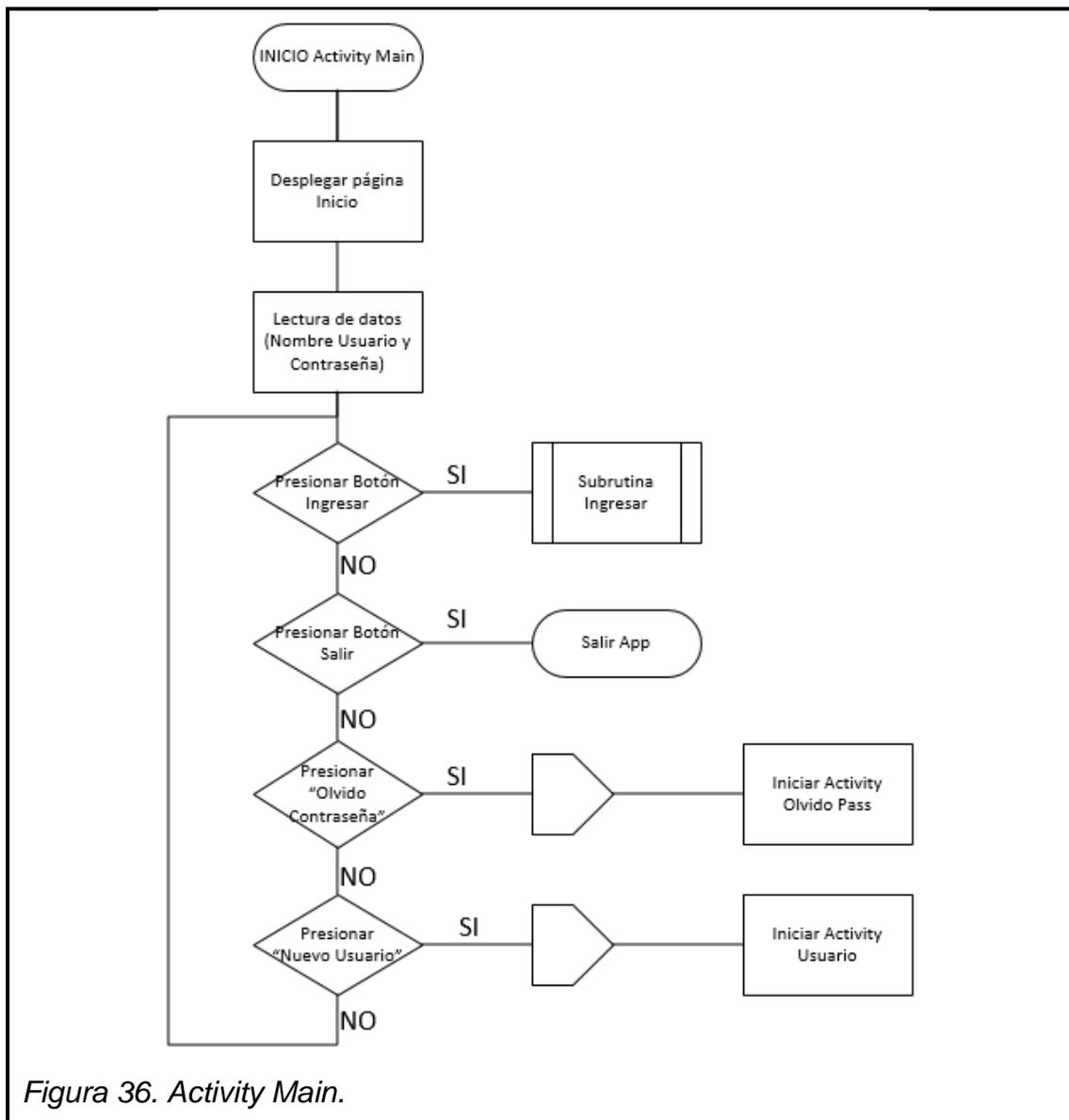


Recibirá datos de los cambios que presente según el movimiento del sensor, trabajará con una librería propia de Arduino llamada ADXL345 que es proporcionada por el fabricante, si no se instancia dicha librería se desplegará un mensaje para comprobar las conexiones del sensor, caso contrario se ejecutará con normalidad y desplegará los resultados que se obtengan de la aceleración, como se muestra en la Figura 35.

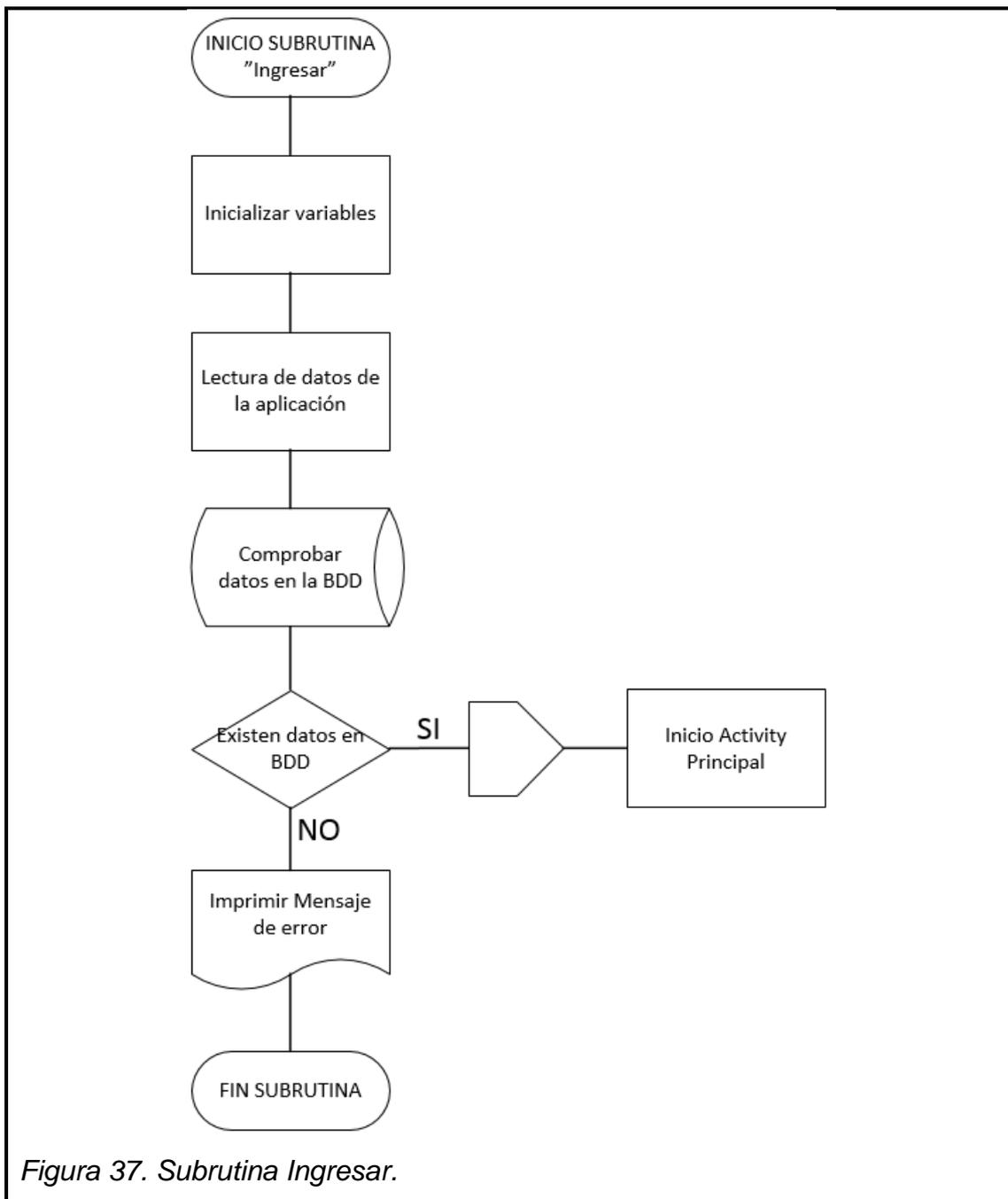


3.3.2 Software Android Aplicación

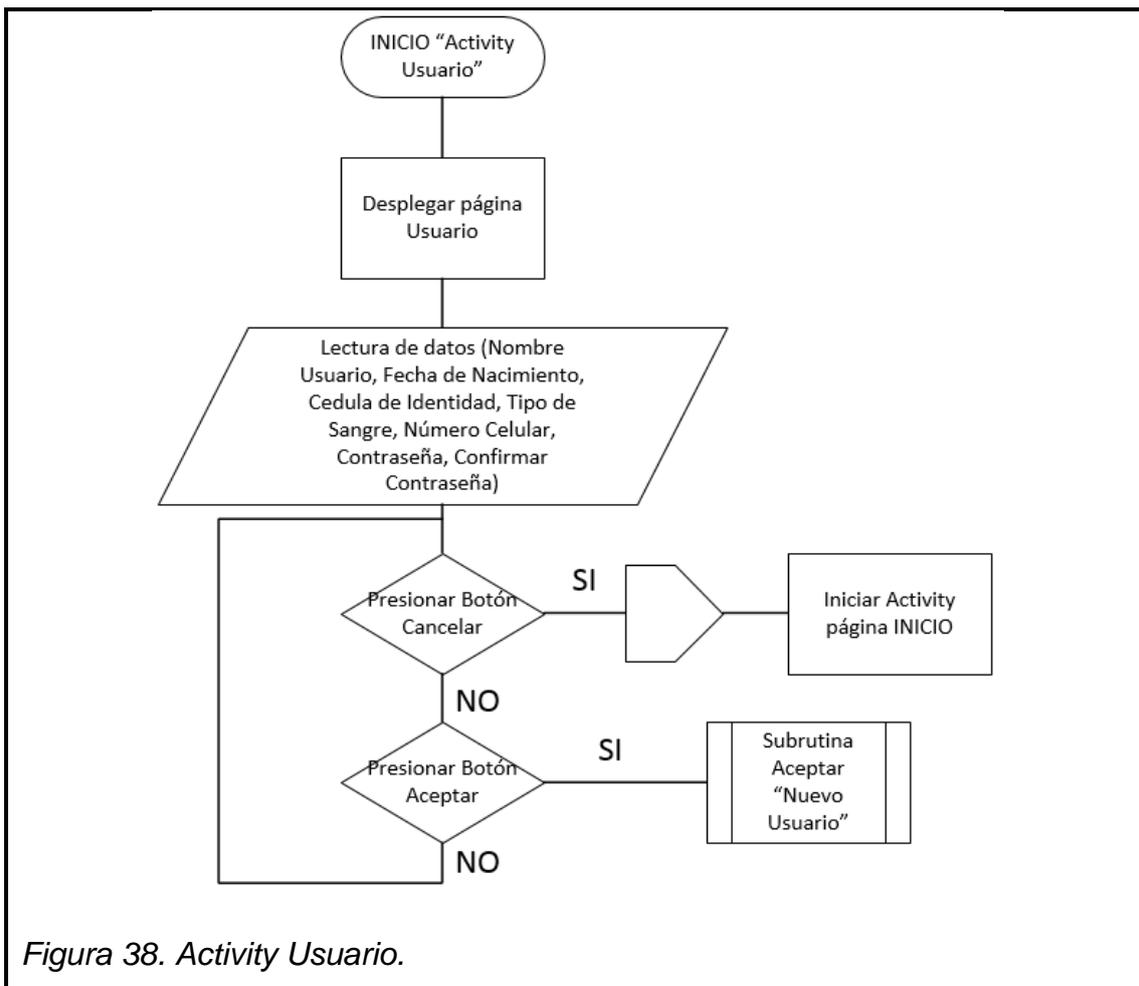
Se desplegará la página principal de la aplicación, la cual permitirá registrarse, autenticarse, salir o recuperar la contraseña de un usuario previamente creado, se desplegará la otra página principal si la autenticación fue exitosa o la página de “Nuevo Usuario” agregará a la BDD. Como se muestra en la Figura 36.



La Subrutina Ingresar comprobará en la BDD los datos ingresados para autenticar. Si los datos son correctos se desplegará la Activity principal de la aplicación, caso contrario regresará al "Activity Main" y desplegará un mensaje de error, como se muestra en la Figura 37.



Se desplegará una pantalla la cual se solicitará que ingrese la información básica del usuario, dicha información será guardada en la BDD, como se muestra en la Figura 38.



La subrutina Aceptar Nuevo Usuario validará si los campos donde se ingresará la contraseña y confirmar contraseña son iguales, si existe igualdad se guardará en la BDD caso contrario desplegará un mensaje de error, como se muestra en la Figura 39.

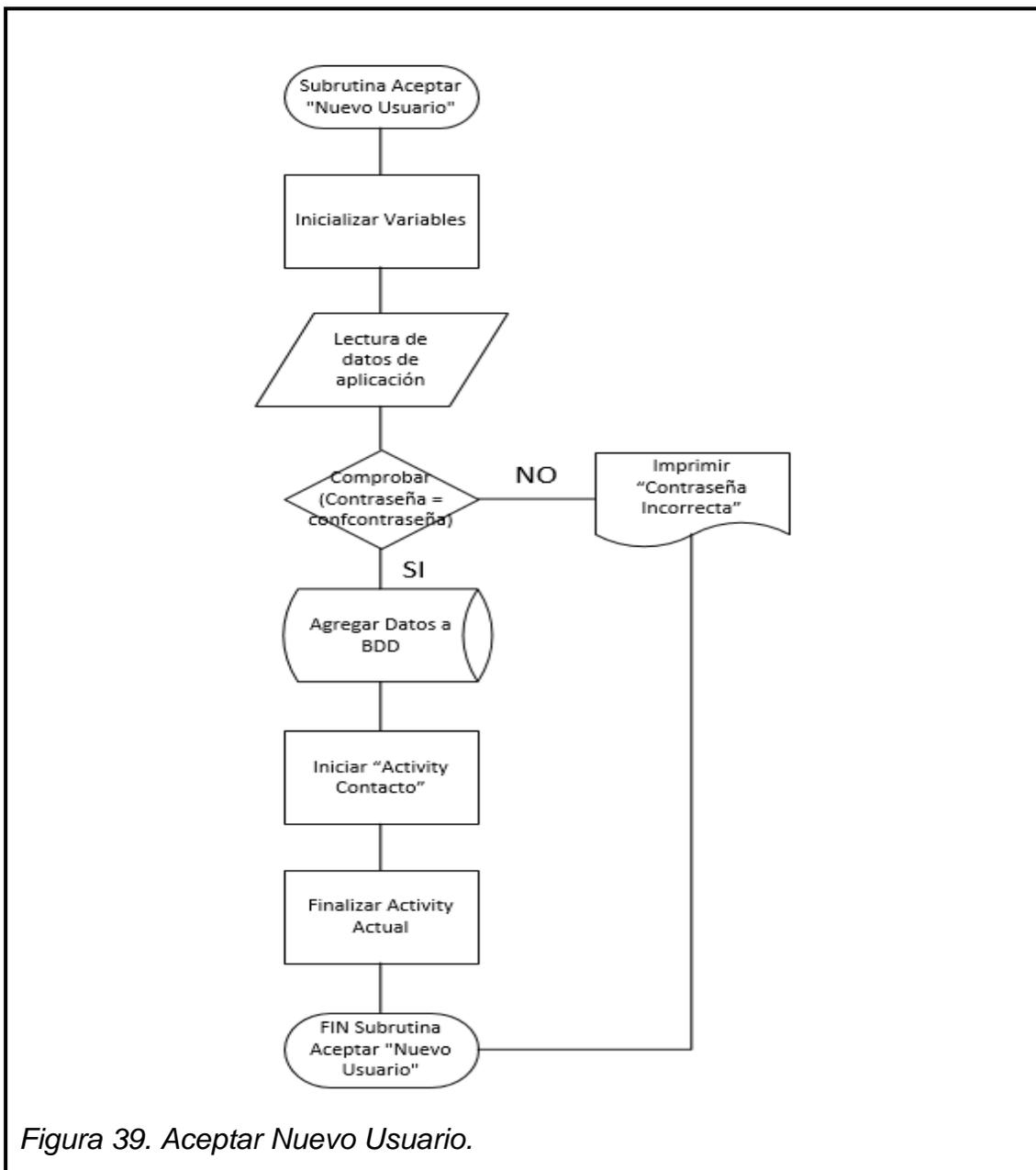
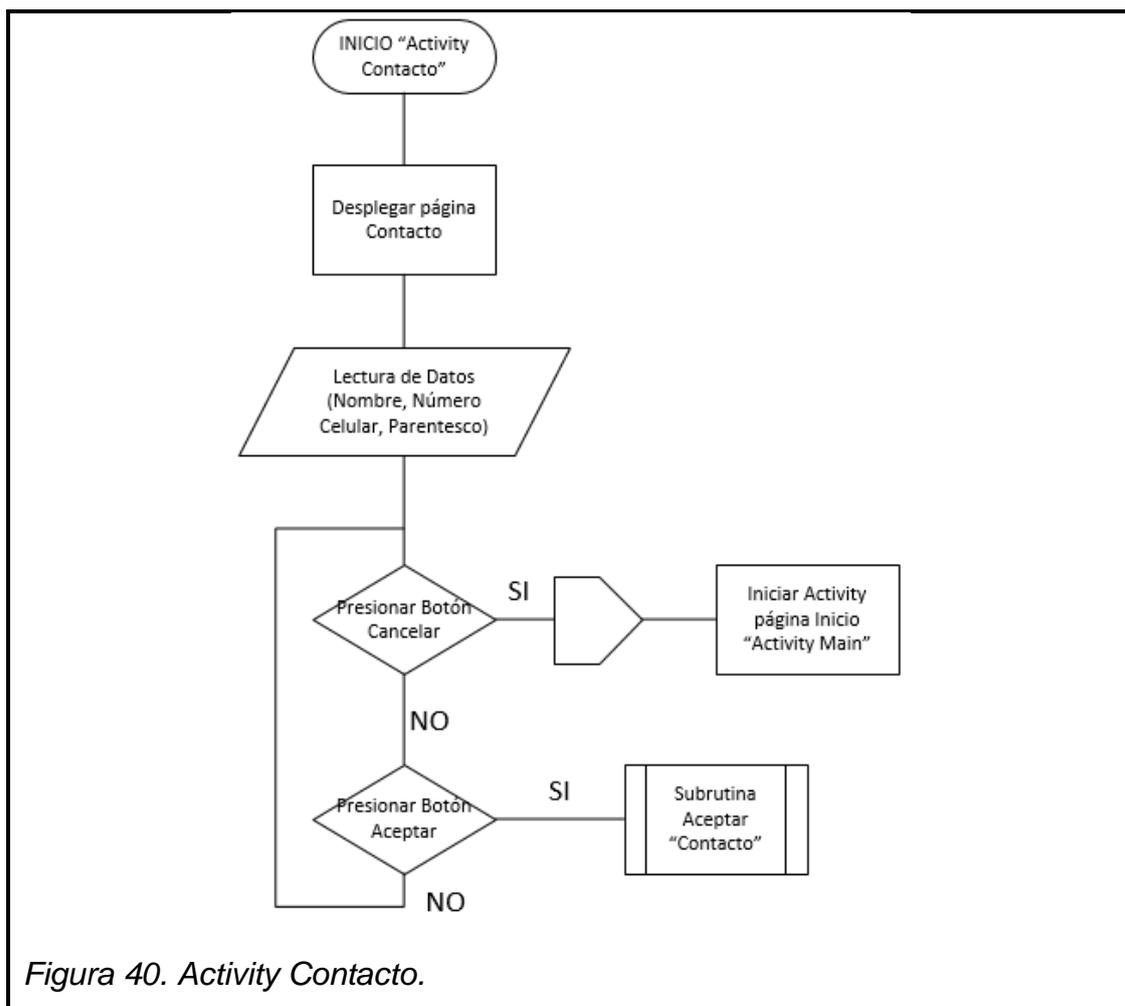
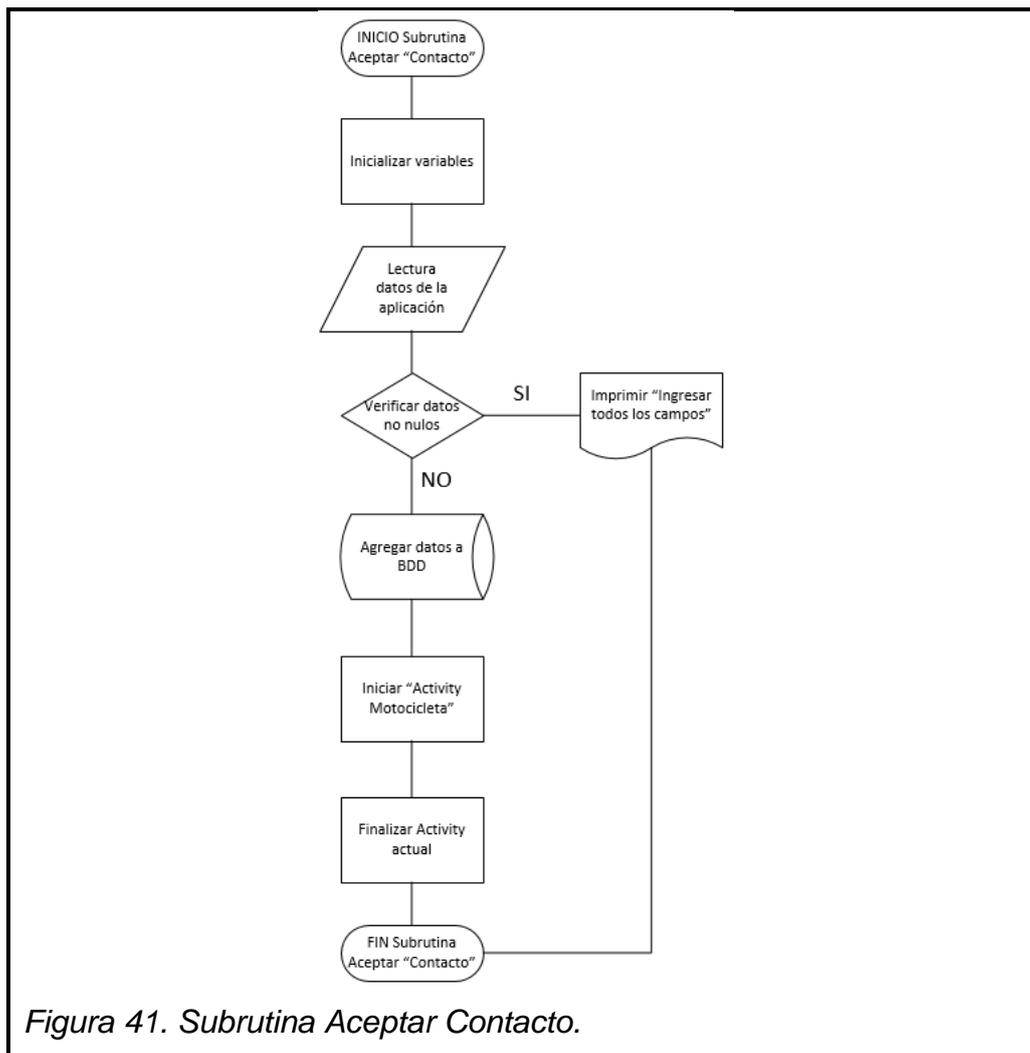


Figura 39. Aceptar Nuevo Usuario.

