ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA BIOMÉTRICO DE TRANSMISIÓN INALÁMBRICA PARA EL RECONOCIMIENTO DE PERSONAS A TRAVÉS DE SU HISTORIAL MÉDICO EN EL HOSPITAL REGIONAL DOCENTE AMBATO

BYRON GENARO SALAZAR MOPOSITA

Trabajo de Titulación modalidad: Proyectos de Investigación y Desarrollo, presentado ante el Instituto de Posgrado y Educación Continua de la ESPOCH, como requisito parcial para la obtención del grado de:

MAGÍSTER EN SISTEMAS DE TELECOMUNICACIONES

Riobamba - Ecuador

ENERO 2022

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

CERTIFICACIÓN:

EL TRIBUNAL DE TRABAJO DE TITULACIÓN CERTIFICA QUE:

El **Trabajo de Titulación modalidad Proyectos de Investigación y Desarrollo**, titulado **DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA BIOMÉTRICO DE TRANSMISIÓN INALÁMBRICA PARA EL RECONOCIMIENTO DE PERSONAS A TRAVÉS DE SU HISTORIAL MÉDICO EN EL HOSPITAL REGIONAL DOCENTE AMBATO**, de responsabilidad del señor Byron Genaro Salazar Moposita ha sido prolijamente revisado y se autoriza su presentación.

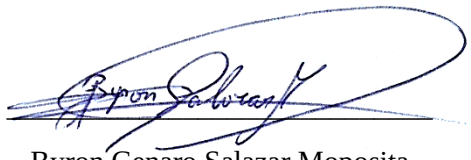
Tribunal:

Ing. Luis Eduardo Hidalgo Almeida; Ph.D. **PRESIDENTE**	LUIS EDUARDO HIDALGO ALMEIDA Firmado digitalmente por LUIS EDUARDO HIDALGO ALMEIDA Nombre de reconocimiento (DN): c=EC, o=BANCO CENTRAL DEL ECUADOR, ou=ENTIDAD DE CERTIFICACION DE INFORMACION-ECIBCE, l=QUITO, serialNumber=0000445780, cn=LUIS EDUARDO HIDALGO ALMEIDA Fecha: 2022.01.21 10:04:54 -05'00' Firma
Ing. Geovanni Danilo Brito Moncayo; Mag. **DIRECTOR**	Firmado electrónicamente por: **GEOVANNI DANILO BRITO MONCAYO** Firma
Ing. Santiago Mauricio Altamirano Meléndez; Mag. **MIEMBRO DEL TRIBUNAL**	Firmado electrónicamente por: **SANTIAGO MAURICIO ALTAMIRANO MELENDEZ** Firma
Ing. Elizabeth Paulina Ayala Baño; Mag. **MIEMBRO DEL TRIBUNAL**	ELIZABETH PAULINA AYALA BANO Firmado digitalmente porELIZABETH PAULINA AYALA BANO DN: c=EC,l=ELIZABETH PAULINA AYALA BANO,ou=SECURITY DATA S.A. 1,cn=ENTIDAD DE CERTIFICACION DE INFORMACION Motivo:He revisado este documento Ubicación: Fecha:2022-01-18 18:03-05:00 Firma

Riobamba, enero 2022

DERECHOS INTELECTUALES

Yo, Byron Genaro Salazar Moposita, declaro que soy responsable de las ideas, doctrinas y resultados expuestos en el Trabajo de Titulación modalidad Proyectos de Investigación y Desarrollo, y que el patrimonio intelectual generado por la misma pertenece exclusivamente a la Escuela Superior Politécnica de Chimborazo.



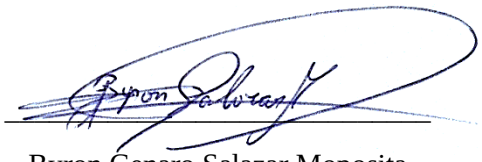
Byron Genaro Salazar Moposita

No. 1803725066

DECLARACIÓN DE AUTENTICIDAD

Yo, Byron Genaro Salazar Moposita, declaro que el presente trabajo de titulación, modalidad Proyectos de Investigación y Desarrollo, es de mi autoría y que los resultados del mismo son auténticos y originales. Los textos constantes en el documento que provienen de otra fuente están debidamente citados y referenciados.

Como autor, asumo la responsabilidad legal y académica de los contenidos de este proyecto de investigación de maestría.



Byron Genaro Salazar Moposita

No. 1803725066

DEDICATORIA

Este trabajo lo dedico con todo mi corazón principalmente a Dios y a la Virgen de Agua Santa, que me han bendecido con salud, vida, una familia maravillosa y me han permitido realizarme como profesional. A mi madre Beatriz Moposita que la admiro y adoro, que siempre ha confiado en mí y me ha apoyado desde la distancia. A mi sobrino Conor Salazar que con una simple sonrisa me llena de alegría. A mi hermano Christian Salazar que se ha convertido en un pilar fundamental en mi vida, para seguir creciendo como persona y profesional. A mi enamorada Cumandá Arroba, que ha llegado a ser una persona valiosa en mi vida, ha estado en las buenas y malas situaciones impulsándome para cumplir este sueño. Finalmente, a toda mi familia que se han convertido en un ejemplo a seguir que por buscar sus sueños han migrado hacia otros países, convirtiéndose en mi inspiración para luchar cada día. Y a todas las personas que confiaron en mí y que de alguna manera me brindaron su apoyo.

AGRADECIMIENTO

Agradezco a Dios por permitirme culminar el presente trabajo de investigación, por darme la fuerza necesaria para levantarme cada día y cumplir este sueño. A mi madre Beatriz Moposita que es mi inspiración, que con su ejemplo me ha enseñado a luchar constantemente. A mi sobrino Conor Salazar que con tan solo una sonrisa me llena de alegría. A Christian Salazar que a más de ser un hermano es un gran amigo que siempre ha estado a mi lado en los buenos y malos momentos. A mi enamorada Cumandá Arroba que se ha convertido en una persona importante en mi vida y me ha alentado apara cumplir mis metas. A mi Tío Franklin Roberto que siempre me ha brindado su apoyo. A Geovanni Brito, Paulina Ayala y Santiago Altamirano que con su experiencia me guiaron para desarrollar y culminar este proyecto.

TABLA DE CONTENIDO

RESUMEN xvii
ABSTRACT xviii
CAPÍTULO I 1
INTRODUCCIÓN 1
1. PLANTEAMIENTO DEL PROBLEMA 1
1.1. Situación Problemática 1
1.2. Formulación del Problema 3
1.3. Sistematización del Problema 3
1.4. Justificación de la Investigación 3
1.5. Objetivos de la Investigación 4
1.5.1. Objetivo General 4
1.5.2. Objetivos Específicos. 4
1.6. Hipótesis 4
1.6.1. Hipótesis General 4
1.6.2. Hipótesis Específicas 4
1.6.3. Identificación de variables 4
CAPÍTULO II 5
2. MARCO DE REFERENCIA 5
2.1. Antecedentes del Problema 5
2.2. Bases Teóricas 6
2.2.1. Biometría. 6
2.2.2. Comunicación inalámbrica 8
2.3. Marco Conceptual 8
2.3.1. Huella Dactilar 8
2.3.2. Lector biométrico para la adquisición de Imágenes. 12
2.3.3. Arduino Mega 2560 13
2.3.4. Pantalla Nextion TFT-LCD NX4827T043 15
2.3.5. Módulo SIM800L 16
2.3.6. Módulo de identificación de abonado (SIM) 17
2.3.7. Servidores 18
2.3.8. Seguridad de base de datos 19
2.3.9. LabVIEW 20
2.3.10. Inkscape 21

2.3.11. SolidWorks 21
CAPÍTULO III 22
3. DISEÑO DE LA INVESTIGACIÓN 22
3.1. Tipos de Investigación 22
3.2. Métodos de Investigación 22
3.3. Enfoque de la Investigación 22
3.4. Alcance de la Investigación 22
3.4.1. Descriptivo 22
3.4.2. Correlacional 22
3.4.3. Explicativo: 22
3.5. Unidad de Análisis 23
3.5.1. Identificación de personas 23
3.5.2. Comunicación inalámbrica 23
3.5.3. Procesamiento Digital de Imágenes 23
3.5.4. Bases de datos 23
3.6. Diseño e Implementación 23
3.6.1. Comparación de las Tecnologías Biométricas 23
3.6.2. Comparación de las Tarjetas de desarrollo. 24
3.6.3. Análisis de los Módulos Inalámbricos 25
3.6.4. Seguridad de base de datos 30
3.6.5. Requerimientos alcanzados 33
3.6.6. Equipo terminal 33
3.6.7. Estación central 55
CAPÍTULO IV 84
4. ANÁLISIS Y RESULTADOS 84
4.1. Medidas de Errores en los Sistemas de Verificación 84
4.1.1. Tasa de Falsa Aceptación FAR 84
4.1.2. Tasa de Falso Rechazo FRR 85
4.1.3. Tasa de Resultado (SR) 86
4.1.4. Tasa de Error (EER) 86
4.2. Resultados obtenidos de la encuesta 87
4.3. Comprobación de la Hipótesis 89
4.3.1. Nivel de Significación 89
4.3.2. Elección de la prueba estadística 89
4.3.3. Zona de aceptación y rechazo (Grados de libertad) 89

4.3.4. Frecuencias Observadas 90
4.3.5. Frecuencias Esperadas 90
4.3.6. Chi Cuadrado 90
CAPÍTULO V 92
5. PROPUESTA 92
5.1. Hipótesis 92
5.2. Sistema Biométrico 92
5.3. Arquitectura 93
5.4. Objetivos 93
5.4.1. Objetivo General 93
5.4.2. Objetivos Específicos 93
5.5. Métodos de Investigación 94
PRESUPUESTO 95
CONCLUSIONES 96
RECOMENDACIONES 97
BIBLIOGRAFÍA
ANEXOS

ÍNDICE DE TABLAS

Tabla 1-2: Características técnicas del sensor biométrico. 12
Tabla 2-2: Características Técnicas del Arduino ATmega2560 13
Tabla 3-2: Características Técnicas de la Pantalla Nextion NX4827T043. 15
Tabla 4-2: Características técnicas del módulo SIM800L. 17
Tabla 1-3: Comparación entre diversas tecnologías biométricas 24
Tabla 2-3: Tabla comparativa entre Arduino Y Raspberry Pi 25
Tabla 3-3: Tabla comparativa entre Módulos Sim 30
Tabla 4-3: Identificadores de los controles de la página principal. 38
Tabla 5-3: Código de los botones de la página principal. 38
Tabla 6-3: Identificadores de los controles de la segunda página. 39
Tabla 7-3: Código de los controles de la segunda página. 40
Tabla 8-3: Identificadores de los controles de las páginas de visualización. 41
Tabla 9-3: Código de los controles de la tercera y cuarta página 41
Tabla 10-3: Trama para escribir los datos consultados en los controles txt. 44
Tabla 11-3: Objetos instanciados para controlar desde el Arduino. 44
Tabla 12-3: Trama de datos del botón "ACEPTAR" con los números de la cédula. 45
Tabla 13-3: Trama de datos de los botones "Datos" y "Huella Digital" 45
Tabla 14-3: Trama de confirmación de conexión entre el Sensor y Arduino 46
Tabla 15-3: Acuse de recibo de la confirmación de conexión. 46
Tabla 16-3: Trama de datos para capturar la imagen de la huella digital. 47
Tabla 17-3: Acuse de recibo de la captura de la imagen 47
Tabla 18-3: Trama para generar archivo de caracteres a partir de la imagen. 48
Tabla 19-3: Acuse de recibo de la generación de archivo de caracteres de la Huella. 48
Tabla 20-3: Formato de comandos para descargar la plantilla del Buffer 1. 49
Tabla 21-3: Acuse de recibo de la Descarga de la Plantilla de minucias. 49
Tabla 22-3: trama de datos y trama de fin de paquete de datos. 49
Tabla 23-3: Formato de la trama de datos de los rasgos característicos. 50
Tabla 24-3: Trama de datos para procesamiento de Imágenes. 50
Tabla 25-3: Envió de mensajes SMS mediante comandos AT 51
Tabla 26-3: Potencia de la señal de recepción inalámbrica 53
Tabla 27-3: Conversor análogo Digital para monitorear la carga de la batería. 54
Tabla 28-3: Usuarios Registrados 58
Tabla 29-3: Datos Personales 59

Tabla 30-3: Datos Médicos. 59
Tabla 31-3: Imágenes y Plantillas de Minucias. 59
Tabla 32-3: Comprimiendo dos bytes en uno solo 63
Tabla 33-3: Sentencias SQL para Registro de Datos de Usuarios. 64
Tabla 34-3: Descompresión de Datos para la Construcción de la imagen. 66
Tabla 35-3: Tipo de minucias obtenidas con el algoritmo. 74
Tabla 36-3: Trama de Datos de entrada a la Estacón Base 77
Tabla 37-3: Estructura de las minucias (Tipo, calidad, ángulo, posición). 79
Tabla 1-4: Análisis e interpretación, pregunta 1. 87
Tabla 2-4: Análisis e interpretación, pregunta 2 87
Tabla 3-4: Análisis e interpretación, pregunta 3 88
Tabla 4-4: Análisis e interpretación, pregunta 4 88
Tabla 5-4: Análisis e interpretación, pregunta 5 88
Tabla 6-4: Frecuencias Observadas, resultados de la encuesta 90
Tabla 7-4: Frecuencias Esperadas 90
Tabla 8-4: Cálculo de Chi Cuadrado 90
Tabla 9-4: Distribución Chi Cuadrado 91
Tabla 1-5: Medida de la eficiencia y funcionamiento del Sistema Biométrico 94