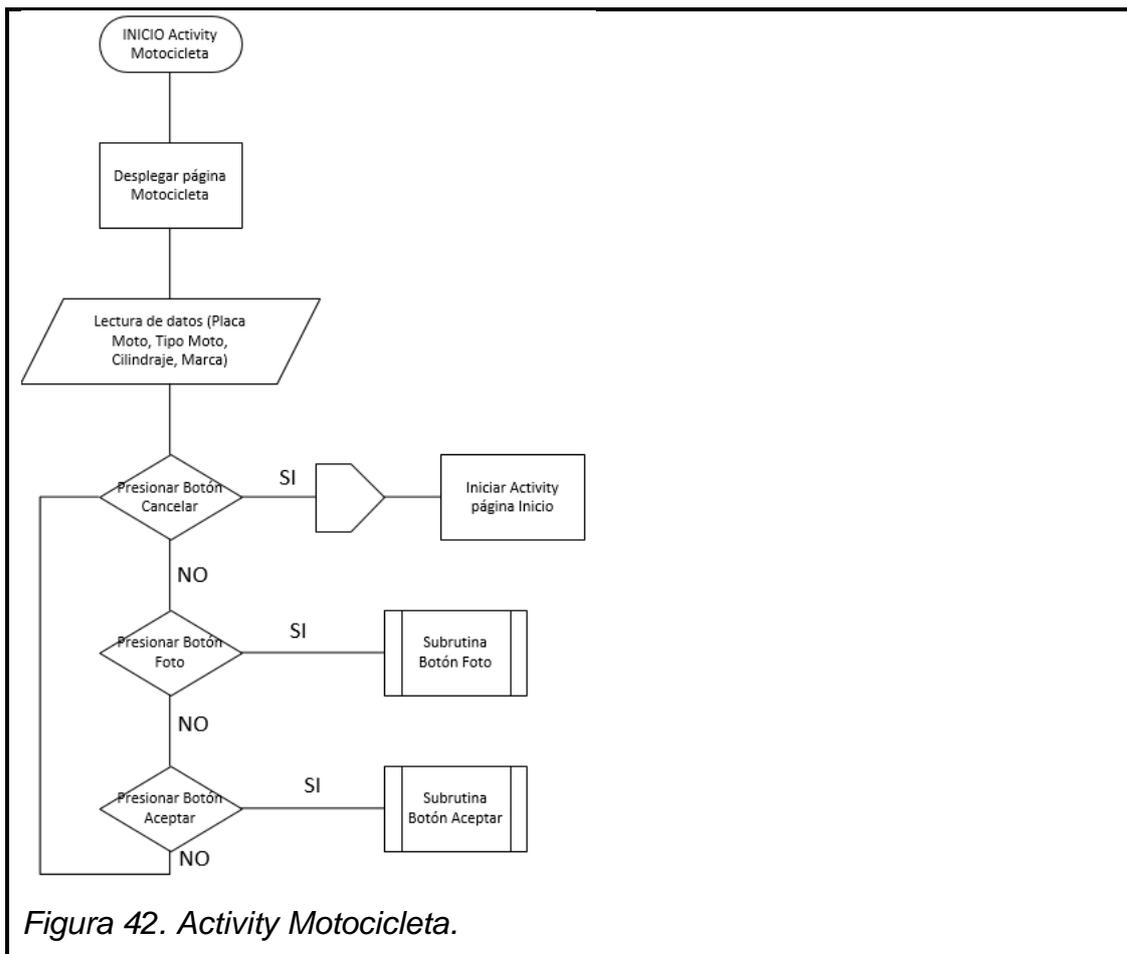
Activity Contacto se encargará de guardar información de una persona específica que agregará el usuario, para que en caso de un accidente, dicho contacto reciba un mensaje, como se muestra en la Figura 40.



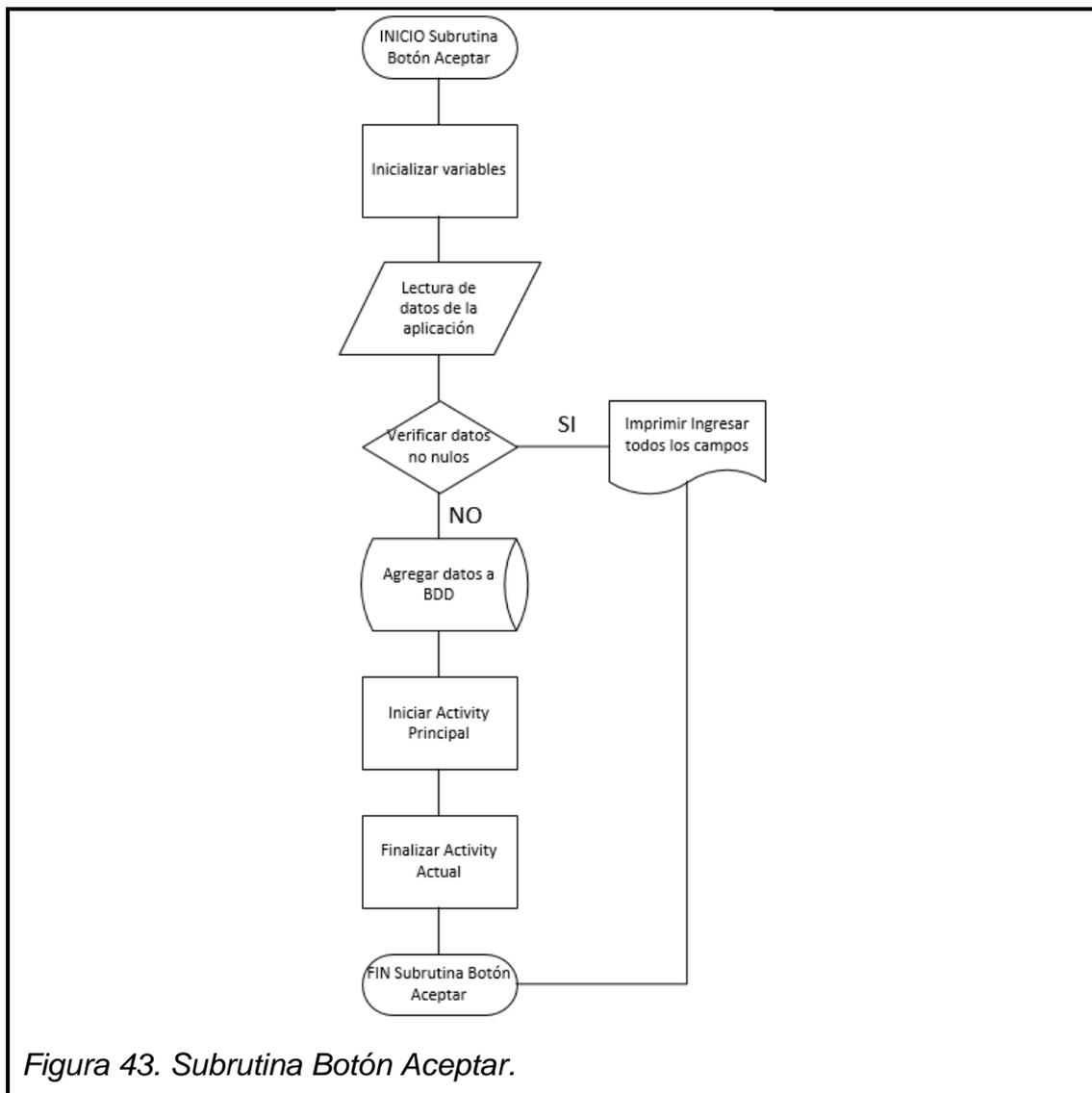
La subrutina Aceptar Contacto validará que todos los campos requeridos se encuentren llenos posteriormente procederá a guardar en la BBD de la aplicación, caso contrario se desplegará un aviso para que el usuario complete todos los campos, como se muestra en la Figura 41.



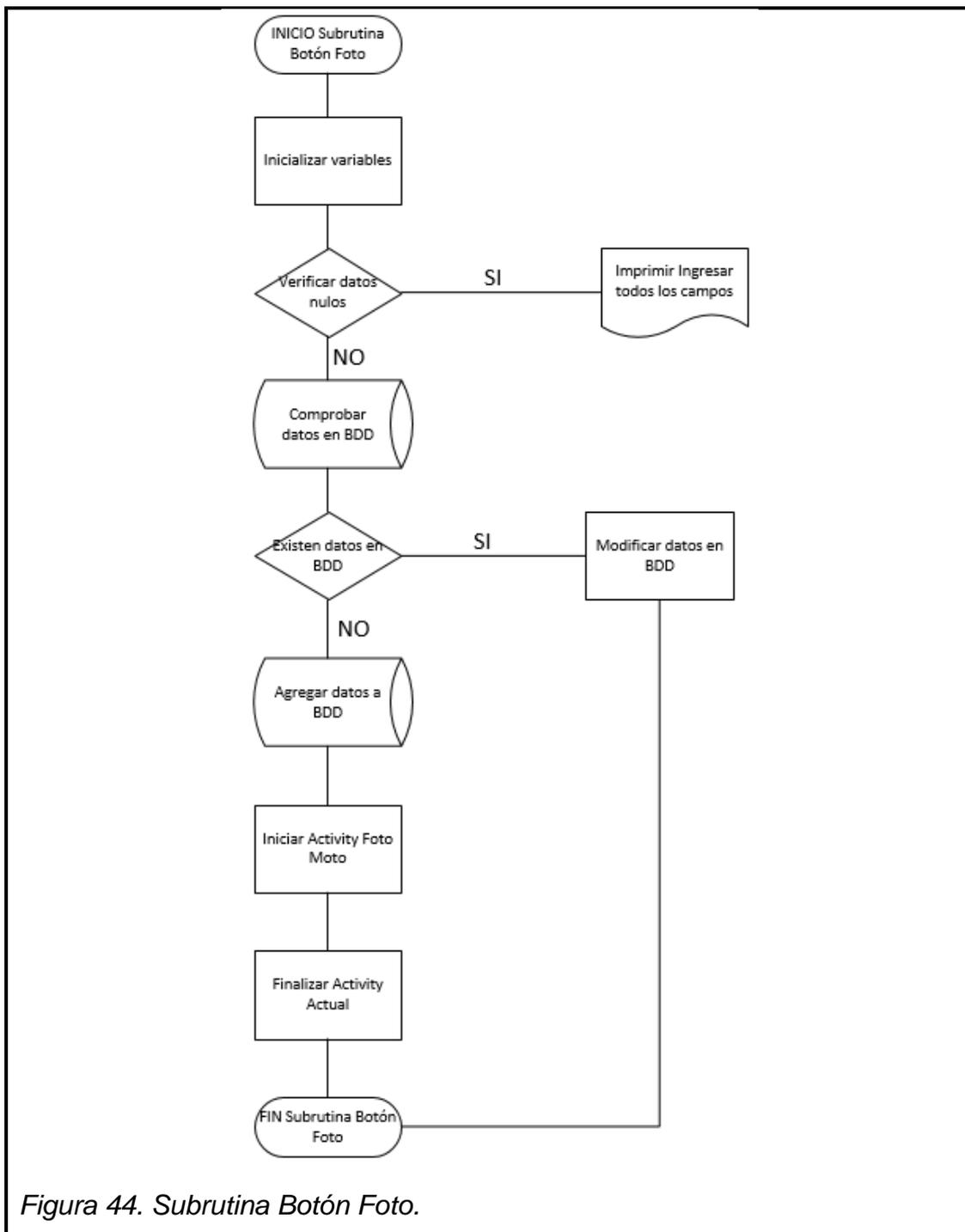
El Activity Motocicleta se desplegará para guardar información básica de la motocicleta del Usuario, tendrá la opción de poner una foto, como se muestra en la Figura 42.



Subrutina Botón Aceptar validará que todos los campos requeridos en el Activity Motocicleta estén llenos, guardará en la BDD, en caso de que algún campo o todos los campos se encontraran vacíos se mostrará un mensaje para que el usuario ingrese la información solicitada, como se muestra en la Figura 43.



La subrutina Botón Foto determinará si la información de la moto se encuentra agregada a la BDD, si no agregará toda la información de la moto con los datos propuestos en el Activity Motocicleta y verificará los campos que no se encuentren vacíos, posteriormente ingresará al Activity Foto Moto, como se muestra en la Figura 44.



Si el usuario deseará una foto de la moto podrá tomar la fotografía y guardará en la BDD, si ya existiese una foto podrá cambiar la foto, como se muestra en la Figura 45.