ÍNDICE DE FIGURAS

Figura 1-2. Crestas y valles de una huella dactilar 9
Figura 2-2. Patrón de Curva de una huella dígital 9
Figura 3-2. Patrón Espiral de una huella dígital 9
Figura 4-2. Arco normal y Arco Entoldado.....10
Figura 5-2. Puntos característicos de una huella dactilar. 11
Figura 6-2. Minucias de Terminación y Bifurcación. 11
Figura 7-2. Sensor Biométrico. 12
Figura 8-2. Arduino Mega 2560. 13
Figura 9-2. Entorno de desarrollo integrado Arduino. 14
Figura 10-2. Pantalla TFT-LCD Nextion NX4827T043 15
Figura 11-2. Editor de Pantallas TFT-LCD Nextion. 16
Figura 12-2. Módulo inalámbrico SIM800L. 17
Figura 13-2. Terminales y Tipos del SIM Card. 18
Figura 1-3. Tarjetas de Desarrollo Electrónicas. 24
Figura 2-3. Arquitectura de la red GSM. 26
Figura 3-3. Esquema de acceso al medio en GSM. 27
Figura 4-3. Topología de Red ZigBee. 28
Figura 5-3. Canales de Frecuencias usadas por IEEE 802.15.4-2006. 29
Figura 6-3. Módulos de Comunicación Inalámbrica. 30
Figura 7-3. Prevención de inyecciones SQL. 32
Figura 8-3. Diagrama de flujo del Equipo Terminal. 33
Figura 9-3. Tarjeta Principal. 34
Figura 10-3. Etapa de alimentación (Batería, Elevador y Reductor de Voltaje). 35
Figura 11-3. Imágenes para las páginas de la Interfaz gráfica. 36
Figura 12-3. Creación de proyecto de la Interfaz Gráfica. 36
Figura 13-3. Pantalla principal de control del Sistema Biométrico. 37
Figura 14-3. Página de ingreso de números de cédula. 39
Figura 15-3. Página tres y cuatro de visualización de datos. 40
Figura 16-3. Pines de conexión hacia el Arduino. 42
Figura 17-3. Conexión entre Arduino Pantalla Nextion. 43
Figura 18-3. Interfaz gráfica con la información de identificación. 43
Figura 19-3. Conexión Arduino, Sensor biométrico y Pantalla. 46
Figura 20-3. Conexión completa del Equipo Terminal. 51

Figura 21-3. Representación gráfica de la intensidad de la señal recibida. 53
Figura 22-3. Intensidad de la señal y nivel de la batería. 55
Figura 23-3. Equipo Terminal ensamblado en la caja 3D. 55
Figura 24-3. Diagrama de la Estación Base. 56
Figura 25-3. Tarjeta de transmisión Inalámbrica. 57
Figura 26-3. Servicios Apache y Mysql levantados. 58
Figura 27-3. Relación entre Tablas de Datos e Imágenes. 60
Figura 28-3. Administrador de origen de datos ODBC. 60
Figura 29-3. Conexión con la base de Datos. 61
Figura 30-3. Comunicación Serie 62
Figura 31-3. Ventana de registro de Huellas Digítales. 62
Figura 32-3. Conexión con la base de datos. 63
Figura 33-3. Ventana de Registro de Usuarios. 64
Figura 34-3. Proceso Digital de imágenes para la extracción características. 65
Figura 35-3. Descompresión y Construcción de la Imagen. 66
Figura 36-3. Selección de la Zona de Interés. 66
Figura 37-3. Imagen Original y Normalizada. 68
Figura 38-3. Imagen Segmentada. 69
Figura 39-3. Operador de Sobel Gx e Gy. 69
Figura 40-3. Mascara para aplicar un filtro pasa bajos. 70
Figura 41-3. Filtros para mejorar la imagen. 71
Figura 42-3. Binarización de la imagen 71
Figura 43-3. Pixel central y definición de sus vecinos 72
Figura 44-3. Imagen Adelgazada. 73
Figura 45-3. Ventana 3 x 3 para análisis de pixeles. 73
Figura 46-3. Condiciones para detectar bifurcaciones. 74
Figura 47-3. Condiciones para detectar terminaciones. 75
Figura 48-3. Minucias encontradas y marcadas. 75
Figura 49-3. Estructuras comunes de falsas minucias. 76
Figura 50-3. Imagen libre de Falsas Minucias 77
Figura 51-3. Obtención de Matriz de minucias. 79
Figura 52-3. Distancias entre plantillas de minucias del mismo dedo. 80
Figura 53-3. Ángulo de Minucias con respecto a la Recta. 80
Figura 54-3. Alta coincidencia entre Huellas Digitales. 81
Figura 55-3. Baja coincidencia entre Huellas Digitales. 81

Figura 56-3. Consulta de la Información y armado de la trama de datos. 83
Figura 57-3. Pantalla Principal de la Interfaz Gráfica. 83
Figura 1-4. (FAR) y (FRR) en funciones del umbral de aceptación (u). 85
Figura 2-4. Curva de Chi Cuadrado para comprobación de hipótesis. 91
Figura 1-5. Diagrama en Bloques del Sistema Biométrico. 92
Figura 2-5. Arquitectura del Sistema Biométrico de Acceso Inalámbrico. 93

ÍNDICE DE ANEXOS

ANEXO A: BAQUELITA EQUIPO TERMINAL Y ESTACIÓN CENTRAL

ANEXO B: PARTES DEL CHASIS EN 3D

ANEXO C: BASE DE DATOS DE HUELLAS DIGITALES

ANEXO D: FOTOS DE PRUEBAS REALIZADAS

ANEXO E: PROGRAMAS DEL ARDUINO

RESUMEN

Se desarrolló un sistema embebido cuya estructura se basa en una placa de Arduino Mega2560 que, en conjunto con un sensor biométrico, una pantalla nextion y un módulo inalámbrico GSM, constituyen un sistema biométrico de transmisión inalámbrica que tiene la capacidad de identificar personas a través de su huella digital. Este sistema captura la imagen de la huella dactilar de una persona mediante un sensor biométrico, digitaliza la información y genera una plantilla con los rasgos característicos, además permite ingresar números de cédula para enviar de forma inalámbrica hacia la estación base. El equipo receptor utilizando el módulo SIM800L captura la información y la transfiere hacia el computador para que el sistema identifique a quien corresponde. La interfaz diseñada lee cada imagen o número de cédula de la base de datos para compáralas con la información recibida desde el equipo terminal, realiza un procesamiento para mejorar la imagen y extraer sus rasgos característicos además de realizar consultas SQL, esto es con el fin de identificar a la persona a la que le corresponde la información tomada desde el Equipo Terminal. Los rasgos característicos almacenados se comparan con la plantilla recibida desde el equipo terminal, si existe coincidencia de aproximadamente quince minucias o más, se establece que las plantillas pertenecen a la misma imagen. Finalmente, se busca en la base de datos la información asociada a la imagen y se envían de forma inalámbrica hacia el equipo terminal para ser mostrados en pantalla y así identificar a las personas por medio de su huella.

Palabras clave: <ARDUINO>, <SENSOR BIOMÉTRICO>, <PLANTILLA>, <INTERFAZ>, <EQUIPO TERMINAL>, <ESTACIÓN BASE>, <MINUCIAS> <BASE DE DATOS>, <SQL>.

ABSTRACT

An embedded system was developed, which structure is based on an Arduino board Mega2560 that together with a biometric sensor, a nextion board and a wireless module GSM, constitute a biometric system of wireless transmission that has the capacity of identifying people through their fingerprints. This system captures the image of a person´s fingerprints through a biometric sensor, digitalizes the information and generates a template with its features. Additionally, it allows introducing ID numbers to be sent wirelessly to its base station. The receptor equipment using module SIM800L captures the information and transfers to another computer so the system identifies who this data belongs to. The designed interface reads each image or ID number of the database to compare them with the information received by the terminal equipment, it executes a procedure to improve the image and extracts its features, while it also makes SQL queries, the latter with the aim of identifying the person whom the information sent by the terminal belongs to. The stored features are compared to the template sent by the terminal equipment; stablishing at least fifteen minutiae to be recognised as that of the same image. Finally, the system activates a search within the database for the information associated to the image and send it back to the terminal equipment wirelessly, in order to be displayed on the screen so people are identified by their fingerprints.

Key words: <ARDUINO>, < BIOMETRIC SENSOR >, <TEMPLATE>, <INTERFACE>, <TERMINAL EQUIPMENT>, <BASE STATION>, <MINUTIAE> <DATABASE>, <SQL>.

CAPÍTULO I

INTRODUCCIÓN

Para identificar a los pacientes, algunos hospitales en Europa, están explorando el uso de la tecnología biométrica, específicamente el reconocimiento de huellas digitales. Emilio Gallego, CEO de Umanick, en una nota para el diario español El Mundo, habla que según la Organización Mundial de la Salud la identificación errónea de una persona, es la segunda razón de perjuicio al paciente por parte del sistema sanitario.

Umanick implementó su software de identificación en el Hospital de Día Virgen de la Arrixaca: esta solución es compatible con sistemas biométricos de huella dactilar, iris, rostro, a través de estos sistemas se realiza un trabajo de identificación tanto de pacientes y profesionales de la salud. En base a estos antecedentes, se ha realizado el diseño e implementación del prototipo, de un sistema biométrico de transmisión inalámbrica para la identificación y reconocimiento de personas a través de su historial médico.

El Sistema Biométrico está compuesto por un equipo terminal que toma la información de la persona y la transmite de forma inalámbrica hacia el módulo receptor ubicado en la estación base. El equipo receptor se comunica con el computador a través del puerto serie mediante el uso de una interfaz gráfica que se encarga de procesar la información para identificar a las personas registradas en la base de datos MariaDB, la búsqueda se realiza por medio de los rasgos característicos o utilizando el número de cédula.

Una vez que se ha detectado a la persona registrada mediante la información recibida, se imprime en pantalla sus datos y su huella digital. Estos datos son reenviados hacia el equipo terminal para ser visualizados en pantalla. Al obtener de forma inmediata la información, los médicos o paramédicos pueden proveer el tratamiento más adecuado en base a los antecedentes del individuo. Con esto se busca evitar las equivocaciones en la administración de medicamentos o tratamientos inadecuados a los pacientes.

1. PLANTEAMIENTO DEL PROBLEMA

1.1. Situación Problemática

Actual y antiguamente, las personas se han visto sujetas a cualquier tipo de percances o accidentes, lo que ocasiona que queden inconscientes o no puedan brindar su información. Esta situación normalmente causa que por desconocimiento de la condición médica o alergias que pueda tener una persona, se le suministre el medicamento equivocado, poniendo en riesgo la integridad física del paciente.

Una de las problemáticas presentes en los países europeos es la carencia de información sobre la salud y es considerada por los especialistas médicos como un elemento que atenta contra la efectivad de la atención (Alcívar y Yapur, sf). Esta situación ocasiona la demora del diagnóstico.

Los hospitales actualmente en Latinoamérica se los puede describir como estructuras sanitarias que implican una gestión y coordinación complejas. Y, si bien la tecnología se ha ido implantando con éxito y cada día cuentan con un mayor número de sistemas informáticos para facilitar la labor diaria de los profesionales (historia clínica electrónica, control farmacéutico, laboratorio, sistemas de radiología), no siempre se puede hacer uso de todo el potencial que ofrecen estos sistemas, ya que, entre otras cosas, su acceso se realiza principalmente usando ordenadores, mientras que el trabajo tanto de los profesionales, como la atención de los pacientes requieren desplazamientos continuos hacia el lugar de los hechos para realizar una valoración y por otro lado, no existe una sinergia de información pese a su potencial.

La presente investigación según la planificación estratégica del Plan Nacional de Desarrollo del Ecuador para el año 2017-2021 descrito por la Secretaria Nacional de Planificación y Desarrollo, plantea en el Contexto Histórico y Diagnóstico de la Realidad Nacional Actual: “en capacidades productivas se ha dado facilidades para la salud pública, infraestructura para la seguridad ciudadana de primera con equipamientos de punta” (Plan Nacional de Desarrollo, 2017-2021). Por tal motivo se ve la oportunidad de complementar el servicio brindado, llevando a cabo el desarrollo de un sistema biométrico inalámbrico que permite conocer de manera inmediata la información de la persona.

El Ecuador tiene la oportunidad histórica para ejercer soberanamente la gestión económica, industrial y científica, además señala que el país debe gestionar sus recursos estratégicos en el marco de una inserción internacional, que permita que el ciclo tecnológico actual basado en la automatización, la robótica y la microelectrónica, contribuya al incremento generalizado del bienestar para sus habitantes, esto permite generar riqueza y elevar en forma general el nivel de vida de nuestra población.

Específicamente el literal g del Plan Nacional de Desarrollo señala: Establecer mecanismos de transferencia de tecnología en la normativa de telecomunicaciones, para permitir el desarrollo local de nuevas aplicaciones” (Plan Nacional de Desarrollo, 2017-2021). Bajo este contenido se busca implementar el prototipo de un sistema biométrico inalámbrico para la identificación y reconocimiento de personas a través de su historial médico en el Hospital Regional Docente Ambato.

1.2. Formulación del Problema

¿Cómo Implementar un sistema biométrico de transmisión inalámbrica para la identificación y reconocimiento de personas a través de su historial médico en el Hospital Regional Docente Ambato?

1.3. Sistematización del Problema

- ¿Cuál es la necesidad para que una persona sea identificada junto con su historial médico?
- ¿Cuál tecnología es la más adecuada para implementar un sistema inalámbrico biométrico?
- ¿Se puede informar vía inalámbrica que un paciente ha sido ingresado para que preparen el mejor tratamiento en el hospital?
- ¿Es posible la creación de un sistema biométrico inalámbrico para la identificación y reconocimiento de personas a través de su historial médico?