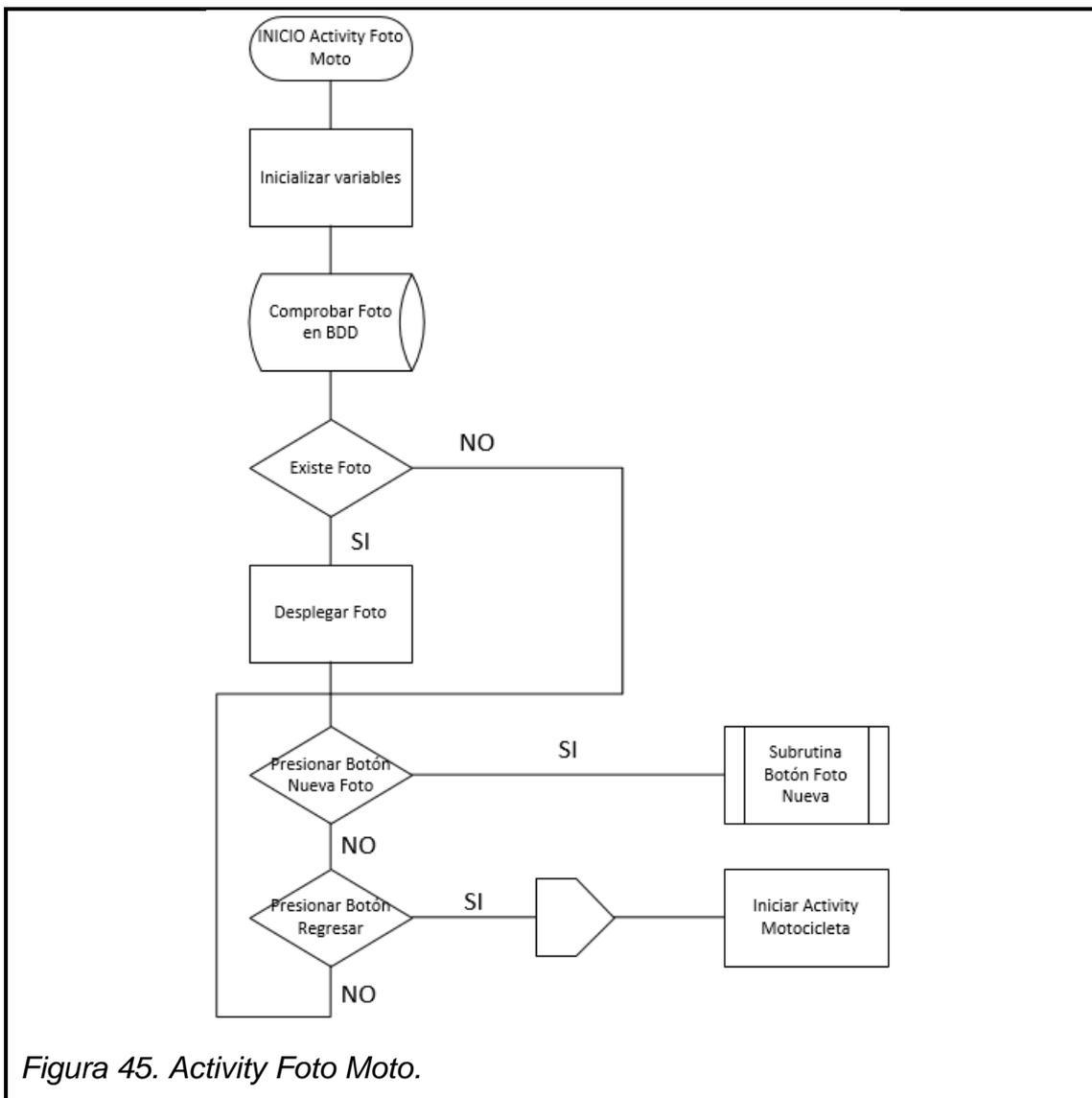
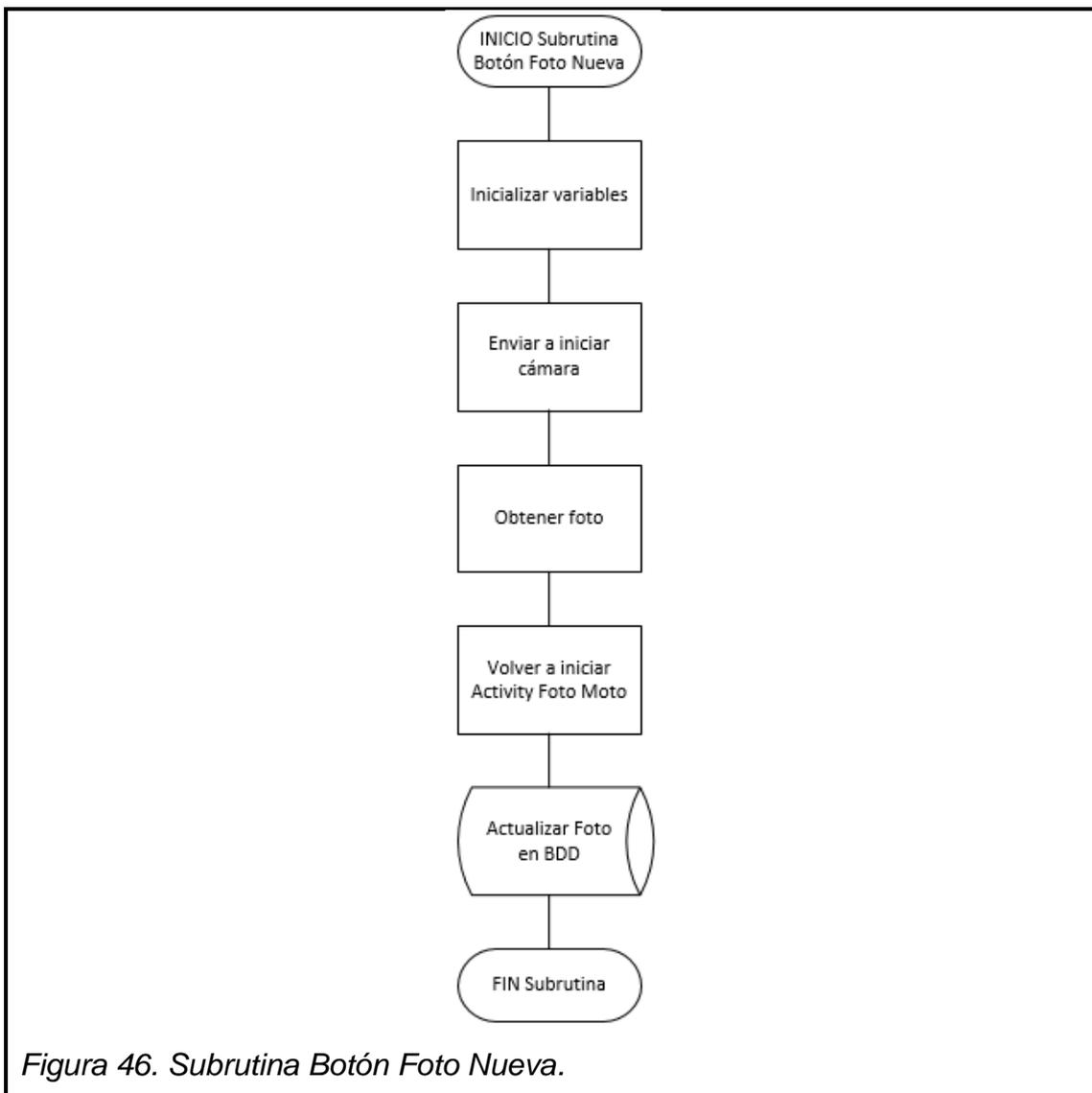
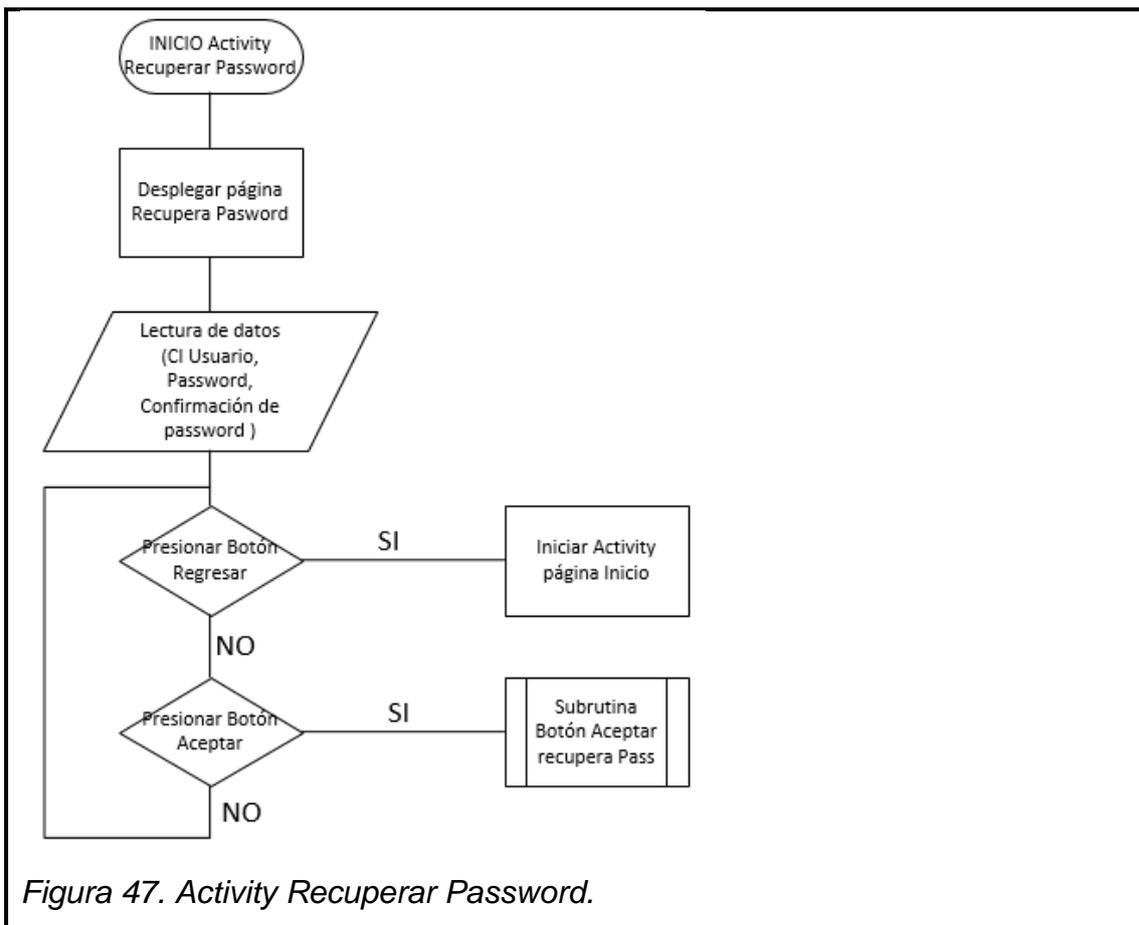


Figura 45. Activity Foto Moto.

Capturará la foto de la motocicleta del usuario y se guardará en la BDD, una vez que se obtiene la foto se desplegará el Activity Foto Moto para seguir completando o modificando la información ingresada, como se muestra en la Figura 46.



Permitirá cambiar la contraseña olvidada mediante el número de cédula que está en la BDD, realizará una lectura de datos y revisando si el usuario se encontrará ya registrado en la BDD del sistema, como se muestra en la Figura 47.



Se validará que todos los campos requeridos en el Activity Recuperar Password estén llenos, guardará en la BDD, verificará si existe algún campo vacío y se mostrará un mensaje para que el usuario ingrese la información solicitada anteriormente. Si el usuario no se encuentra registrado en la BDD se desplegará un mensaje, caso contrario se actualizará la BDD de la aplicación y se volverá a ejecutar el Activity Principal, como se muestra en la Figura 48.

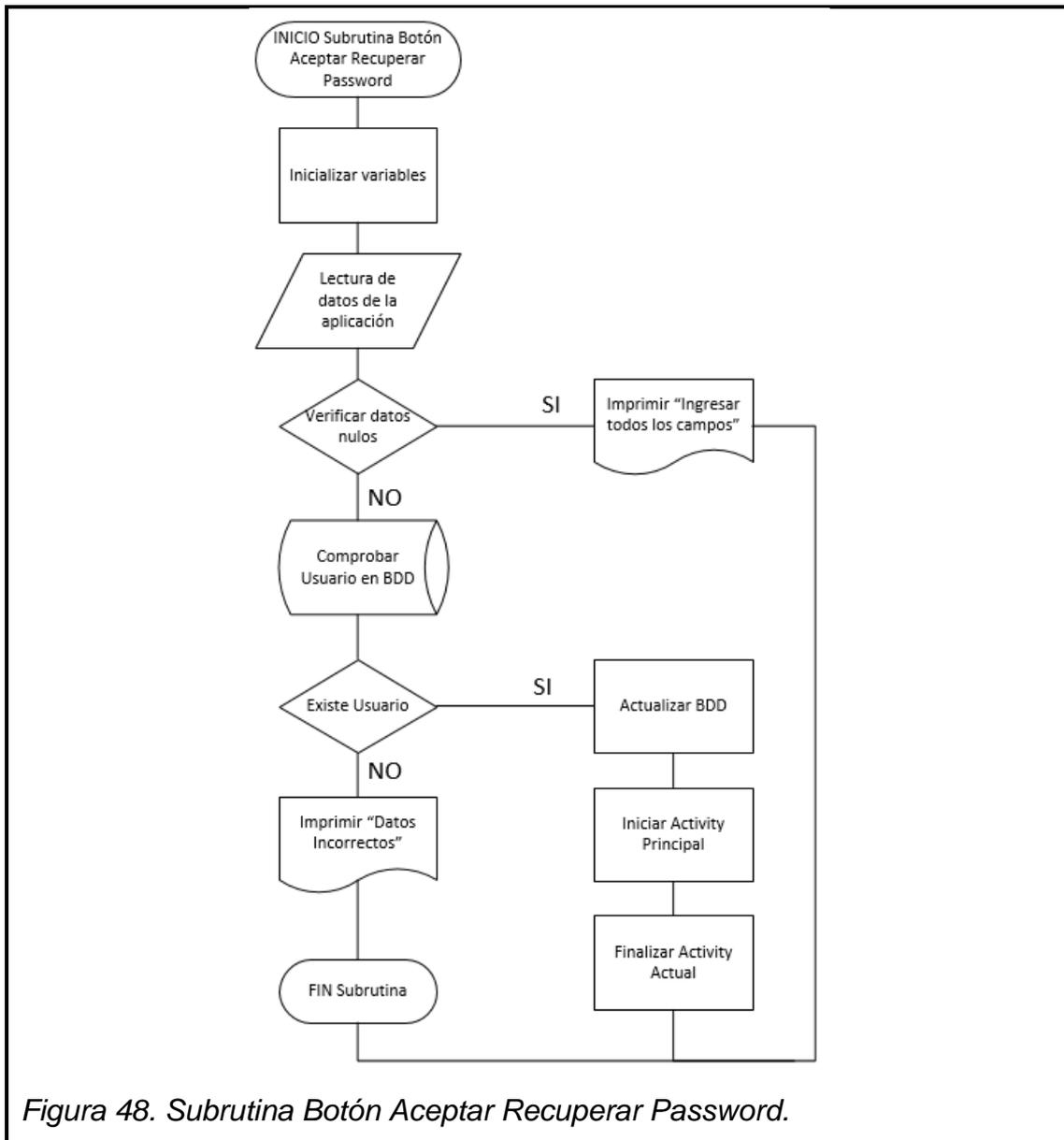


Figura 48. Subrutina Botón Aceptar Recuperar Password.

Se activará el bluetooth del celular y buscará a todos los dispositivos que se encuentren cerca del mismo, posteriormente procederá con la conectividad, como se muestra en la Figura 49.

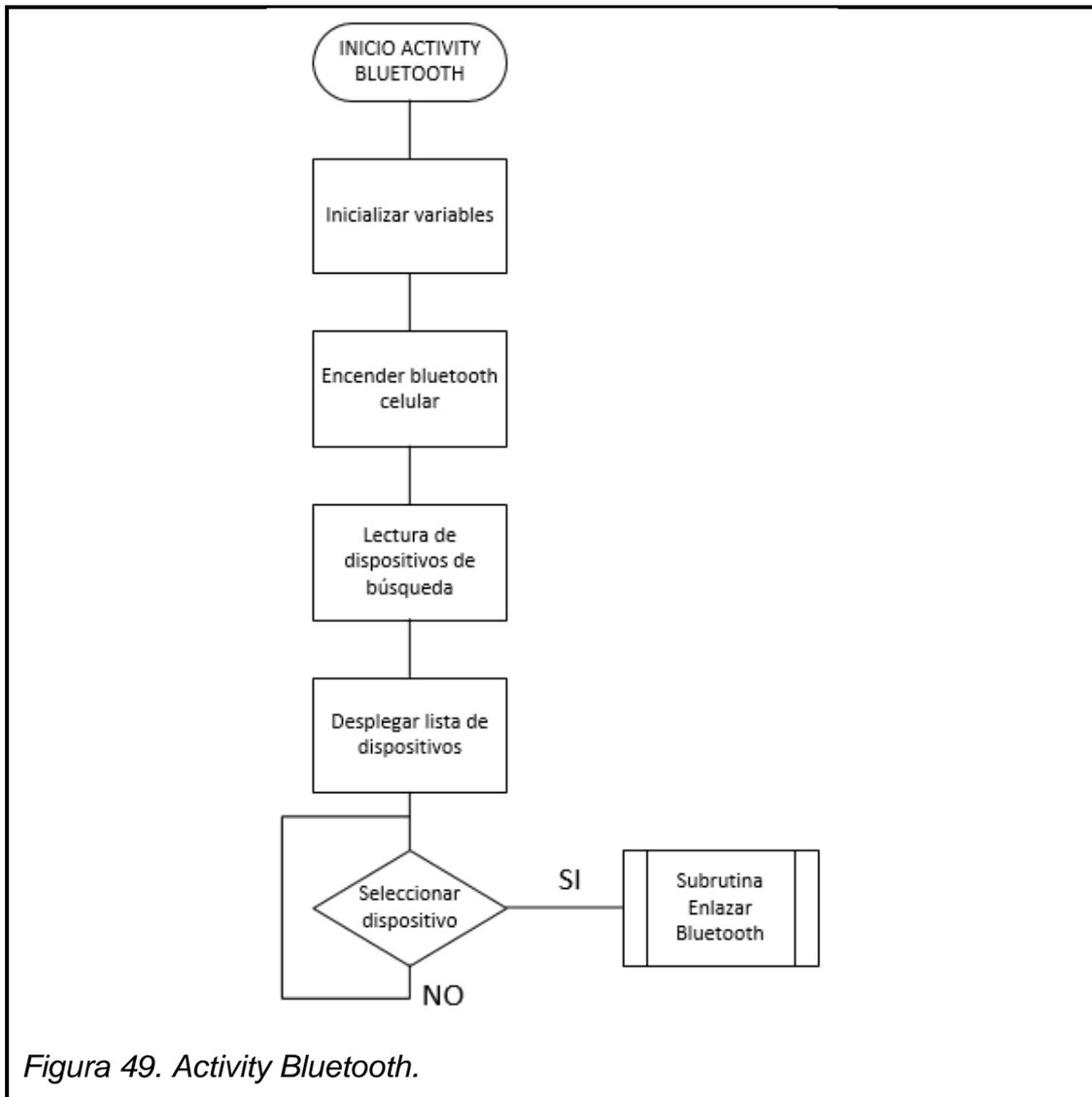
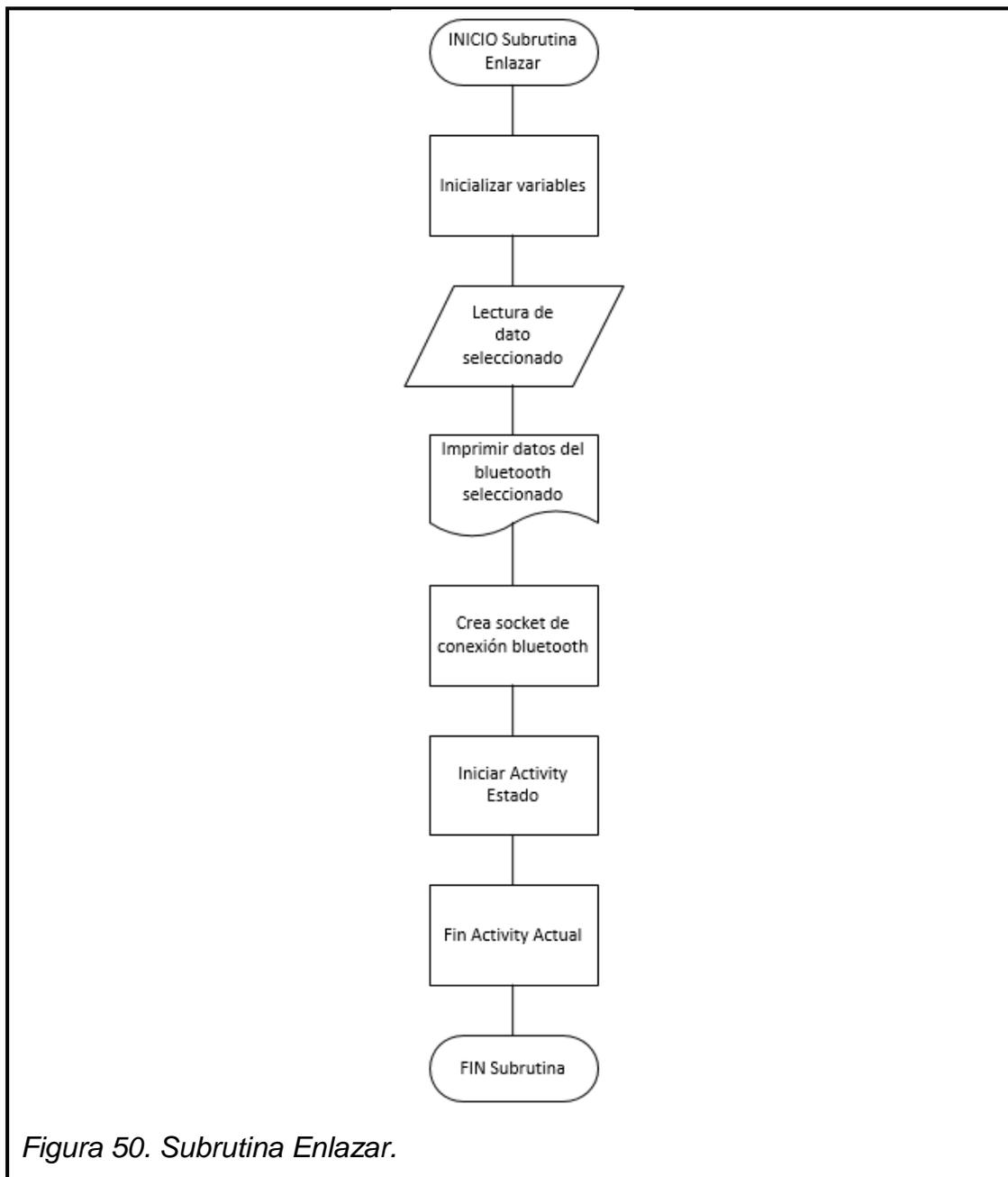
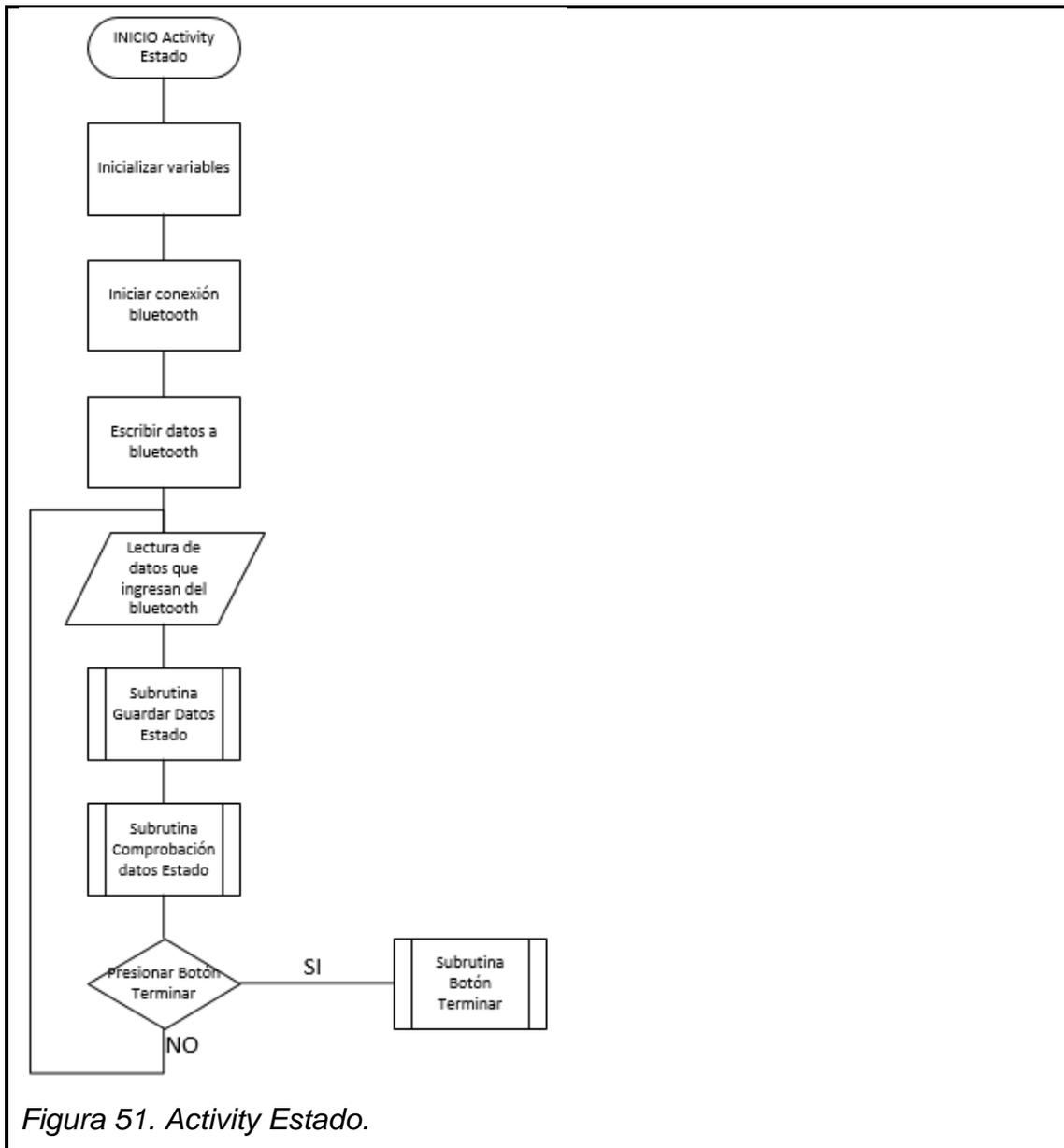


Figura 49. Activity Bluetooth.