1.4. Justificación de la Investigación

En el Ecuador no existe investigaciones sobre sistemas biométricos de transmisión inalámbrica para la identificación y reconocimiento de personas a través de su historial médico. El presente trabajo genera una base de datos con información relevante de las personas en el sistema de salud público, la cual sirve al sistema médico encargado del cuidado de las personas como fuente para la toma de decisiones y diagnósticos de forma rápida y eficiente.

En la actualidad la tecnología inalámbrica y biométrica se ha convertido en un instrumento primordial para la solución de problemas; por lo tanto, su utilización permite obtener información médica de forma inmediata de cualquier persona registrada. El sistema biométrico, permite digitalizar las características propias que posee cada persona en su huella digital mediante el procesamiento digital de imágenes para enviar de forma inalámbrica la información; a través de una base de datos los encargados de la salud, tienen acceso a la identidad e historial médico de los pacientes con el fin de brindarles la atención medica más adecuada, dicha información es reenviada al lugar de los hechos para proveer inmediatamente los primeros auxilios.

De conformidad con el Plan Nacional de Desarrollo, según lo establecido en el Acceso Equitativo a Infraestructura, Equipamiento y Conocimiento: “El Ecuador tiene la oportunidad histórica para ejercer soberanamente la gestión económica, industrial y científica, de sus sectores estratégicos, además señala que el país debe gestionar sus recursos estratégicos en el marco de una inserción internacional, que permita que el ciclo tecnológico actual basado en la automatización, robótica, microelectrónica y telecomunicaciones, contribuya a crear bienestar para sus habitantes”. Con la aplicación del sistema biométrico, se apunta directamente a contribuir con el desarrollo, proponiendo una solución de tecnología aplicada a un problema existente, como es la atención medica rápida y eficiente de las personas.

1.5. Objetivos de la Investigación

1.5.1. Objetivo General

Diseñar e implementar un sistema biométrico de transmisión inalámbrica para la identificación y reconocimiento de personas a través de su historial médico en el Hospital Regional Docente Ambato

1.5.2. Objetivos Específicos.

- Investigar y seleccionar la base de datos, tecnologías de comunicación inalámbrica y tarjetas de desarrollo que se ajusten a los requerimientos del sistema.
- Diseñar un sistema de detección biométrico de acceso inalámbrico y una base de datos de prueba para almacenar la información médica de las personas.
- Implementar un prototipo del sistema biométrico de transmisión y recepción inalámbrica y base de datos para registrar el historial médico.
- Realizar la validación del sistema biométrico de transmisión y recepción de datos.

1.6. Hipótesis

1.6.1. Hipótesis General

El sistema biométrico de transmisión inalámbrica permite la identificación y reconocimiento de personas a través de su huella digital o número de cédula y hace más eficiente la atención médica.

1.6.2. Hipótesis Específicas

- La implementación de un sistema biométrico permite digitalizar las características propias que posee cada persona en su huella digital.
- El sistema biométrico de transmisión inalámbrica permite enviar de forma inmediata la información digitalizada del paciente al hospital.
- El acople de distintas tecnologías utilizadas genera eficiencia en la comunicación de datos, esto mejora la toma de decisiones al tratar a una persona en el lugar de los hechos.
- El sistema implementado informa en el centro de atención médico que un paciente se encuentra en camino y despliega sus datos personales y médicos en el sistema para que los profesionales decidan cual es el mejor tratamiento al momento de su llegada.
- El sistema implementado reenvía la información del paciente para que los datos sean visualizados en una pantalla y se pueda brindar los primeros auxilios en base su información.

1.6.3. Identificación de variables

- Variable dependiente: sistema biométrico de transmisión inalámbrica.
- Variable Independiente: identificación y reconocimiento de personas para una atención médica más eficiente.

CAPÍTULO II

2. MARCO DE REFERENCIA

2.1. Antecedentes del Problema

La Organización Mundial de la Salud, hace mención sobre la telemedicina a nivel global y de las oportunidades que se puede obtener: “Telemedicine: opportunities and developments in member states”. Dentro de las conclusiones principales de un estudio llevado a efecto, está el reconocimiento de los países más desarrollados como aquellos en los que este tipo de soluciones están siendo implementadas.

Existe un déficit de políticas y respaldo gubernamental para implementar servicios basados en TIC, también se necesita involucrar a todo el personal médico para que la implementación de estos servicios sea un éxito (es decir, que médicos, pacientes, cuidadores y técnicos se sumen de manera conjunta en proyectos de telemedicina). Finalmente existe la necesidad de realizar una evaluación sistemática y objetiva de los beneficios que aportan estos sistemas. (Perdices, 2016)

La Carpeta de Salud, es una herramienta interactiva online que posibilita a cada paciente acceder a su historial médico y consultar informes radiológicos, análisis, subir documentos, hablar con su médico, etc. Así lo expresaron el consejero de Salud del Gobierno Vasco, Jon Darpón, y el director de Asistencia Sanitaria de Osakidetza, Andoni Arcelay. Darpón señalo que el objetivo, es alentar a las personas a que accedan a través de Internet a su historial médico y sean autónomas en la gestión de su salud.

Hoy por hoy, Darpón es el canal más potente por los servicios que ofrece y las oportunidades que brinda. Una de sus ventajas de ‘Carpeta de Salud’ es que se trata de una herramienta online y, por tanto, está disponible las 24 horas del día, todos los días del año. Además, su acceso se puede realizar desde un smartphone, una tablet o un ordenador mediante la web de Osakidetza “www.osaki detza.euskadi.eus” o descargando la app de Osakidetza. (Carpeta de Salud, 2018).

El Consorcio Hospitalario Provincial de Castellón ha iniciado la implantación de la historia clínica electrónica móvil en las plantas de hospitalización para mejorar el servicio que presta tanto a los pacientes como a sus familias. Se trata de un novedoso servicio que también se ha implantado en la Unidad de Hospitalización a Domicilio (UHD) (Noticias, 2013). El personal médico y de enfermería de las plantas de hospitalización del Consorcio Hospitalario Provincial de Castellón cuenta con tabletas táctiles de última generación que le permite acceder a la historia clínica electrónica de los pacientes a pie de cama, lo que proporciona una atención “más rápida, segura y eficaz”, en las plantas de hospitalización, el Servicio de Informática ha habilitado una red local

inalámbrica, que permite consultar las historias clínicas electrónicas de los pacientes desde sus habitaciones. (Noticias, 2013)

En un esfuerzo por modernizar el sistema sanitario de Estados Unidos, el presidente Bush, en 2004, concluyó que la mayoría de los historiales clínicos electrónicos americanos deberían estar conectados antes del 2015 (Hesse, 2010). En esta misma línea, la iniciativa Health Information Technology for Economic and Clinical Health (HITECH) del presidente Obama ha hecho que se despierte un gran interés por las historias clínicas electrónicas de acceso inalámbrico.

Estados Unidos muestra un retraso para el acceso al historial médico de las personas, con respecto a la mayoría de los países de la Unión Europea en cuanto a la implementación de este tipo de sistemas. En España, cada comunidad autónoma ha desarrollado su propio sistema digital de historias clínicas, con grandes similitudes, y es habitual que las comunidades compartan experiencias y buenas prácticas.

Los sistemas de historias clínicas electrónicas pueden proporcionar grandes beneficios, ya que la información de un paciente procede de múltiples organizaciones sanitarias: 1) se obtiene información completa integrada, 2) evita duplicidades e inconsistencias; y 3) hay una alta disponibilidad de los datos. (Carnicero, 2011) Los aspectos de privacidad y seguridad en los sistemas de historias clínicas electrónicas son de vital importancia.

En el Ecuador no existen investigaciones desarrolladas sobre sistemas de identificación de personas de acceso remoto vía internet o inalámbrico, sin embargo, se ha aprovechado los recursos que se posee para implementar un prototipo con el fin de enviar de forma inalámbrica la información tomada de las personas, esto es con el objetivo de reconocer al paciente mediante su ficha médica ya que los datos consultados son reenviados y visualizados en una pantalla, esto evita que se produzcan accidentes al suministrar algún tratamiento que podría poner en peligro la integridad física de una persona.

2.2. Bases Teóricas

2.2.1. Biometría.

Biometría es el conjunto de características físicas y de comportamiento que se utilizan para verificar la identidad de algún individuo. Biometría proviene de las palabras bio (vida) y metría (medida), por lo tanto, todo equipo o sistema biométrico mide e identifica las características propias de las personas. Para la identificación de personas, se puede establecer dos tipos, la biometría estática para referirse al estudio del conjunto de características físicas y la biometría dinámica para referirse al conjunto de características conductuales. La biometría estática establece las siguientes características: Huella dactilar, rostro y características del ojo (retina e iris). Dentro de la biometría dinámica nos encontramos con la voz.

2.2.1.1. Huellas Dactilares

Las huellas dactilares se forman en la epidermis de la piel en los dedos. Para capturar las huellas, se utiliza sensores electrónicos que almacenan la información en una memoria interna, los algoritmos usan técnicas basadas en minucias, crestas y correlación para determinar su similitud, las principales ventajas de estos métodos, es que la imagen es fácil de adquirir, los tiempos de comparación son cortos y los equipos son económicos.

Actualmente se encuentran muchas aplicaciones en esta área que involucran control de acceso a zonas restringidas, aplicaciones civiles (biométricos de control de asistencia), y forenses (que sirven para determinar la identidad de una persona).

2.2.1.2. Rostro

El análisis del rostro es uno de los métodos que usualmente se utiliza para identificar una persona de otra. Sin embargo, los algoritmos trabajan bien cuando se tiene un rostro en condiciones adecuadas, en ella se buscan proporciones entre los ojos, cejas, ancho de la boca, la nariz y el mentón. En la actualidad existen programas comerciales que pueden comparar rostros, pero tienen limitaciones en cuanto a la iluminación y la pose del rostro.

2.2.1.3. Iris y Escaneo Retinal

La textura del ojo es determinada por la morfo-genética caótica durante el desarrollo del embrión, la textura del ojo continúa cambiando hasta los dos años, después de ello se estabiliza, sin embargo, el pigmento sigue cambiando durante toda la vida. La captura del iris se hace mediante cámaras de buena resolución y con adaptación óptica, por ello al principio fueron costosas. Estos dispositivos son extremadamente precisos y rápidos.

La retina vascular del ojo es rica en su estructura y se puede considerar como única en cada persona. Actualmente es uno de los sistemas biométricos más seguros que existen, ya que no es fácil cambiar o replicar la retina vascular, debido a que puede contener hasta 400 puntos característicos.

2.2.1.4. Voz

Siempre que una persona habla por teléfono, se puede determinar a quién pertenece esa voz. Las características de la individualidad de la voz dependen de la forma y tamaño de las cuerdas vocales, boca, cavidad nasal y labios, por ello que es aceptado como una medida biométrica. Sin embargo, no puede ser utilizado para identificar a una persona ya que se degrada con el tiempo, depende del estado emocional, la salud, y existen personas con una extraordinaria habilidad para imitar la voz de otros.

2.2.2. Comunicación inalámbrica

La comunicación inalámbrica, es la conexión entre el emisor y el receptor sin la necesidad de un agente o un hardware que conecte ambos puntos físicamente, para esto se utiliza la modulación de ondas electromagnéticas a través del espacio. Los dispositivos físicos sólo están presentes en los emisores y receptores de la señal, entre los cuales encontramos: antenas, computadoras portátiles, teléfonos móviles, etc.

2.2.2.1. Tipos de redes inalámbricas

- **Red inalámbrica de Área Personal (WPAN):** este tipo de redes son de corto alcance de aproximadamente 10 metros. Tecnologías como el Bluetooth pertenecen a estas redes, se emplean en periféricos (teclado, ratón, etc.) (Econectia, 2018).
- **Red Inalámbrica de Área Local (WLAN):** tecnologías como WiFi se encuentran en este tipo de red. WiFi convierte una conexión (ADSL, fibra óptica) en ondas de radio frecuencia y permite la interconexión de dispositivos como portátiles, tabletas, etc) (Econectia, 2018).
- **Red Inalámbrica de Área Metropolitana (WMAN):** en estas redes, se emplean tecnologías como WiMax que permiten la interconexión con gran cobertura y ancho de banda, utilizado para tener internet en zonas sin cobertura (Econectia, 2018).
- **Red Inalámbrica de Área Amplia (WWAN):** este tipo de redes inalámbricas son de largo alcance. Tecnologías como GSM o GPRS pertenecen a estas redes, se emplean para la telefonía móvil o smartphones (Econectia, 2018).

2.2.2.2. Sistema Global para las Comunicaciones Móviles (GSM)

GSM es un estándar muy utilizado, también conocido como 2G, representa un salto de las comunicaciones analógicas a las digitales. La banda de frecuencia en la que opera corresponde a los 850/900/1800/1900MHz. Los equipos que se utilizan se denominan estaciones móviles. Para operar, se necesita una tarjeta SIM, este chip contiene información sobre el terminal, operador de red, tipo de contrato y usuario.

2.3. Marco Conceptual

2.3.1. Huella Dactilar

Las huellas dactilares son los parámetros biométricos más utilizados para la identificación de personas. Corresponden a una representación de la epidermis de los dedos caracterizada por un conjunto de líneas que forman un patrón único de crestas y valles en la superficie del dedo. Una cresta se define como un segmento de curva, y un valle como la región entre dos crestas adyacentes (Shea, 2004). Es posible ver los componentes de una huella digital en la Figura 1-2.