Enlazará el bluetooth y enviará a ejecutar el Activity Estado, como se muestra en la Figura 50.



Una vez enlazados los dispositivos, los datos que ingresan del bluetooth serán registrados y se procederá a guardar o comprobar el estado hasta que el usuario finalice la sesión, como se muestra en la Figura 51.



Leerá los datos ingresantes del casco, los cuales se guardarán en la BDD y verificará la cantidad de datos ingresados hasta 300 muestras, como se muestra en la Figura 52.

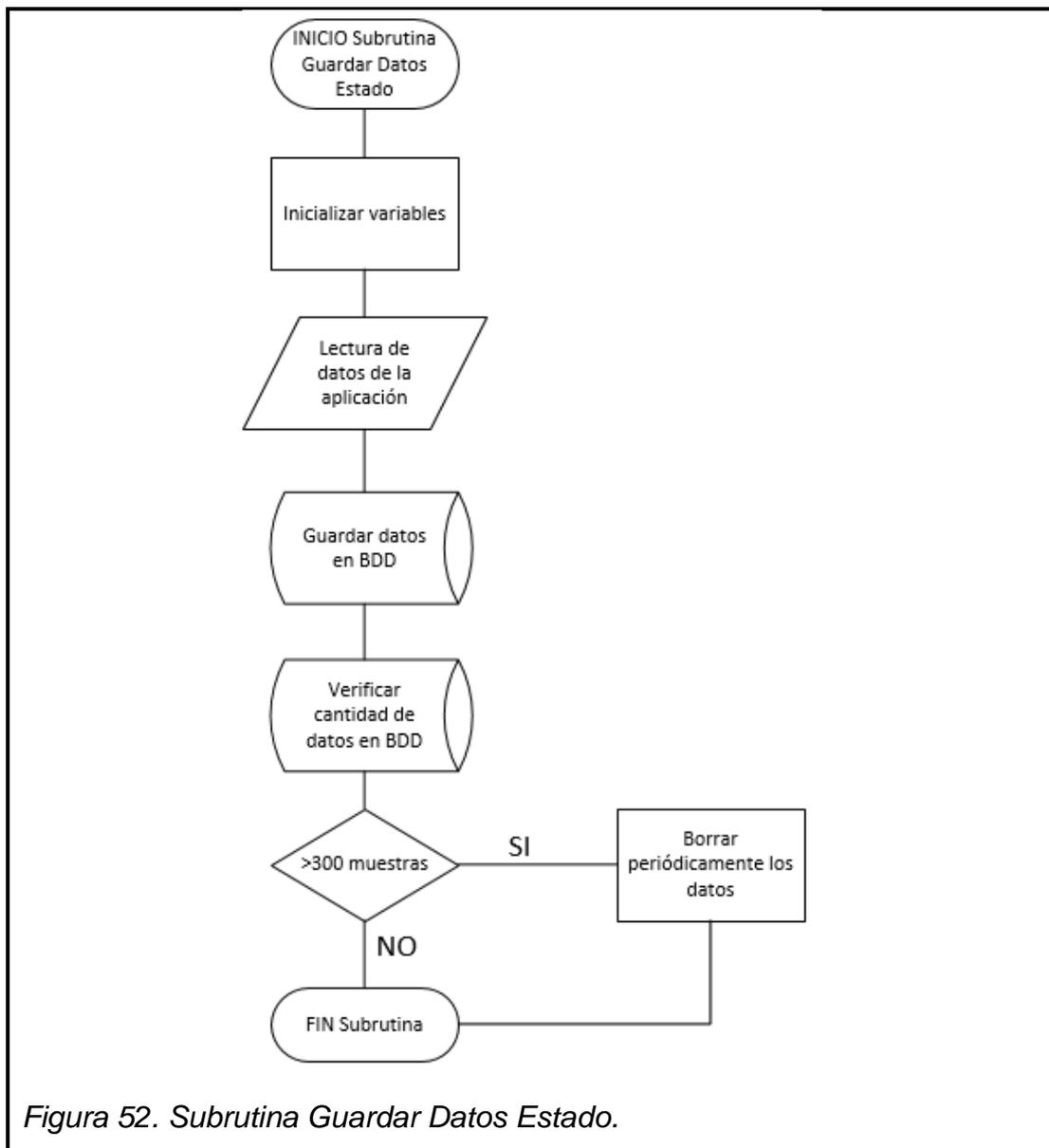


Figura 52. Subrutina Guardar Datos Estado.

Con los datos que se guardaron previamente, verificará si existe una variación de datos, si existe se enviará el mensaje correspondiente al contacto predeterminado, caso contrario continuará con la aplicación, como se muestra en la Figura 53.

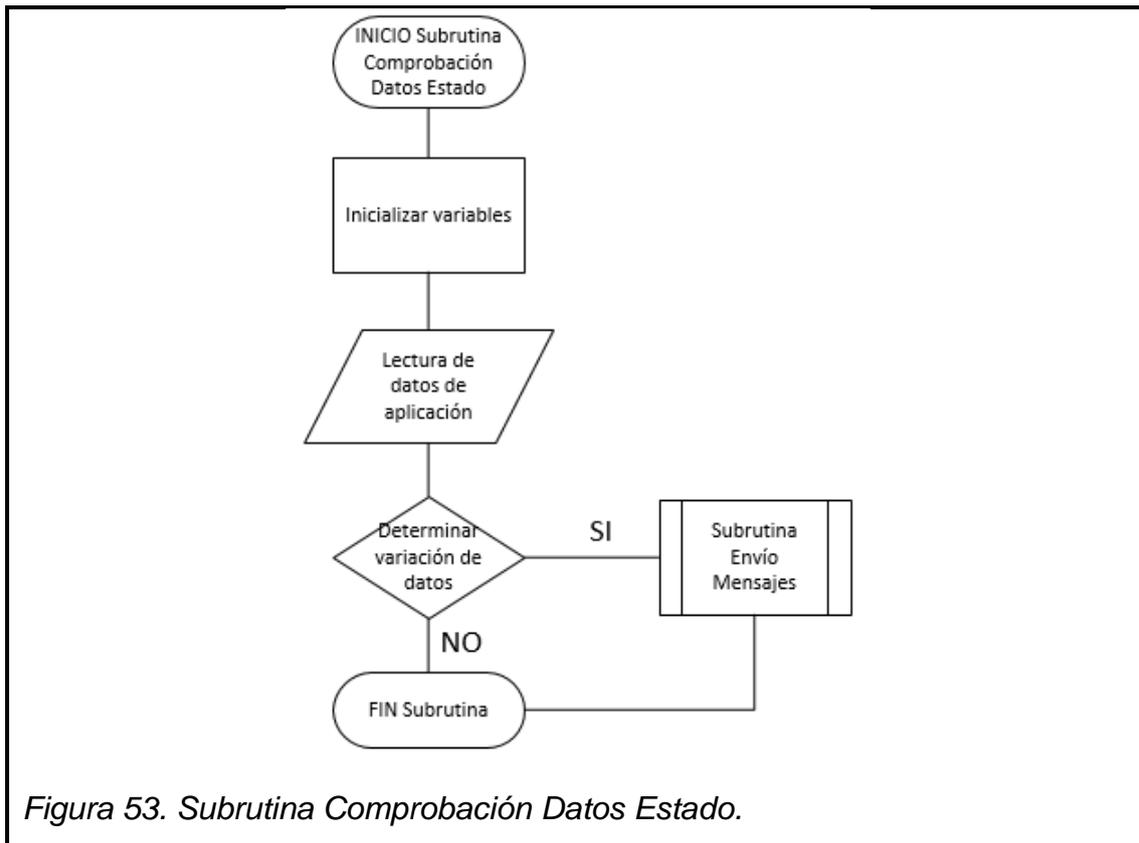
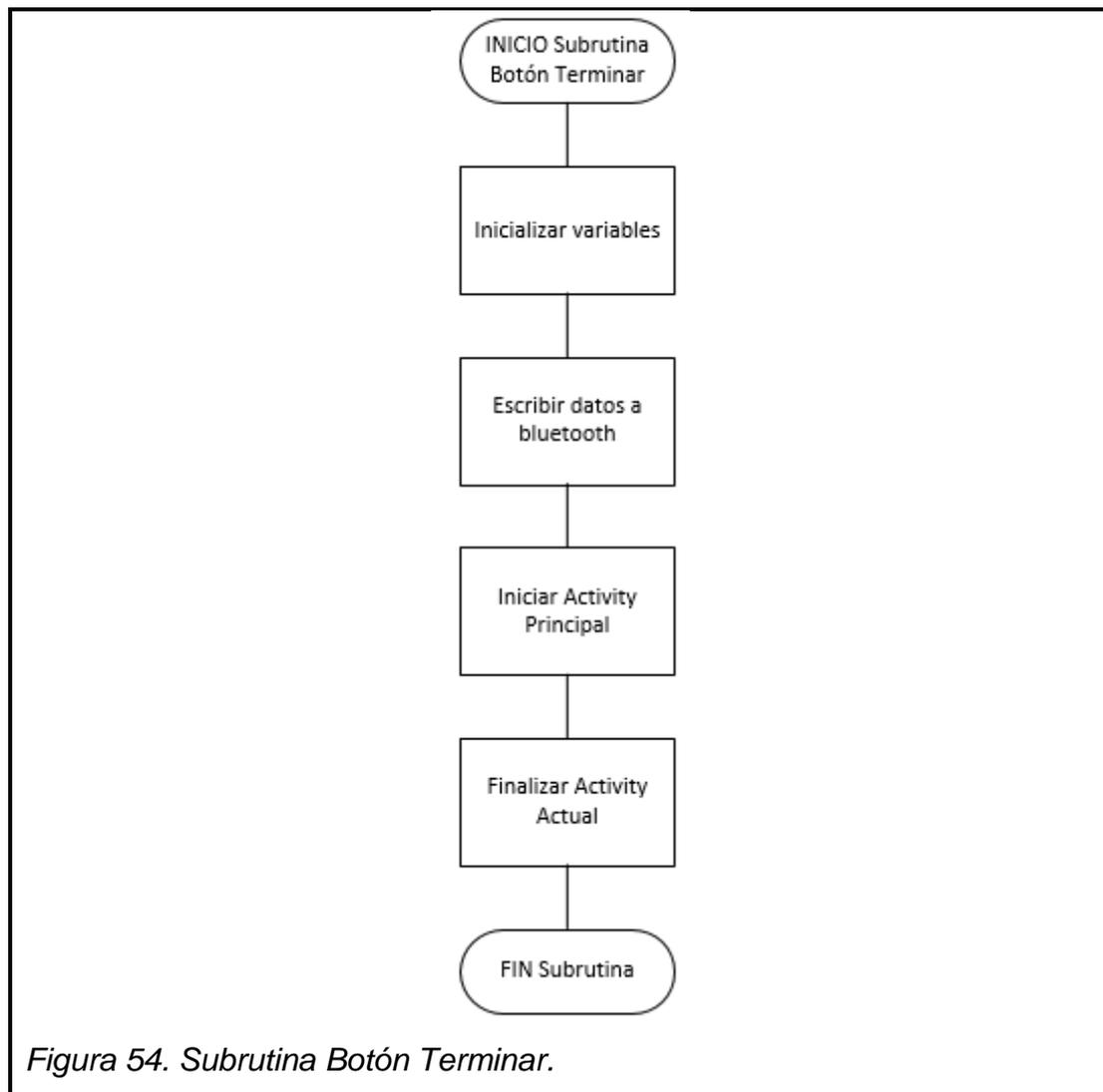
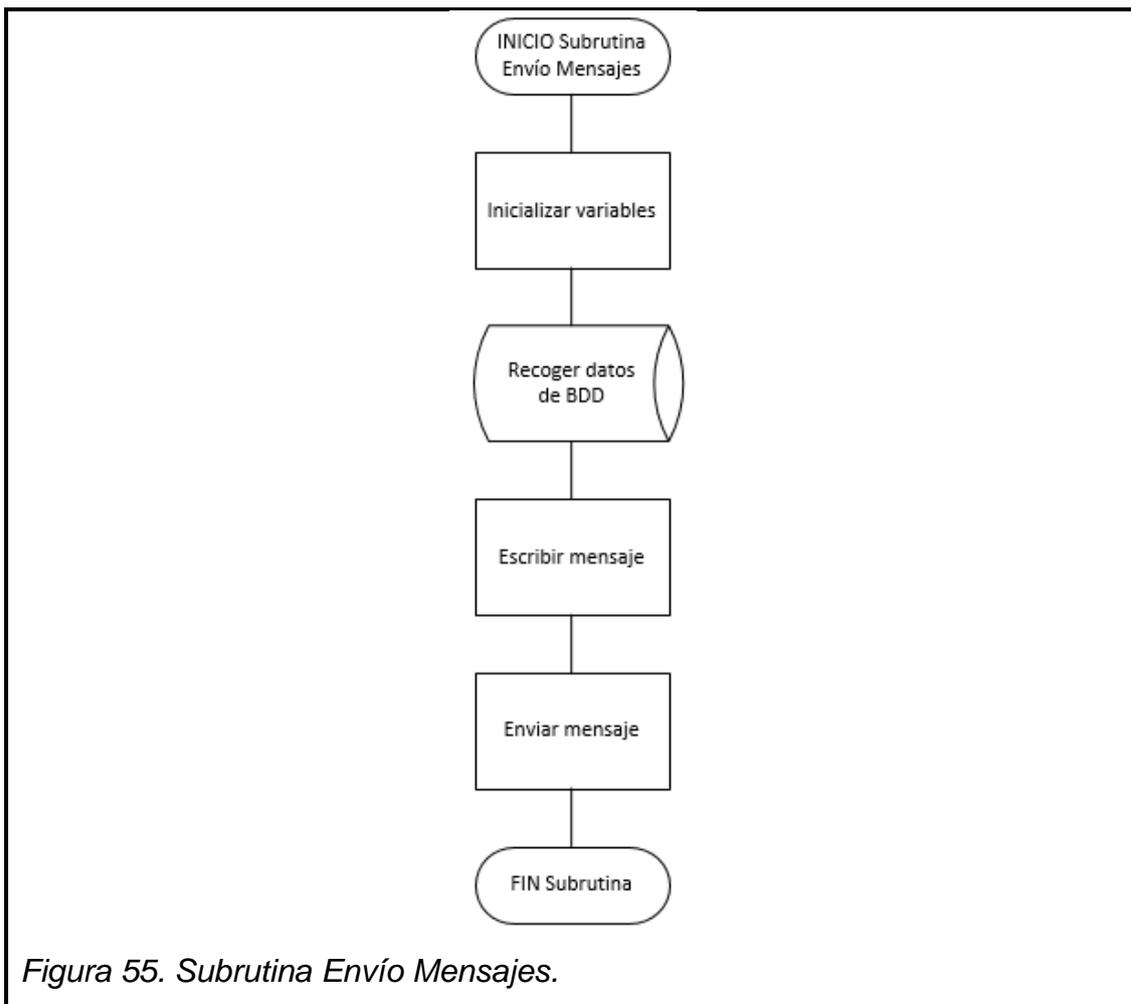


Figura 53. Subrutina Comprobación Datos Estado.

Escribirá en el bluetooth los datos que fueron agregados o comprobados en el Activity Estado, luego procederá a desplegar el Activity Principal para seguir utilizando la aplicación, como se muestra en la Figura 54.



Enviaré un mensaje en caso de que existiera un accidente al contacto que se guardó en la BDD y que fue ingresado por el usuario, como se muestra en la Figura 55.



La página principal de la aplicación se encargará de direccionar a los demás Activity, dependiendo de opción que elija el usuario y así continuar con el resto de la aplicación, como se muestra en la Figura 56.