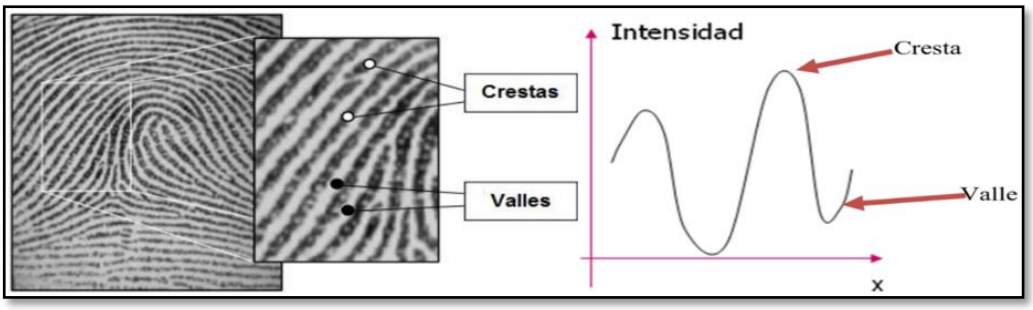


Figura 1-2. Crestas y valles de una huella dactilar
Fuente: (Sarzuri)

2.3.1.1. Tipos de patrones de las huellas Digitales.

Las huellas digitales están formadas por valles y crestas que se encuentran en la mano de las personas, en la yema de los dedos estos valles y crestas forman tres patrones: Curva o Bucle, Espiral y Arco.

- En el patrón Bucle existen dos rasgos característicos, estos son el núcleo o core y delta. Delta es el área donde se forma una triangulación o división de las líneas (Barbazán & Candela , 2017). El patrón curva es el más común. Se encuentra en el 70 por ciento de la población. En este patrón las aristas forman un bucle inclinado hacia arriba que se apoya en el delta. Sólo un delta está presente en los patrones de bucle. La Figura 2-2, muestra el patrón de curva.

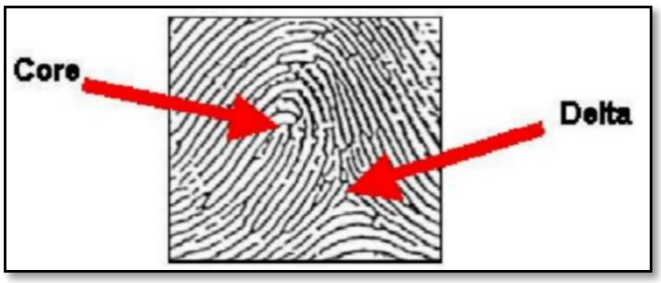


Figura 2-2. Patrón de Curva de una huella dígital
Fuente: (Barbazán & Candela, 2017)

- "Un patrón Espiral tendrá dos o más deltas" (Barbazán & Candela , 2017). Aproximadamente el 25 al 35 por ciento de las huellas digitales son espirales. Un delta debe aparecer en ambos lados del bucle. La Figura 3-2, muestra el patrón Espiral.

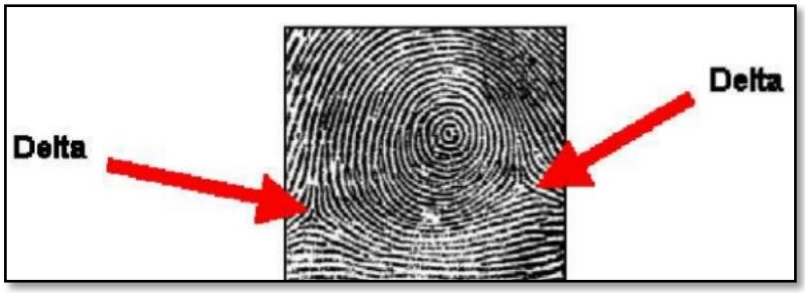


Figura 3-2. Patrón Espiral de una huella dígital
Fuente: (Barbazán & Candela, 2017)

- Las huellas con patrones de arco son la más raras, se encuentra solo en el 5 por ciento de la población. En los patrones de arco las líneas de cresta son divergentes, sin curvas deltas presentes. Existen dos tipos específicos de patrones de arco, la Figura 4-2, muestra el arco normal y el arco entoldado:

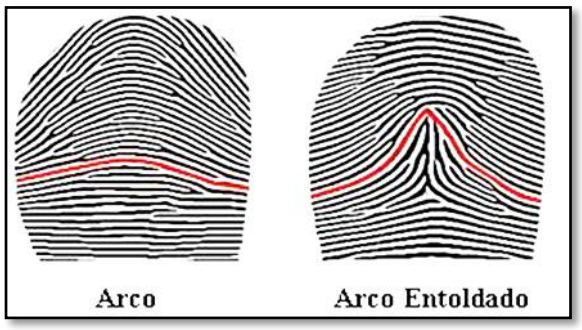


Figura 4-2. Arco normal y Arco Entoldado
Fuente: (Segura, 2012)

1. El más simple es el arco normal donde las líneas continúan de lado a lado con la dirección ligeramente hacia arriba, visualmente similares a las olas del mar.
2. El arco de tiendas de campaña o entoldado tiene unas líneas más nítidas hacia arriba, se encuentran típicamente en el centro de la impresión de la huella.

2.3.1.2. Puntos característicos de una huella Dactilar

Estos puntos también llamados Minucias, son rasgos característicos de las huellas dactilares y se representan de la siguiente manera:

$$minucias = \{x, y, \theta, tipo\},$$

dónde:

- X e Y representan la posición en la imagen de la huella.
- θ representa el ángulo de las minucias (en radianes).
- Tipo, pueden ser terminaciones o bifurcaciones.

Las minucias se dividen en varios tipos como se observa en la Figura 5-2. Principalmente se manejan dos tipos de minucias: las bifurcaciones y las terminaciones, de aquí se desprenden todas las demás formas que se usan en el sistema Henry.

- **Bifurcación:** Una minucia de bifurcación es donde una cresta se divide, formando un ángulo más o menos agudo.
- **Terminación:** Una minucia de terminación es donde la cresta finaliza.
- **Cortada o interrupción:** Línea discontinua, que a lo largo de su recorrido se interrumpe en una o varias ocasiones.

- **Empalme:** Entre dos líneas que van en paralelo, existe otra que las une en diagonal.
- **Ojal o Encierro:** Es la unión de 2 líneas, las cuales forman un círculo en medio.
- **Extremo de línea:** Línea que queda interrumpida en sus extremos.
- **Horquilla:** Línea que se une a otra, pero sin formar un ángulo.
- **Islote:** Línea que es un poco más grande que el punto, formada por 2 o más puntos.
- **Punto:** Como su nombre indica es un punto, y es lo más pequeño que es posible encontrar en la cresta papilar.

Figura 5-2. Puntos característicos de una huella dactilar.
Fuente: (Edward, 1900)

Esta investigación se enfoca en las minucias de terminación y bifurcación, debido a que los otros tipos se forman mediante la combinación de estas dos. En la Figura 6-2, se puede observar los dos tipos, como se representan, la posición y el ángulo.