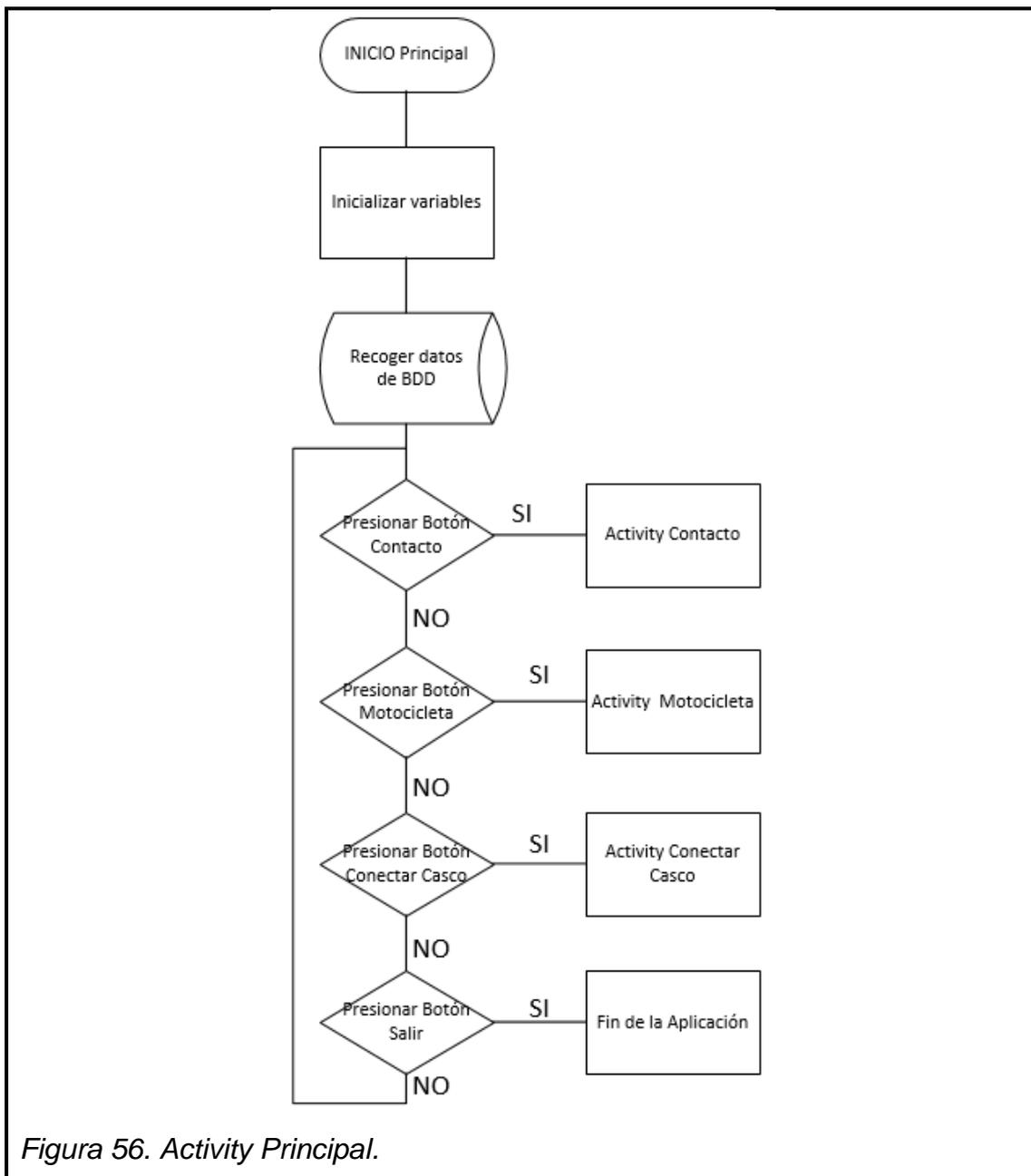
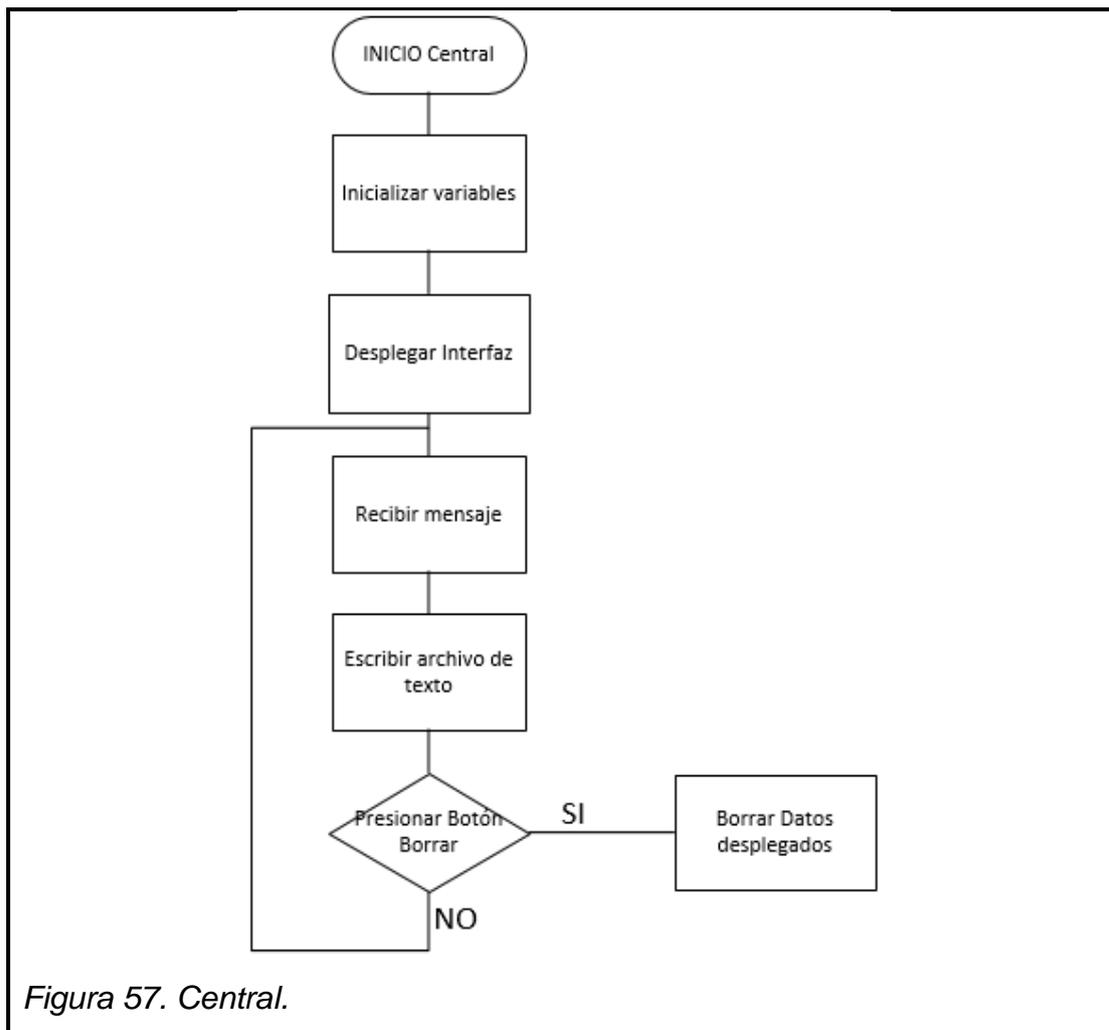


Figura 56. Activity Principal.

Recibirá los datos obtenidos de los sensores e imprimirá en la pantalla un mensaje si acontece un evento extraño. Además contará con un botón que servirá para borrar todos los datos desplegados, como se muestra en la Figura 57.



3.3.3 Diseño de Base de Datos

El diseño de la base de datos a utilizarse, parte del análisis de requerimientos de los datos necesarios a almacenar, que son los cuales permitirán fácil interacción con la aplicación, creación de campos primarios, relaciones entre tablas, entre otros.

Al utilizar bases de datos móviles, se debe establecer la cantidad de datos primordiales a almacenar por no tener disponibilidad de memoria de almacenamiento. Las tablas y campos de la base de datos móvil según el análisis de requerimientos son las siguientes:

Tabla Usuario, a contener columna “Cedula” el registro principal y clave principal de la tabla, columna “Name”, el nombre del usuario, no nulo, columna “FechaNacimiento”, la fecha de nacimiento del usuario, no nulo, columna “TipoSangre”, el tipo de sangre del usuario, no nulo, columna “Celular”, el número celular del usuario, no nulo, y columna “Password” con la contraseña de ingreso al sistema del usuario no nulo. En la Figura 58 se muestra la tabla Usuario.

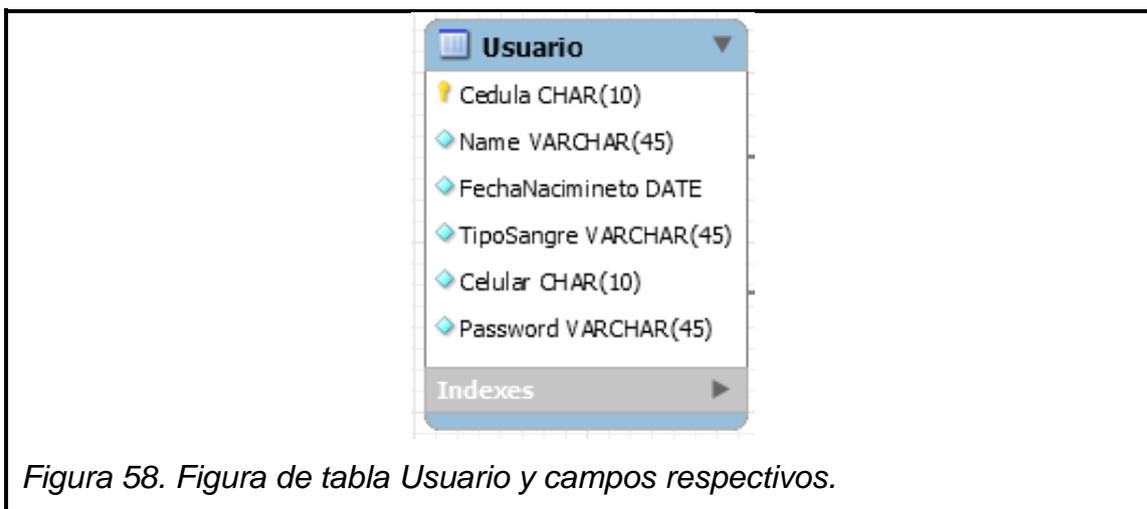


Tabla Contacto, a contener columna “PhoneNumber” el registro principal y clave principal de la tabla, columna “Name”, el nombre del contacto, no nulo, columna “Parnership”, el parentesco del contacto con usuario, no nulo y columna “Usuario_Cedula”, el campo relacional entre tablas Usuario y Contacto. En la Figura 59 se muestra la tabla Contacto.

Tabla Moto, a contener columna “idMoto” el registro principal y clave principal de la tabla, columna “Type”, el tipo de moto que el usuario ingresará, no nulo, columna “Marca”, la marca de moto que el usuario ingresará, no nulo, columna “Cilindraje”, el cilindraje de moto que el usuario ingresará, no nulo columna “Picture”, la foto de la moto que el usuario ingresará, y columna “Usuario_Cedula” el campo relacional entre tablas Usuario y Moto. En la Figura 60 se presenta la tabla Moto.

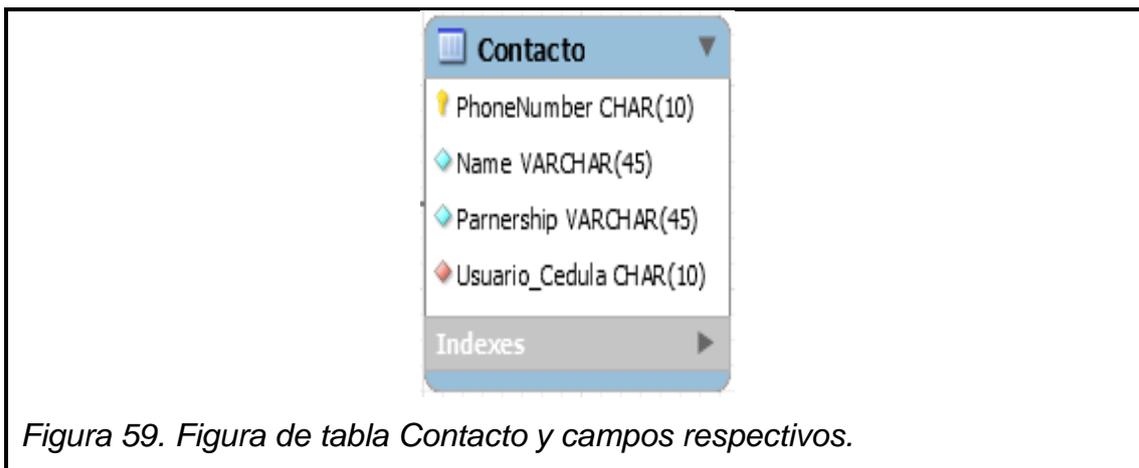


Figura 59. Figura de tabla Contacto y campos respectivos.

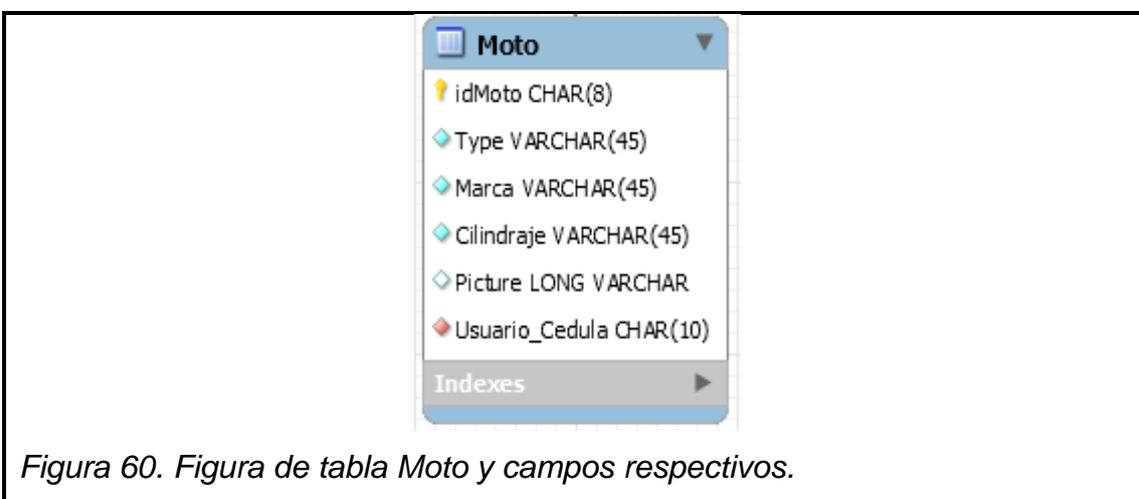


Figura 60. Figura de tabla Moto y campos respectivos.

Tabla Casco, a contener columna "idEvento" el registro principal y clave principal de la tabla auto incrementable, columna "Pulso", el dato del pulso que ingresará desde casco, no nulo, columna "Aceleración", el dato del aceleración que ingresará desde casco, no nulo, columna "Impacto", el dato del impacto que ingresará desde casco, no nulo, y columna "Usuario_Cedula" el campo relacional entre tablas Usuario y Casco. En la Figura 61 se presenta la tabla Casco.

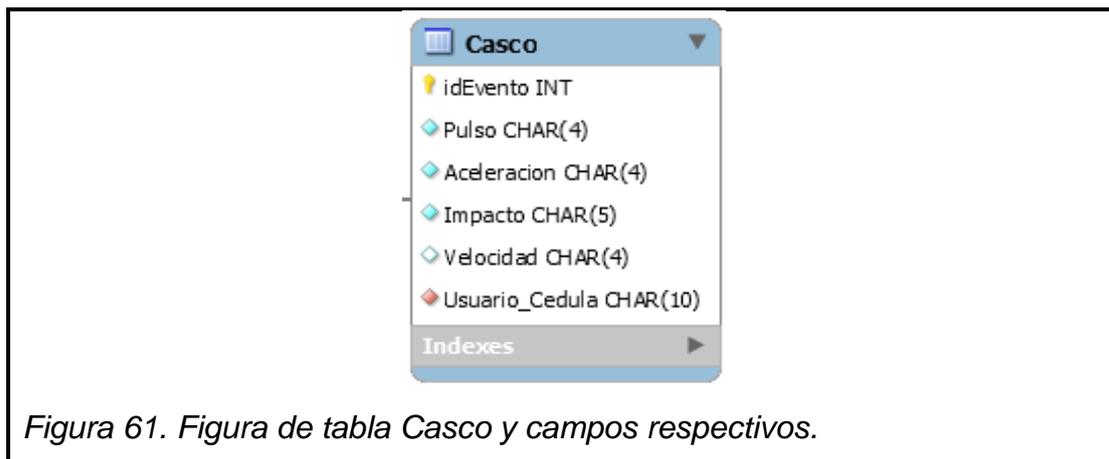


Figura 61. Figura de tabla Casco y campos respectivos.

El diagrama de base de datos contiene las tablas previamente descritas y relaciones creadas entre las tablas para su relación y dependencia. En la Figura 62 se muestra el diagrama total de la base de datos.

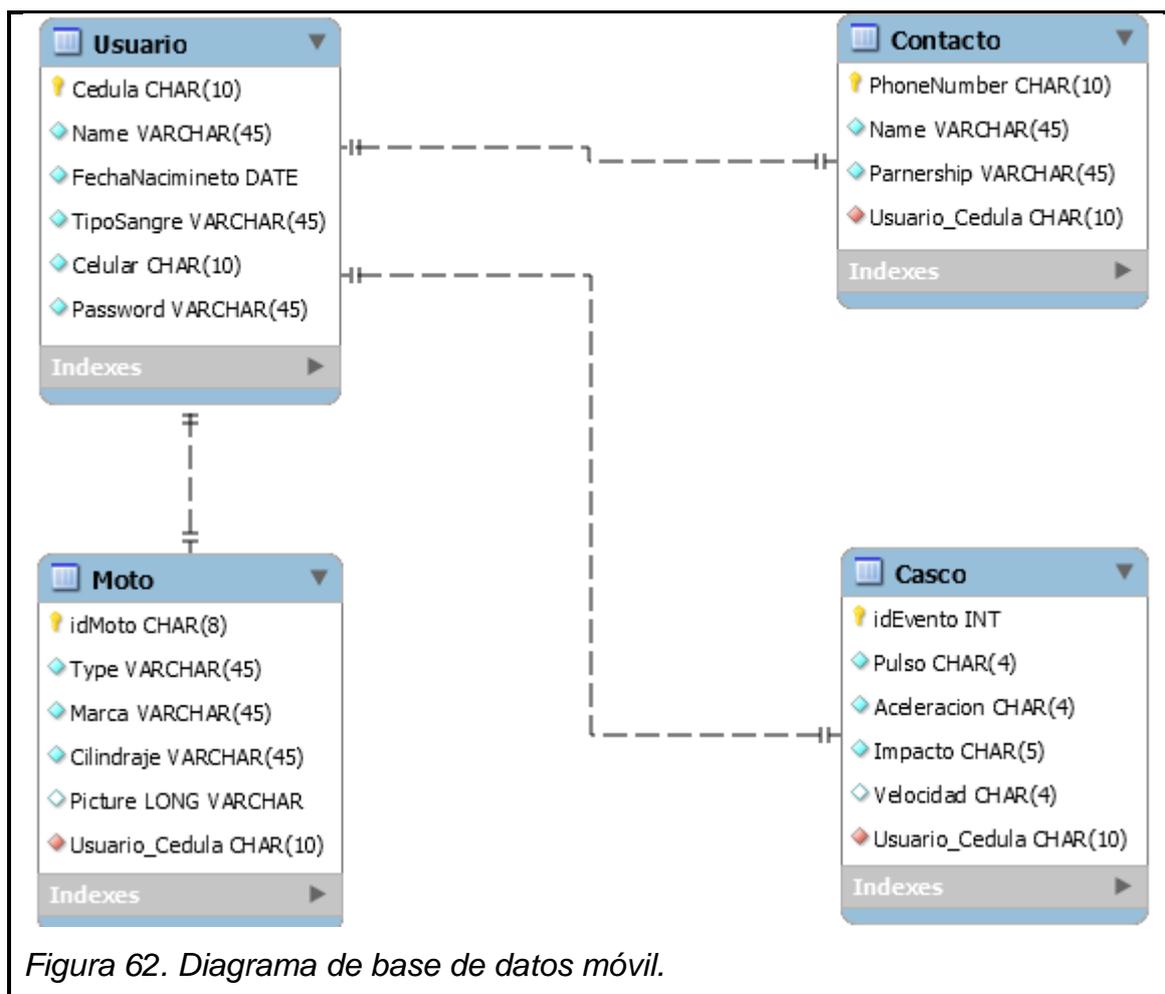


Figura 62. Diagrama de base de datos móvil.

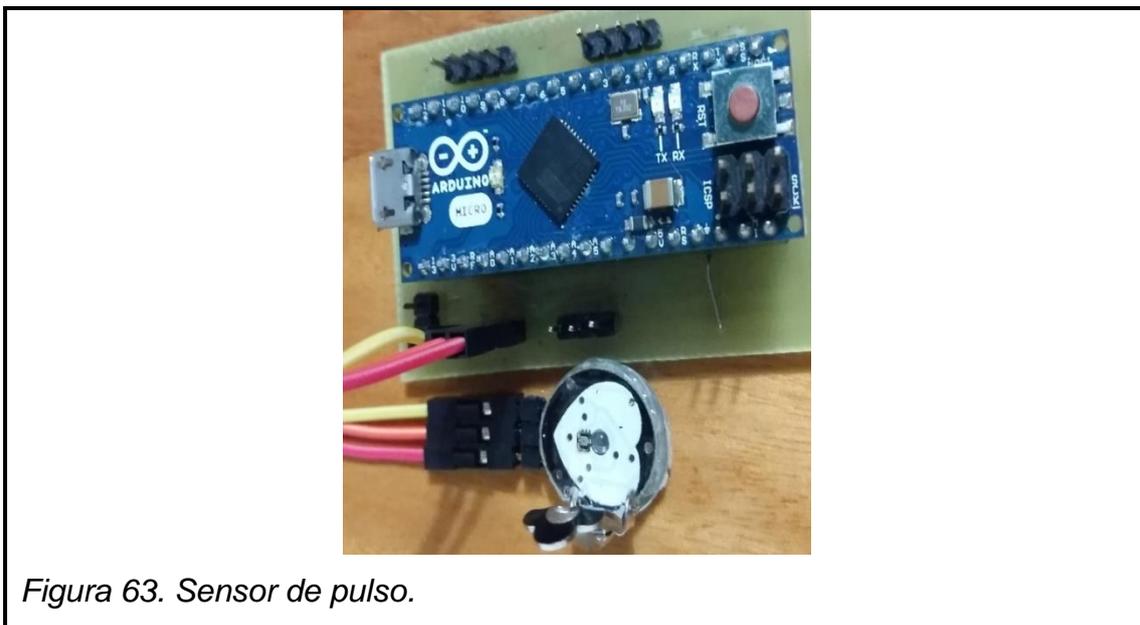
4. CAPÍTULO IV. Implementación del Prototipo

4.1 Implementación del Sistema

La implementación de los siguientes elementos se ha establecido siguiendo el diagrama realizado en el diseño.

4.1.1 Sensor de Pulso

Para comprobar la funcionalidad del sensor de Pulso SEN-11574 se realizó la configuración y pruebas en el protoboard. Posteriormente se procedió a colocar al sensor en una placa con el Arduino Micro, como se muestra en la Figura 63. Los pines propuestos para la comunicación entre el sensor y el Arduino se muestran en la Tabla 7.



El sensor de pulso consta con los siguientes pines, los cuales son conectados al Arduino Micro:

Tabla 7. Conexión del sensor de Pulso.

PIN SENSOR	PIN ARDUINO
1	Ground
2	Salida voltaje 5V
3	Pin Análogo 0 (A0)

4.1.2 Sensor de Sonido

Para verificar la funcionalidad del sensor y los datos que entregará, se procedió a comprobar en un protoboard. Y para la colocación del sensor en el casco se procederá a poner en una placa con el Arduino Micro, como se muestra en la Figura 64. Los pines propuestos para la comunicación entre el sensor de sonido y el Arduino se muestran en la tabla 8.

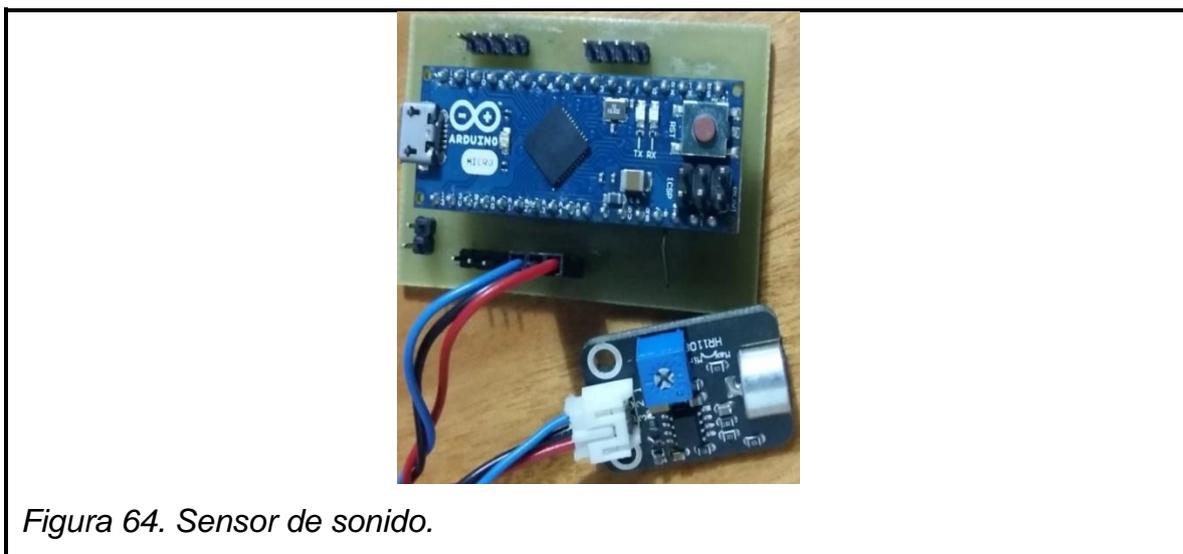


Figura 64. Sensor de sonido.

El sensor de sonido consta de tres pines:

Tabla 8. Conexión del sensor de Sonido.

PIN SENSOR	PIN ARDUINO
1	Salida voltaje 5V
2	Ground
3	Pin Análogo 1 (A1)

4.1.3 Sensor de Aceleración

Para configurar el funcionamiento del sensor de aceleración ADXL354 en el prototipo, se realizó las pruebas implementado primero en el protoboard. Para colocar en el casco se necesitará poner al sensor en la placa con el Arduino Micro, como se muestra en la Figura 65. Estableciendo la conexión entre el Arduino y el sensor con los pines que se especifican en la tabla 9.

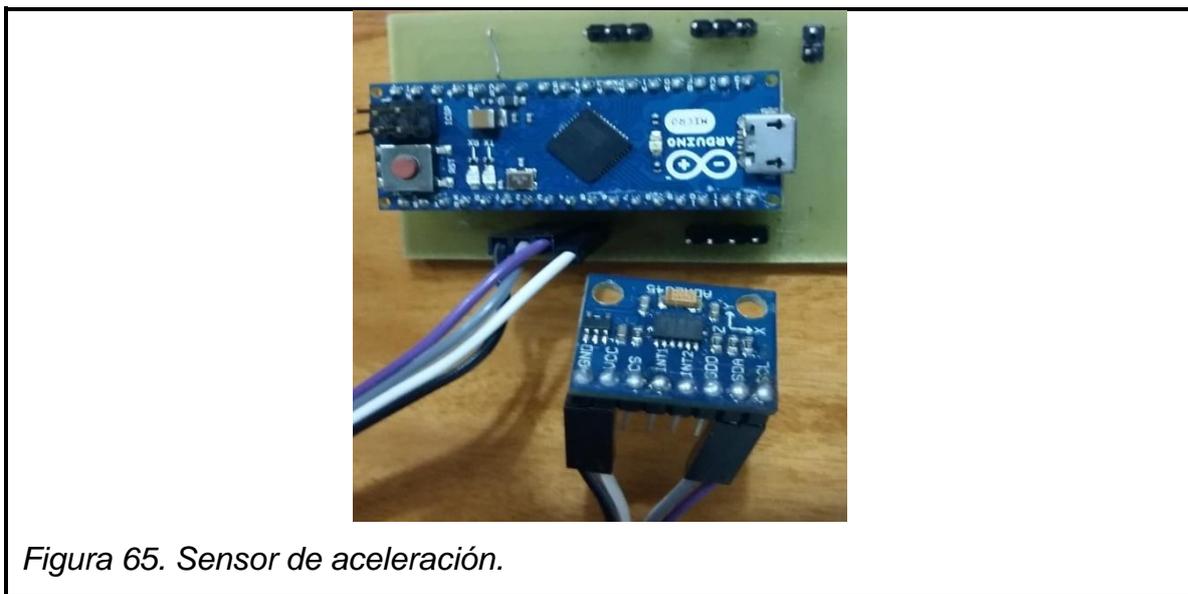


Figura 65. Sensor de aceleración.

Consta de los siguientes pines:

Tabla 9. Conexión del sensor de Aceleración.

PIN SENSOR	PIN ARDUINO
VCC	Salida voltaje 5V
GND	Ground
SDA	PIN 2 (SDA)
SCL	PIN 3 (SCL)

4.1.4 Transmisión de señal Bluetooth

Para lograr la comunicación vía Bluetooth se comprobó realizando pruebas con el protoboard y analizando con la codificación. Bluetooth también va a ir

colocado en el casco, por lo tanto, se necesita poner la placa con el Arduino Micro, como se muestra en la Figura 66. Estableciendo la conexión entre el Arduino y el Bluetooth con los pines especificados en la tabla 10.

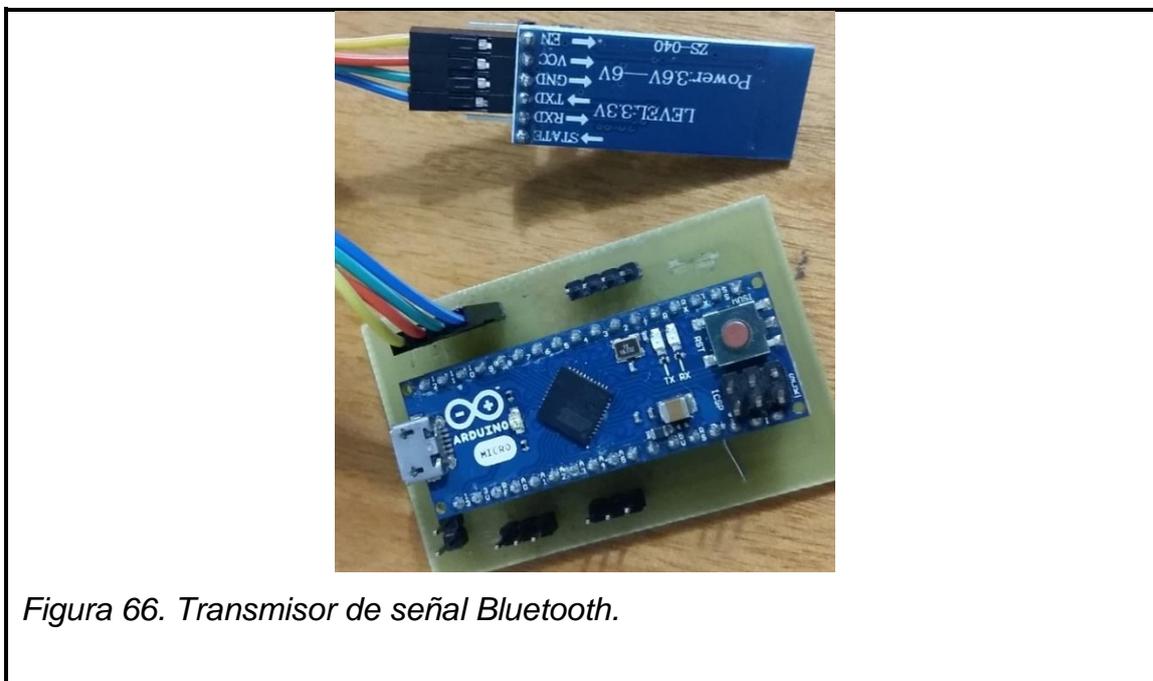


Figura 66. Transmisor de señal Bluetooth.

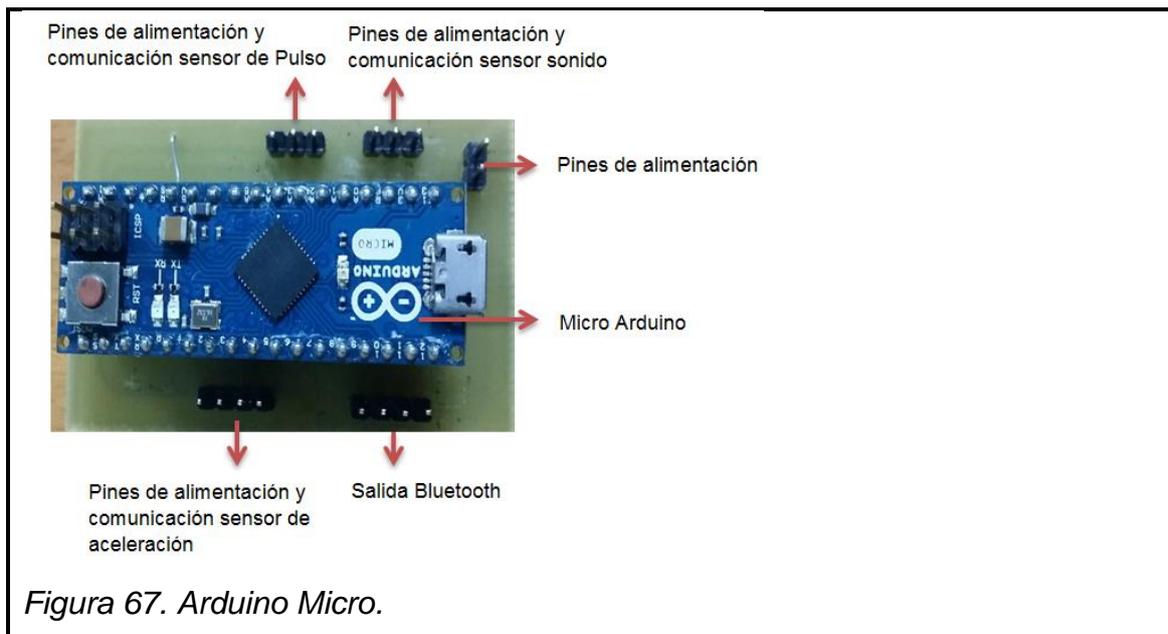
Los pines del bluetooth se detallan a continuación:

Tabla 10. Conexión Bluetooth.

PIN BLUETOOTH	PIN ARDUINO
VCC	Salida voltaje 5V
GND	Ground
TxD	PIN 10
RxD	PIN 11

4.1.5 Arduino micro

El Arduino Micro es el principal dispositivo, en el cual todos los sensores y Bluetooth se conectan con el Arduino Micro, el protoboard sirvió para realizar pruebas y codificación. En la Figura 67 se muestra la placa con el Arduino Micro que será colocada en el casco. Se especifican las salidas de los pines establecidos para los sensores, como se muestra en la tabla 11.



Los pines utilizado en el Arduino Micro son:

Tabla 11. Conexión Arduino Micro.

PIN ARDUINO	PIN SENSOR
VIN	Salida voltaje 5V
GND	Ground
A0	Sensor de Pulso
A1	Sensor de Sonido
2	Sensor de Aceleración
3	Sensor de Aceleración
10	Bluetooth
11	Bluetooth

Circuito Completo

Con el correcto funcionamiento de cada sensor y el bluetooth, se procedió a integrar todos los dispositivos para tener un sistema completo, en la tabla 12 se muestra el consumo de energía del sistema completo, a partir de este cálculo se eligió una Batería de 5V con corriente de 1 A.

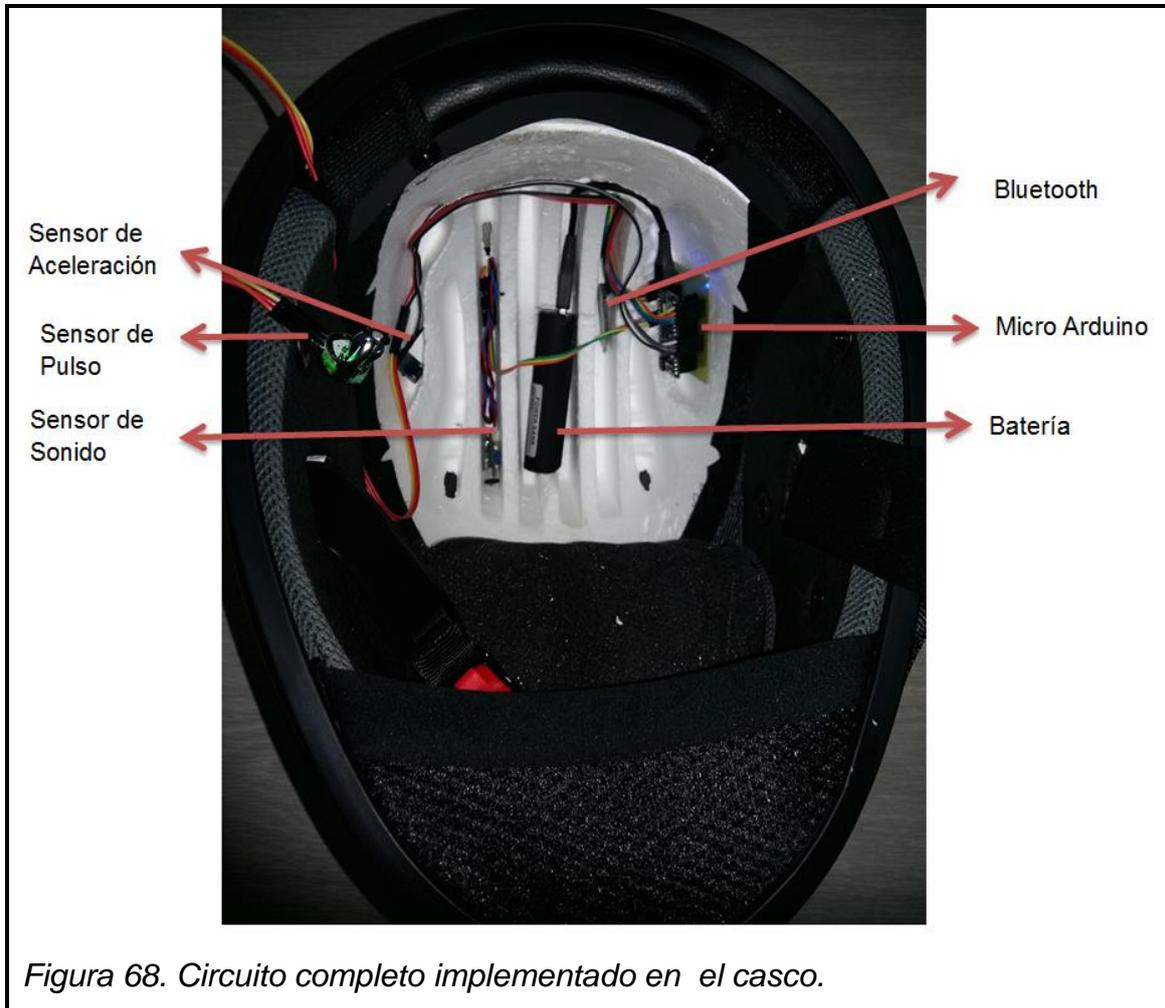
Corriente del circuito

Tabla 12. Corriente del circuito a implementar

SENSOR	CORRIENTE
Pulso	4 mA
Sonido	8 mA
Aceleración	0,023 mA
Micro Arduino	40 mA
Bluetooth	25 mA
Corriente Total	77,023 mA

En el casco, los sensores, controladores y batería se colocaron entre el relleno absorbente y la espuma interna, según lo establecido anteriormente. La batería de 5 voltios, se colocó en la parte superior del casco, ya que requiere mayor espacio. El controlador principal Arduino se colocó en la parte lateral derecha, arriba de la oreja, por facilidad de conexión de los sensores. El sensor de sonido se colocó en la parte superior del casco, para poder tener mayor reflexión de las ondas de sonido que se producirían en cualquier parte del casco en caso de un impacto. El sensor de aceleración se colocó en la parte lateral izquierda a la altura del controlador Arduino. El sensor de pulso se encuentra en el exterior del casco por la parte lateral izquierda, para poder colocarse en el lóbulo de la oreja y obtener correctamente el pulso. Finalmente el módulo de transferencia de datos Bluetooth se encuentra en la parte lateral derecha, cerca del controlador Arduino, para no tener largas distancias entre el controlador Arduino, el módulo de transferencia y el dispositivo móvil del

usuario. En la Figura 68 se muestra como está establecido el circuito dentro del casco.



4.2 Implementación Aplicación Móvil

Según el análisis de requerimientos y el diseño propuesto, se implementará la aplicación móvil en el lenguaje de programación seleccionado, Java junto al kit de desarrollo para Android. El desarrollo del sistema es basado para dispositivos con sistema Android con API 19 o mayor.

Para desarrollar la aplicación propuesta, se utilizará SQLite, que es funcional como base de datos móvil, de fácil manipulación e interacción con el Android.

La pantalla de inicio, la primera en desplegarse al instalar la aplicación, se encontrará la página de inicio para ingresar a la aplicación móvil, la cual permitirá el acceso de usuarios creados previamente, si no se encuentran datos ingresados en la base de datos, permitirá la opción de crear un nuevo usuario, el cual continuará con su proceso mientras se continúe con la aplicación. Contiene la opción de Ingresar a recuperar la contraseña, y finalmente salir de la aplicación. En la Figura 69 se muestra la pantalla de inicio que se desplegará.



Figura 69. Página principal de la aplicación.

Al ingresar por primera ocasión, deberá crear un nuevo usuario, lo que conlleva el llenar formularios los cuales interactivamente se desplegarán en la pantalla del celular, para hacerlo es necesario la selección de la palabra “Nuevo Usuario”, que como lo indica, permite crear un nuevo usuario y agregarlo a la base de datos. En la Figura 70 se muestra la pantalla de formulario usuario nuevo.



Figura 70. Página “Usuario” de la aplicación “Casco Inteligente”.

Al completar con el formulario que se presenta en la pantalla “Usuario”, se procede a guardar datos en la base de datos, comprobando que estos no estén repetidos, a través del número de cédula. Luego de almacenar los datos se visualizará una nueva página de formulario para el contacto en caso de emergencia. La pantalla de contacto se muestra en la Figura 71.

Al completar los datos de la página “Contacto”, se verificará si se encuentran completos los datos y se guardarán en la base de datos, en la respectiva tabla con el usuario que está ingresando. Los datos del contacto permitirán enviar un mensaje desde el celular del usuario al celular del contacto en un caso de emergencia.



Figura 71. Página "Contacto" de la aplicación "Casco Inteligente".

La página "Motocicleta" es la siguiente que se visualiza, desplegará varios campos que se deben llenar por seguridad del usuario que utiliza la aplicación. En la Figura 72 se presenta la página "Motocicleta".



Figura 72. Página "Motocicleta" de la aplicación "Casco Inteligente".

Al completar la información, el usuario cuenta con dos opciones, pasar a la página principal, o tomar una fotografía de la moto, para ser almacenada en la base de datos y tener información extra de esta. Al ingresar a tomar la fotografía se despliega la pantalla en la Figura 73.

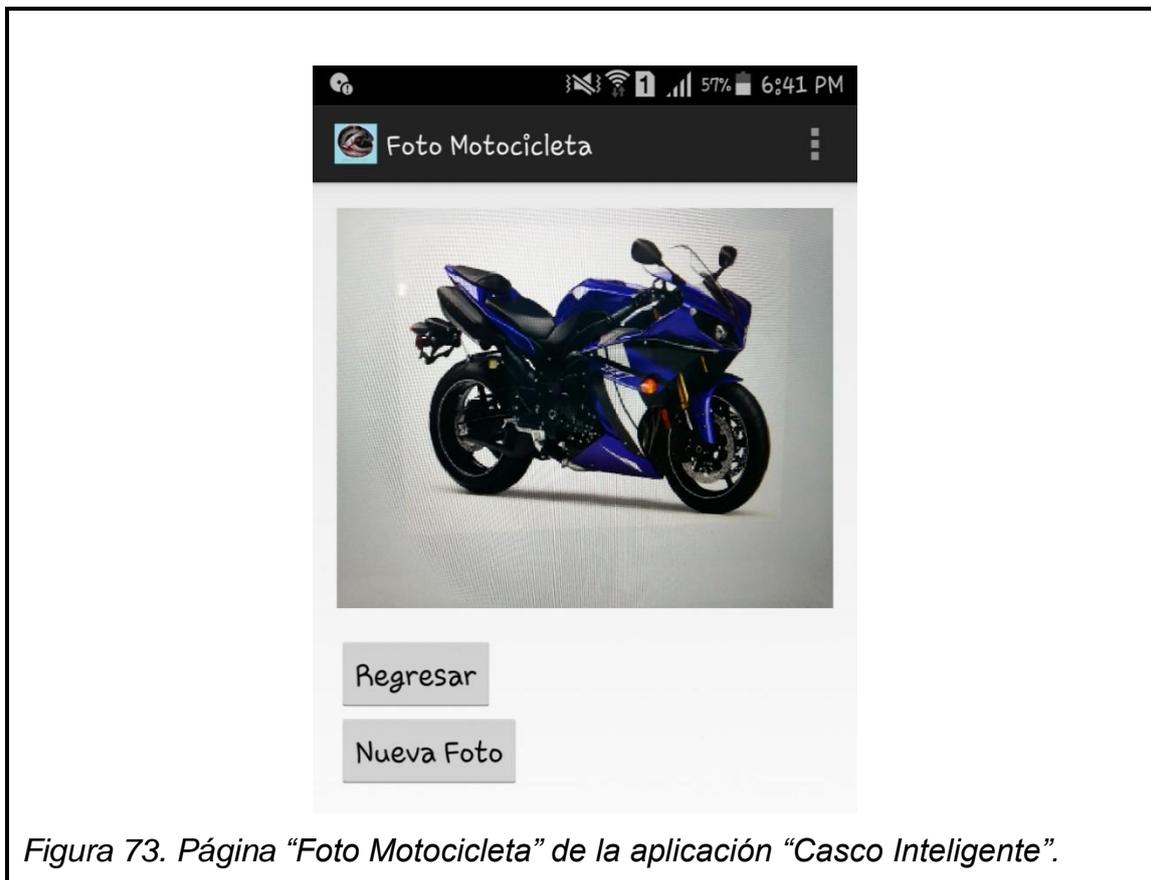


Figura 73. Página “Foto Motocicleta” de la aplicación “Casco Inteligente”.

La opción de nueva fotografía, permitirá desplegar la cámara para poder tomar fotos a la motocicleta que el usuario está guardando, si se encuentra una fotografía previa, se visualizará en la pantalla actual, y aún se tiene la opción de cambiarla. Al tomar la fotografía, se regresará a la página “Foto Moto”, Figura 73, y se despliega la fotografía tomada el instante previo. Al colocar regresar se almacenará la fotografía que se muestra y se vuelve a la página “Moto”.

Finalizando los pasos de creación de usuario, creación de contacto y motocicleta, la página principal se muestra en el celular del usuario, sede

donde es posible conectarse con el casco a utilizar, y establecerá sesión para comenzar la adquisición de datos. La Figura 74 muestra la página "Principal".



Al activar la aplicación se tiene que conectar el casco para poder empezar la transmisión de datos hacia la aplicación, y listo para utilizarlo. Los datos que se recibirán del casco son los que se almacenarán y se verificarán para que en caso de algún accidente se envíe un mensaje predeterminado vía red móvil, con los datos almacenados en la base de datos del usuario, al contacto previamente ingresado según el usuario. En la figura 75, se visualizará la página a desplegarse al crear conexión al casco, y en la figura 76, se muestra la página "Estado" que se desplegará posteriormente a la conexión del casco.

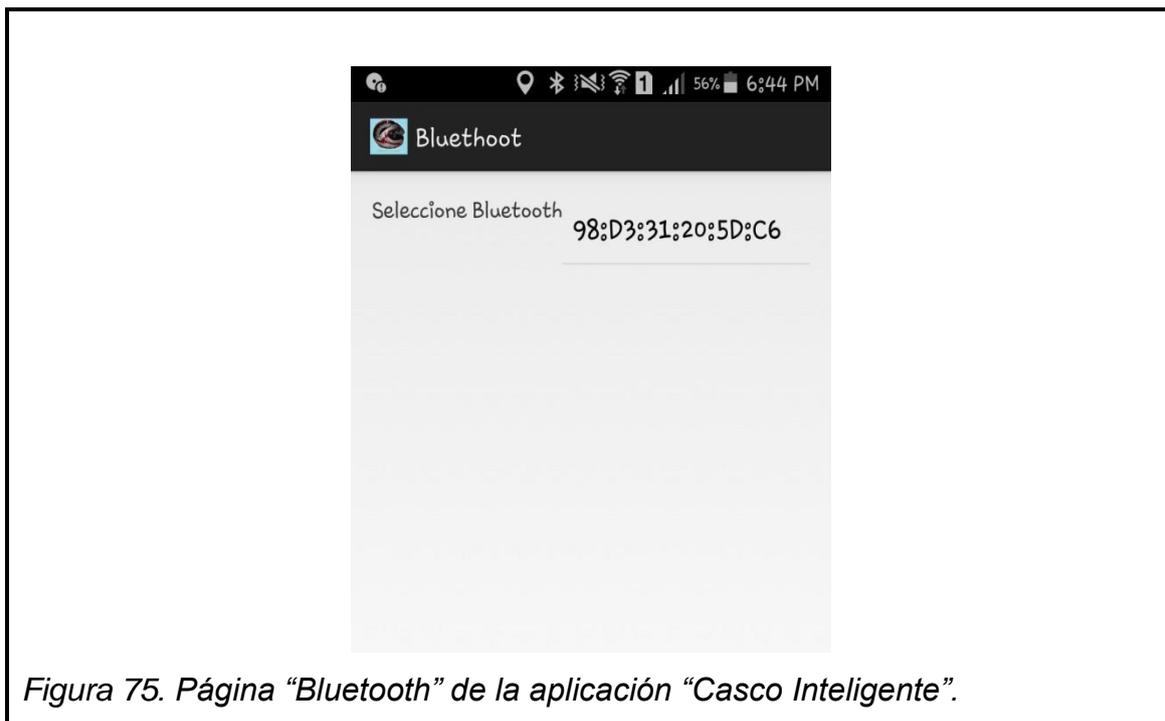


Figura 75. Página “Bluetooth” de la aplicación “Casco Inteligente”.



Figura 76. Página “Estado” de la aplicación “Casco Inteligente”.

La central en la computadora desplegará un mensaje de aviso en caso de que un usuario tenga un evento irregular, dicho mensaje brindará toda la

información de dicho usuario junto a la ubicación del acontecimiento, como se muestra en la Figura 77.

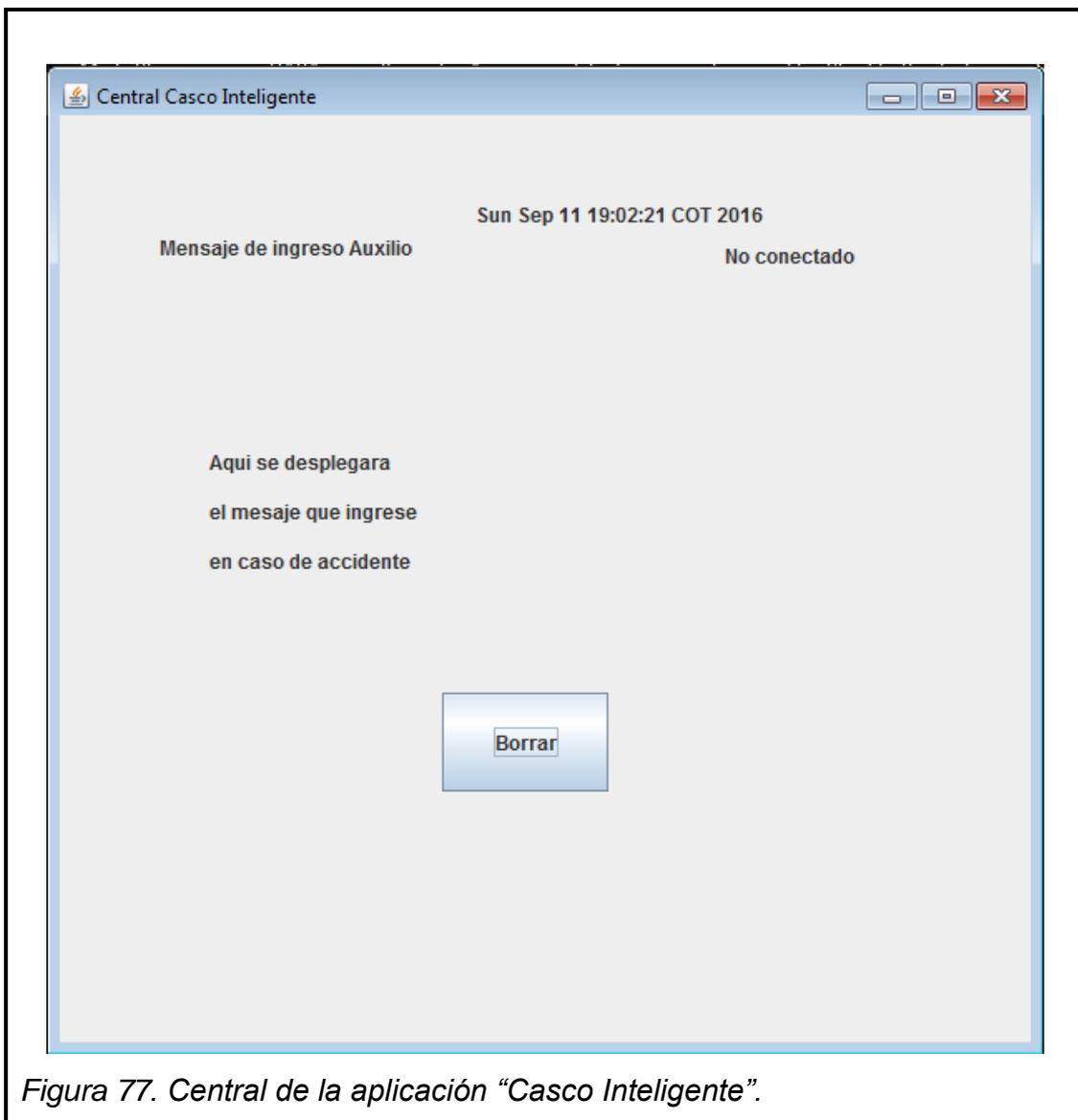


Figura 77. Central de la aplicación "Casco Inteligente".

5. CAPÍTULO V. Pruebas de funcionamiento y Resultado

En el presente trabajo de titulación se realizaron diferentes pruebas de los dispositivos utilizados. Las pruebas respectivas a cada sensor se hicieron para demostrar su buen funcionamiento. Las pruebas realizadas en conjunto demuestran la funcionalidad del prototipo, probando su eficaz funcionamiento.

5.1 PULSO

Para las pruebas del sensor de pulso se utilizó un oxímetro de pulso para dedo, en conjunto con el sensor Sen-11574, tomando 10 muestras por prueba realizada. Cada muestra tomada del sensor de pulso SEN-11574 es ejecutada cada 4050 milisegundos.

Las pruebas se realizaron a un mismo individuo, de 23 años de edad, de sexo femenino, en diferentes lapsos de tiempo y partes del cuerpo en las cuales se pueden medir el pulso.

Primera y segunda prueba

Estas pruebas se realizaron utilizando el sensor de pulso SEN-11574 colocado en el dedo índice de la mano derecha del individuo y el oxímetro de pulso en el dedo índice de la mano izquierda del mismo individuo. Las tablas 13 y 14 demuestran los valores obtenidos en las pruebas realizadas.

Tabla 13. Primera prueba sensor Pulso

NÚMERO DE MUESTRA	VALORES SENSOR DE PULSO SEN-11574 (bpm)	VALORES OXIMETRO DE PULSO (bpm)	CÁLCULO DE ERROR (%)
1	72	73	1%
2	73	73	0%
3	83	72	13%
4	84	73	13%
5	73	74	1%
6	74	75	1%
7	75	76	1%

8	73	77	5%
9	73	78	5%
10	82	80	2%
		Promedio:	4,2%

Tabla 14. Segunda prueba sensor Pulso

NÚMERO DE MUESTRA	VALORES SENSOR DE PULSO SEN-11574 (bpm)	VALORES OXIMETRO DE PULSO (bpm)	CALCULO DE ERROR (%)
1	76	79	3%
2	77	80	3%
3	76	81	4%
4	77	82	5%
5	77	81	4%
6	76	80	4%
7	74	79	5%
8	75	78	3%
9	79	79	0%
10	80	78	2%
		Promedio:	3,3%

Tercera y cuarta prueba

Estas pruebas se realizaron en utilizando el sensor de pulso SEN-11574 colocado en el lóbulo del oído del lado izquierdo del individuo y el oxímetro de pulso en el dedo índice de la mano izquierda del mismo individuo. Las tablas 15 y 16 demuestran los valores obtenidos en las pruebas realizadas.

Tabla 15. Tercera prueba sensor Pulso

NÚMERO DE MUESTRA	VALORES SENSOR DE PULSO SEN-11574 (bpm)	VALORES OXIMETRO DE PULSO (bpm)	CALCULO DE ERROR (%)
1	83	84	1%

2	77	83	6%
3	76	82	6%
4	75	81	6%
5	74	80	6%
6	78	79	1%
7	82	78	4%
8	78	77	1%
9	77	76	1%
10	76	75	1%
		Promedio:	3,3%

Tabla 16. Cuarta prueba sensor Pulso

NÚMERO DE MUESTRA	VALORES SENSOR DE PULSO SEN-11574 (bpm)	VALORES OXÍMETRO DE PULSO (bpm)	CALCULO DE ERROR (%)
1	76	76	0%
2	75	77	2%
3	75	78	3%
4	74	79	5%
5	75	78	3%
6	76	77	1%
7	77	78	1%
8	75	77	2%
9	75	78	3%
10	76	77	1%
		Promedio:	2,1%

Quinta Prueba

Esta prueba se realizó en utilizando el sensor de pulso SEN-11574 colocado en el lóbulo del oído del lado izquierdo del individuo ya colocado el casco utilizado para el prototipo y el oxímetro de pulso en el dedo índice de la mano izquierda del mismo individuo. La tabla 17 se demuestra los valores obtenidos en las pruebas realizadas.

Tabla 17. Quinta prueba sensor Pulso

NÚMERO DE MUESTRA	VALORES SENSOR DE PULSO SEN-11574 (bpm)	VALORES OXIMETRO DE PULSO (bpm)	CALCULO DE ERROR (%)
1	81	77	4%
2	76	76	0%
3	75	76	1%
4	73	75	2%
5	72	74	2%
6	72	73	1%
7	74	72	2%
8	74	73	1%
9	76	74	2%
10	73	73	0%
		Promedio:	1,5%

Los valores obtenidos en las pruebas realizadas al sensor de pulso demuestran un promedio de error de 2,8%. El error puede ser debido a mala posición del sensor. El error obtenido se considera no representativo en este caso.

5.2 SONIDO

Para las pruebas y calibración del sensor de sonido se utilizó una aplicación descargada para celulares inteligentes llamada Sonómetro, la cual fue descargada en dos celulares Samsung. La aplicación utiliza el micrófono de los celulares para determinar el nivel de sonido existente.

Las pruebas se realizaron a un mismo objeto, en diferentes lapsos de tiempo y diferentes ambientes.

Las pruebas realizadas fueron en función del casco y el ambiente externo, y pruebas de golpes en el casco utilizado para determinar diferentes valores de atenuación de la señal. La aplicación se encontraba calibrada con 40 decibeles menos. La calibración del sensor se realizó con las pruebas entre los celulares y el sensor en un mismo ambiente.

Tabla 18. Primera prueba sensor Sonido, calibrado con -40 dB

NÚMERO DE MUESTRA	VALORES SENSOR DE SONIDO (dB)	VALORES APLICACIÓN SONÓMETRO (dB)	CALCULO DE ERROR (%)
1	36	38	5,45%
2	33	33	0%
3	32	31	3,32%
4	30	32	5,8%
5	16	17	5,8%
		Promedio:	4,07%

Tabla 19. Segunda prueba sensor Sonido, calibrado con -40 dB

NÚMERO DE MUESTRA	VALORES SENSOR DE SONIDO (dB)	VALORES APLICACIÓN SONÓMETRO (dB)	CALCULO DE ERROR (%)
1	25	27	7,4%
2	27	30	10%
3	27	27	0%
4	25	26	3,8%
5	30	31	3,2%
		Promedio:	4,88%

Tabla 20. Tercera prueba sensor Sonido, valores externos al casco y sensor en el casco, calibrado con -40 dB

NÚMERO DE MUESTRA	VALORES SENSOR DE SONIDO (dB)	VALORES APLICACIÓN SONÓMETRO (dB)	DIFERENCIA DE SONIDO EXTERIOR Y DEL CASCO (dB)
1	33	36	3
2	33	37	4
3	35	38	3
4	30	34	4
5	25	29	4
		Media de diferencia:	3,6



5.3 ACELERACIÓN

Para las pruebas y calibración del sensor de aceleración se utilizó una aplicación descargada para celulares inteligentes llamada Acelerómetro, la cual fue descargada en un celular Samsung. La aplicación utiliza el acelerómetro del celular para determinar la aceleración en cada plano. Las pruebas se realizaron a un mismo objeto, el casco, en diferentes lapsos de tiempo.

El sensor se calibró en la posición establecida en el casco para anular los valores de ejes cuando el usuario se coloca el casco. Las muestras tomadas en la tabla 21 determinan los valores respecto al sensor y a la aplicación estableciendo los mismos ejes lo más acertadamente posible.

Tabla 21. Prueba sensor Aceleración

NÚMERO DE MUESTRA	VALORES SENSOR DE ACELERACIÓN ADXL345	VALORES APLICACIÓN ACELERÓMETRO	CALCULO DE ERROR (%)
1	x: 2,35 m/s ² y: 1,14 m/s ² z: 20,04–11,62 m/s ² -1g	x: 2,55 m/s ² y: 1,24 m/s ² z: 9,98 m/s ²	x: 7,8% y: 8% z: 1,7%
2	x: 4,71 m/s ² y: -3,06 m/s ² z: 20,04 –11,62 m/s ² -1g	x: 4,8 m/s ² y: -3,15 m/s ² z: 9,98 m/s ²	x: 1,8% y: 2,8% z: 1,7%
3	x: 4,16 m/s ² y: 0,00 m/s ² z: 20,04–11,62 m/s ² -1g	x: 3,99 m/s ² y: 0,50 m/s ² z: 9,98 m/s ²	x: 4,2% y: 1% z: 1,7%
4	x: 1,22 m/s ² y: -5,65 m/s ² z: 20,04–11,62 m/s ² -1g	x: 1,32 m/s ² y: -5,82 m/s ² z: 9,63 m/s ²	x: 7,5% y: 2,9% z: 1,8%
		Media de diferencia:	x: 5,35% y: 3,67% z: 1,72%

Los valores obtenidos en las pruebas realizadas al sensor de aceleración demuestran un promedio de hasta 5,35%. El error puede ser debido a mala posición entre el sensor del celular y el sensor colocado en el casco, por motivos de direccionamiento de ejes y demás. El error obtenido se considera no representativo en este caso. Aplicación que permitió realizar las pruebas para comprobar la funcionalidad del sensor de Aceleración.



Figura 79. Prueba con la Aplicación de Aceleración.

5.4 Prueba conjunta del sistema integrado

Para las pruebas en conjunto del software y hardware se utilizó todos los elementos anteriormente vistos. Se tomaron dos pruebas las cuales demostraron la funcionalidad del prototipo.

Las pruebas se realizaron en un mismo entorno, en diferentes lapsos de tiempo. Durante la ejecución de la aplicación móvil, se generaron nuevos usuarios, contacto, motocicletas e información de motocicleta del usuario creado, llegando hasta la pantalla principal en la que se puede ejecutar la conexión del casco o el regreso a sus diferentes opciones.

Al tener conectado el casco inician las pruebas ejecutadas, teniendo en cuenta que los valores dispuestos para la aplicación y verificación de funcionalidad del sistema son efectivos.

Para realizar las pruebas se utilizó el prototipo creado. En este caso se utilizaron los valores mínimos posibles para ejecutar un supuesto accidente. En la Figura 80 se muestra como el correcto envío del mensaje de alerta hacia la

central del casco inteligente. En la Figura 81 se despliega los datos obtenidos desde la aplicación que fueron almacenados en su base de datos para tener un respaldo posteriormente a un accidente.

En la Figura 82., se muestra que los datos a enviar el mensaje han sido incluidos satisfactoriamente.

Al finalizar las pruebas con éxito, se comprobó la funcionalidad principal del sistema, el cual será la alerta temprana por mensaje de texto o a una central. Además de la recepción de los datos almacenados en la base de datos móvil al ocurrir un accidente.

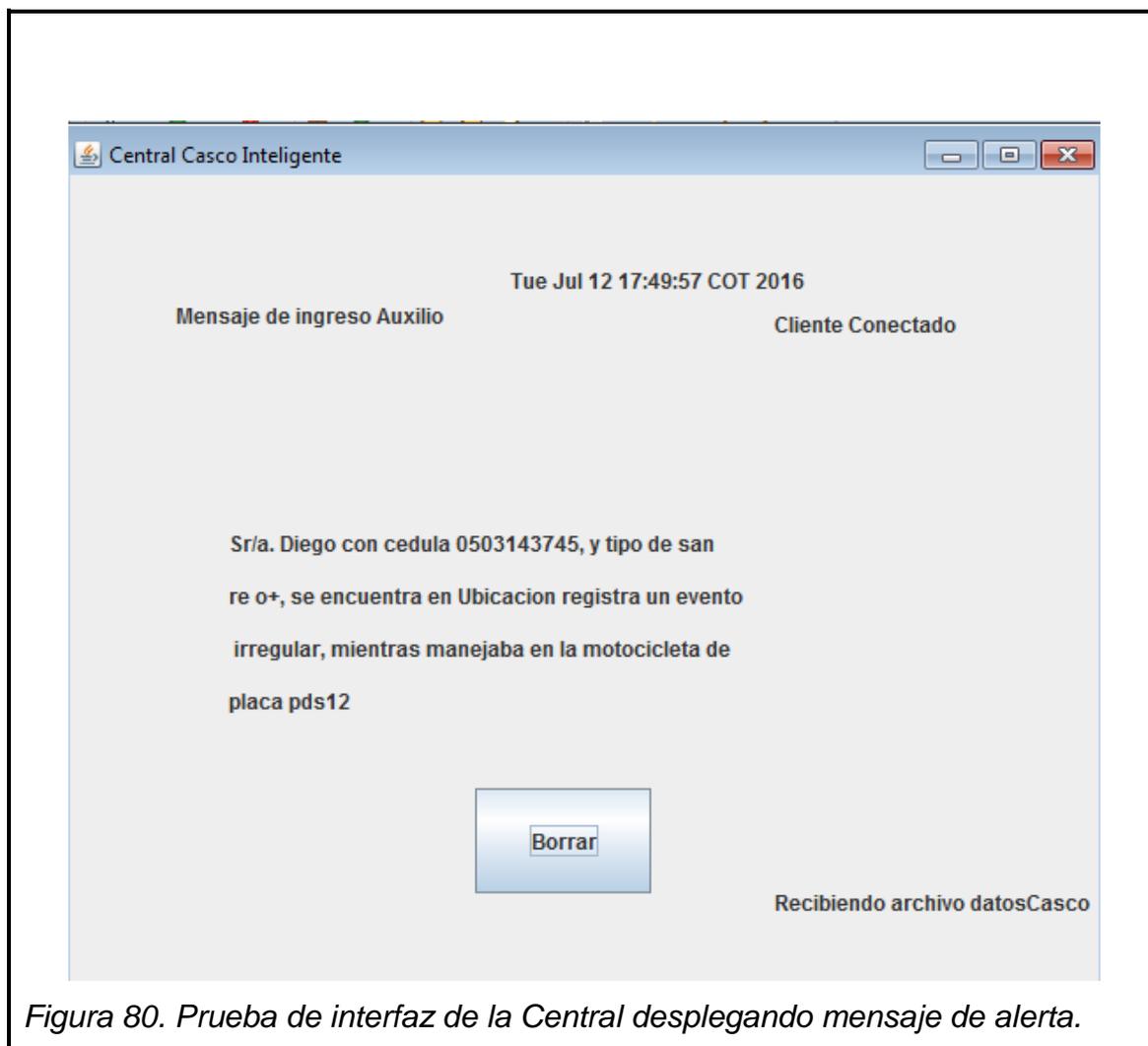


Figura 80. Prueba de interfaz de la Central desplegando mensaje de alerta.

Evento=1	Pulso=196	Aceleracion= X:0 Y:0 Z:0	Impacto=No	Velocidad=0.28
Evento=2	Pulso=196	Aceleracion= X:0 Y:0 Z:0	Impacto=No	Velocidad=0.28
Evento=3	Pulso=143	Aceleracion= X:3 Y:2 Z:0	Impacto=No	Velocidad=0.28
Evento=4	Pulso=143	Aceleracion= X:0 Y:0 Z:0	Impacto=No	Velocidad=0.28
Evento=5	Pulso=142	Aceleracion= X:0 Y:1 Z:0	Impacto=No	Velocidad=0.28
Evento=6	Pulso=142	Aceleracion= X:2 Y:2 Z:2	Impacto=No	Velocidad=0.28
Evento=7	Pulso=77	Aceleracion= X:0 Y:0 Z:0	Impacto=No	Velocidad=0.28
Evento=8	Pulso=77	Aceleracion= X:0 Y:0 Z:0	Impacto=No	Velocidad=0.28
Evento=9	Pulso=55	Aceleracion= X:1 Y:3 Z:3	Impacto=No	Velocidad=0.28
Evento=10	Pulso=55	Aceleracion= X:0 Y:0 Z:0	Impacto=No	Velocidad=0.28
Evento=11	Pulso=43	Aceleracion= X:0 Y:0 Z:0	Impacto=No	Velocidad=0.28
Evento=12	Pulso=43	Aceleracion= X:0 Y:0 Z:0	Impacto=No	Velocidad=0.28
Evento=13	Pulso=37	Aceleracion= X:0 Y:2 Z:1	Impacto=No	Velocidad=0.28
Evento=14	Pulso=37	Aceleracion= X:0 Y:0 Z:0	Impacto=No	Velocidad=0.28
Evento=15	Pulso=38	Aceleracion= X:0 Y:0 Z:0	Impacto=No	Velocidad=0.96
Evento=16	Pulso=38	Aceleracion= X:0 Y:3 Z:4	Impacto=No	Velocidad=0.96
Evento=17	Pulso=58	Aceleracion= X:0 Y:0 Z:0	Impacto=No	Velocidad=0.96
Evento=18	Pulso=58	Aceleracion= X:0 Y:0 Z:0	Impacto=No	Velocidad=0.96
Evento=19	Pulso=52	Aceleracion= X:1 Y:3 Z:0	Impacto=No	Velocidad=0.96
Evento=20	Pulso=52	Aceleracion= X:0 Y:0 Z:0	Impacto=No	Velocidad=0.96
Evento=21	Pulso=108	Aceleracion= X:2 Y:12 Z:5	Impacto=Si	Velocidad=0.96

Figura 81. Prueba de Datos Almacenados.



Figura 82. Prueba mensaje de alerta.

6. Conclusiones y Recomendaciones

6.1 Conclusiones

En el desarrollo del prototipo del sistema de monitoreo, detección de accidentes y alerta inmediata para motociclistas, se logró cumplir con todos los objetivos que incluyen el diseño, la implementación y correcto funcionamiento del sistema.

Es importante que el casco a utilizar para el prototipo esté homologado, pues según las leyes tránsito todo casco utilizado en el territorio nacional debe cumplir con ciertas características definidas por la ANT.

El montaje de los sensores en el casco tiene que ser meticulosamente definido pues se debe considerar diferentes factores tales como la comodidad del usuario, que no se afecte el diseño del casco, que la orientación de los mismos sea tal que se satisfaga las necesidades propias de la medición y se evite en medida de lo posible todo tipo de interferencias.

Para esta aplicación en particular, el uso de tecnología Bluetooth como medio de comunicación inalámbrica es altamente conveniente porque es compatible con la mayoría de dispositivos móviles y además el rango de cobertura de hasta 10 metros según el estándar IEEE 802.15 es más que suficiente para cubrir la distancia usual entre el casco del motociclista y su teléfono celular.

La implementación de la aplicación para sistema móvil en Android hace posible el llegar a una gran cantidad de usuarios ya que la mayoría de teléfonos inteligentes usado en país cuentan con este sistema operativo, además es sencilla y de fácil uso para cualquier persona.

Para el establecimiento de alertas fue necesario, establecer umbrales en cada sensor, para ello fue importante tener información real de los valores de las

diferentes variables en un accidente; si el valor de la variable sobrepasa dicho umbral es alto se puede considerar que es un accidente y así poder emitir una señal de alerta.

Para la realización de pruebas fue necesario el bajar los valores de los umbrales de impacto y aceleración, pues resulta imposible reproducir las condiciones reales de un accidente motociclístico. Las pruebas realizadas se configuraron con umbrales muy bajos, que hagan posible la emisión de mensajes de alerta.

Durante la realización del prototipo, se encontraron varios fallos al enviar datos entre el Arduino y la aplicación, debido a que el buffer del Bluetooth envía 8 bits de datos. Se logró controlar esto, enviando más pequeña la cadena de caracteres a la aplicación; adicionalmente en la aplicación se implementó condiciones de decisión, las cuales revisan diferentes banderas con la finalidad de identificar el inicio y el final de cada cadena de datos.

Para lograr establecer la lectura de datos en la programación del Arduino, se necesita establecer el tiempo de ejecución, y así obtener datos reales de cada sensor.

En la elaboración del prototipo se implementó una aplicación que receipta datos desde la aplicación móvil, la cual al estar en un dispositivo de escritorio debería tener una dirección IP pública para la comunicación a través de internet, pero para reducir costos y comprobar su funcionalidad se utilizó una red LAN.

Para concluir satisfactoriamente el proyecto de tesis planteado se utilizó los conocimientos obtenidos durante la carrera, además de investigaciones y recopilaciones de información externa.

6.2 Recomendaciones

Para ahorrar el uso de la batería se recomienda colocar un Switch para evitar que se descargue rápido, una opción que ayudaría al casco es contar con un sistema de alimentación mediante recarga solar que este ubicado en el casco con el objetivo de suministro de energía autónomo.

Para poder calibrar los sensores en la desarrollo de la codificación es necesario que los sensores se encuentren colocados en el casco para poder encerrarles. Este proceso solo se lo realiza la primera vez cuando se está haciendo la codificación del Arduino.

Para darle un mejor diseño y acabado al prototipo se recomienda hablar con los fabricantes y llegar a un acuerdo para que en el modelo de casco se pueda adecuar los elementos sin tener que modificar, dañar, obstruir la integridad física del casco homologado.

Se recomienda al implementador tener en cuenta el tiempo de ejecución de los sensores y la cantidad de datos que se va a transmitir entre hardware y software para no tener dificultades en lectura de datos e inconvenientes en pérdida de información.

Referencias

- AG Electrónica S.A. de C.V. (s.f.). *KIT de Sensores y Actuadores Compatibles con Arduino*. Recuperado el 28 de Mayo de 2016 de <http://www.agspecinfo.com/pdfs/K/KITSENAT.PDF>
- Agencia Nacional de Tránsito. (s.f.). *Estadísticas de transporte terrestre seguridad vial*. Recuperado el 10 de Marzo de 2016 de <http://www.ant.gob.ec/index.php/noticias/estadisticas#.V4K4hfnhDX7>
- Aguirre, L., & Sinche, H. (2013). *Diseño de una aplicación móvil para la consulta académica de la FIIS-UTP*. Universidad Tecnológica del Perú, Perú.
- Aprender JAVA. (s.f.). *Curso Java Web Developer*. Recuperado el 08 de Diciembre de 2016 de <http://aprender-java.blogspot.com/>
- Astudillo, G. (s.f.). *Programación en C para Arduino*. Universidad de Valparaíso. Recuperado el 20 de Agosto de 2016 de http://informatica.uv.cl/~gabriel/docs/arduino/arduino_guia.pdf
- Biblioteca de Ingeniería. (s.f.). *Microcontroladores*. Recuperado el 27 de Agosto de 2016 de <http://bibing.us.es/proyectos/abreproy/11141/fichero/PFC%252F3+Microcontroladores.pdf>
- Calderón, J., & Caluguillin, L. (2011). *Construcción de un banco didáctico de un motor de inyección electrónica multipunto, para la escuela de ingeniería automotriz de la ESPOCH*. Escuela Superior Politécnica de Chimborazo, Riobamba - Ecuador.
- CCM. (s.f.). *Sistema operativo*. Recuperado el 27 de Enero de 2016 de <http://es.ccm.net/contents/651-sistema-operativo>
- Chemes, C. (2008). *La enfermera y la valoración de los signos vitales*. Argentina – Tucumán.
- Delandes, M. (2011). *Historia de las bases de datos*. Recuperado el 26 de Agosto de 2016 de <http://culturainformatica.es/articulos/historia-de-las-bases-de-datos/comment-page-1/>.

- Díaz, C., & Hommy, L. (2014). *Desarrollo de un prototipo de casco inteligente para la prevención de lesiones y seguridad en vehículos motorizados de dos ruedas*. Universidad Nueva Esparta, Caracas, Venezuela.
- Díaz, J. (s.f.). *Protección a prueba*. Recuperado el 09 de Junio de 2016 de http://ftp.inti.gob.ar/comunicacion/FLOR2/CASCOS/nota%20cascos_2parte.pdf (2^{da} ed.).
- RAE. (s.f.) *Diccionario de español de la Real Academia de la Lengua*. Recuperado el 27 de Enero de 2016 de <http://dle.rae.es/?w=diccionario>
- Fernández, J. (2012). *Ejemplo de aplicación con Arduino*. Universitat Rovira I Virgili.
- IEEE Xplore Digital Library. (s.f.). *Bluetooth, carrying voice over ACL links*. Recuperado el 27 de Enero de 2016 de http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=1045792&url=http%3A%2F%2Fieeexplore.ieee.org%2Fexpls%2Fabs_all.jsp%3Farnumber%3D1045792
- Jaén, J. (2013, Diciembre). *Soft Computing para el análisis de datos en Redes de Sensores Inalámbricos*. Recuperado a partir de file:///D:/Downloads/TFM_Jaen_Mari%C3%B1as.pdf
- Játiva, A. (2010). *Sistema de puertas con sensores para la apertura en paradas autorizadas en buses del servicio urbano del Distrito Metropolitano de Quito*. Universidad Internacional del Ecuador, Quito-Ecuador.
- Martínez, J. (s.f.). *Fundamentos de programación en Java*. EME.
- Microcontroladores PIC. (s.f.). Recuperado el 27 de Agosto de 2016 de <http://www.ceduvirt.com/resources/Microcontroladores.pdf>
- Morillo D. (s.f.). *Entornos de programación móviles*. Recuperado el 08 de Diciembre de 2016 de [https://www.exabyteinformatica.com/uoc/Informatica/Tecnologia_y_desarrollo_en_dispositivos_moviles/Tecnologia_y_desarrollo_en_dispositivos_moviles_\(Modulo_3\).pdf](https://www.exabyteinformatica.com/uoc/Informatica/Tecnologia_y_desarrollo_en_dispositivos_moviles/Tecnologia_y_desarrollo_en_dispositivos_moviles_(Modulo_3).pdf)
- Motorpasion. (s.f.). *Homologación que debe cumplir un casco de motocicleta para ser seguro y legal*. Recuperado el 09 de Junio de 2016 de

<http://www.motorpasionmoto.com/seguridad/cual-es-la-homologacion-que-debe-cumplir-un-casco-de-motocicleta-para-ser-seguro-y-legal>

- Pedrozo, G. (2012). *Sistemas Operativos en Dispositivos Móviles*. Universidad Nacional del Nordeste.
- Perry, J. (2012). Introducción a la programación Java, parte 1. Conceptos básicos del lenguaje Java. IBM. Developer Works. Recuperado el 08 de Agosto de 2016 de <http://www.ibm.com/developerworks/ssa/java/tutorials/j-introtojava1/j-introtojava1-pdf.pdf>
- Programar en JAVA. (s.f.). *Manual Básico de JAVA*. Recuperado el 08 de Diciembre de 2016 de <https://docs.google.com/a/udlanet.ec/file/d/0Byy7aUI9u4fBZ05peFd5MmpTbjg/edit>
- Ronquillo, J. & Salgado, P. (2013). *Diseño y Construcción de un oxímetro de pulso portátil*. Universidad del Azuay. Recuperado el 29 de Agosto de 2016 de <http://dspace.uazuay.edu.ec/bitstream/datos/3291/1/10065.pdf>
- Sierra, M. (2012). Qué es una base de datos y cuáles son los principales tipos. *Aprender a programar*. Recuperado el 30 de Agosto de 2016 de
- Tecnologías de la Información y Comunicación. (s.f.). *Sistemas Operativos*. Recuperado el 08 de abril de 2016 de <http://proyectoticsmn.com/VirtSO/contenidos/fundamentosso.pdf>
- Universidad de Cádiz. (2012). *Comenzando con Arduino*. Cádiz, España: Universidad de Cádiz.
- Vahora, A., Patel, R., Goradiya, B., & Desai, A. (2014). *Heart Beat Monitoring and Wireless Data Logging Using Arm Cortex A8*. India: BVM Engineering College.
- Valle, C., & Vallejo, J. (2014). *Construcción e implementación de un simulador de sensores y actuadores del motor, ABS, aire acondicionado e inmovilizadores, para reparar computadoras automotrices para la*

escuela de ingeniería automotriz. Escuela Superior Politécnica de Chimborazo, Riobamba - Ecuador.

Ventura, V. (2015). *Principios de funcionamiento del oxímetro para monitorización del pulso. Recuperado el 27 de Mayo de 2016 de <http://polaridad.es/monitorizacion-sensor-pulso-oximetro-frecuencia-cardiaca/>*

Villalón, J., & López, A. (s.f.). *El corazón del deportista. Recuperado el 26 de Abril de 2016 de http://www.fbbva.es/TLFU/microsites/salud_cardio/mult/fbbva_libroCorazon_cap68.pdf*

World Famous Electronic Ilc. (s.f.). *Pulse Sensor Amped. Recuperado el 20 de Mayo de 2016 de <http://pulsesensor.com/pages/pulse-sensor-amped-arduino-v1dot1>*

Zambrano, J. (2013, 14 de agosto). *Ambulancias aún no cumplen el tiempo óptimo de respuesta. Diario La Hora. Recuperado el 01 de Abril de 2016 de http://lahora.com.ec/index.php/noticias/show/1101548674/-1/Ambulancias_a%C3%BA_n_no_cumplen_el_tiempo_%C3%B3ptimo_de_respuesta.html#.V9I8fx7hDIU*

ANEXOS

ANEXO 1. DATASHEET DE SENSORES

Esquema de la placa para Arduino Micro.