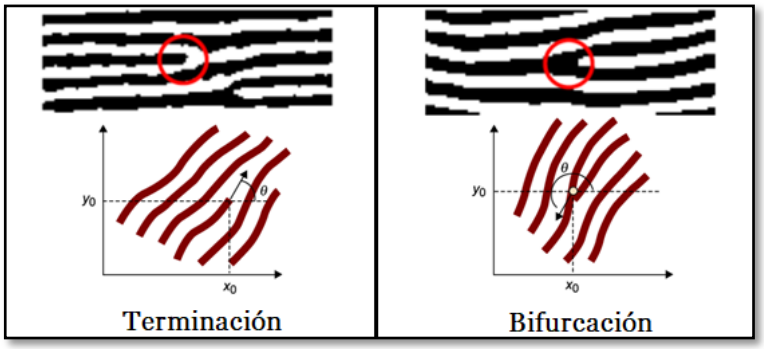


Figura 6-2. Minucias de Terminación y Bifurcación.
Fuente: (Rosales, 2009)

2.3.2. Lector biométrico para la adquisición de Imágenes.

El Lector Biométrico que se observa en la Figura 7-2, transforma en datos binarios las imágenes para que puedan ser procesadas digitalmente gracias a un Procesador Digital de Señales (DSP) incorporado, realiza una serie de funciones como la inscripción y coincidencia de las huellas digitales, búsqueda y almacenamiento de plantillas. El dispositivo se comunica con cualquier sistema embebido utilizando el protocolo serial (Torres, 2018).

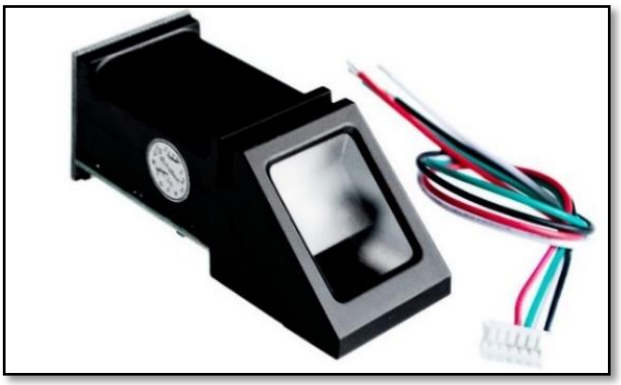


Figura 7-2. Sensor Biométrico.
Fuente: En "Lector de Huellas Dactilares Sensor Biométrico Serial TTL"

2.3.2.1. Características:

La Tabla 1-2, muestra las características técnicas que posee el sensor biométrico.

Tabla 1-2: Características técnicas del sensor biométrico.

Sensor Biométrico	
Voltaje de alimentación:	3.6V - 6V
Corriente de operación:	100mA - 150mA
Interfaz:	Serial/UART TTL
Modo de paridad de la huella:	1:1 1: N
Tasa de transferencia o Baud Rate:	9600*N
N:	de 1 a 12 (Por defecto es 6)
Tiempo de adquisición:	menor a 1 segundo
Niveles de seguridad:	5
Dimensiones de la ventana:	14 x 18mm
Dimensiones de la estructura:	5.5 x 2.1 x 2.0 cm
Peso:	22g

Fuente: (Grow, 2015)
Realizado por: Salazar Byron, 2019

2.3.2.2. Recursos del sistema

El sistema biométrico cuenta con los siguientes recursos:

- **Bloc de notas:** El sistema reserva una memoria de 512 bytes, donde se pueden almacenar datos que requieren protección al apagar el módulo.

- **Búfer de la imagen:** ImageBuffer almacena las imágenes, el tamaño es de 256 * 288 píxeles. Cuando se transfiere a través de UART, para acelerar la velocidad, solo se transfieren los 4 bits superiores del píxel (es decir, 16 grados de gris). Y dos píxeles adyacentes de la misma fila formarán un byte antes de la transferencia. Cuando se cargue en la PC, la imagen de 16 grados de gris se ampliará al formato de 256 grados de gris. Se almacena en formato BMP de 8 bits. Cuando se transfiere a través de USB, la imagen es un píxel de 8 bits, es decir, 256 grados de gris.
- **Búfer de archivo de caracteres:** Los búferes CharBuffer1, CharBuffer2, se usan para almacenar tanto el archivo de caracteres como el de la plantilla de minucias.
- **Biblioteca de huellas dactilares:** El sistema establece un cierto espacio dentro de la Flash o biblioteca para el almacenamiento de plantillas de las huellas dactilares.

2.3.3. Arduino Mega 2560

La tarjeta Arduino mega que se observa en la Figura 8-2, es una plataforma de hardware libre, basada en una placa con un microcontrolador, bootloader ejecutado en la placa y el entorno de desarrollo integrado por sus siglas en inglés Integrated Development Environment. La principal característica del software de programación es su sencillez y facilidad de uso (Crespo, 2018). Bootloader, es un programa que permite cargar código sin necesidad de hardware adicional.

2.3.3.1. Características:

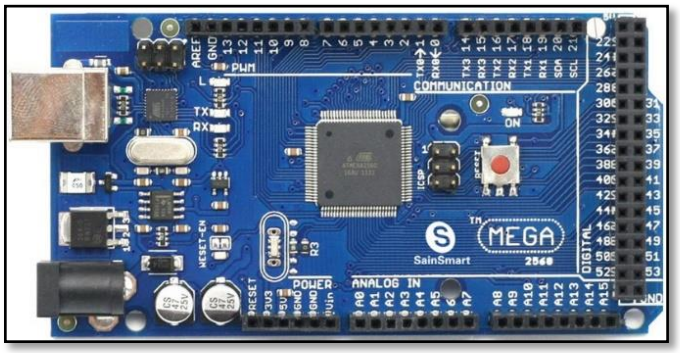


Figura 8-2. Arduino Mega 2560.
Fuente: (Bernardo, 2018)

La Tabla 2-2, muestra las características y recurso técnicos del Arduino ATmega2560.

Tabla 2-2: Características Técnicas del Arduino ATmega2560

Arduino ATmega2560	
Voltaje de alimentación:	5V
Voltaje de Entrada (Recomendado):	7-12V
Voltaje de Entrada(límites):	6-20V
Pines digitales de Entrada/Salida:	54
Pines digitales de Salida PWM:	15
Pines análogos de entrada:	16

Corriente DC por cada PIN Entrada/Salida:	40 mA
Corriente de CC para 3,3v PIN:	50 mA
Memoria Flash:	256 KB
Memoria Flash para Bootloader:	8KB
SRAM:	8KB
EEPROM:	4KB
Clock Speed	16 MHz
Largo X Ancho	101,52mm X 53,3mm

Fuente: (Bernardo, 2018)
Realizado por: Salazar Byron, 2019

2.3.3.2. Comunicación

El Arduino Mega2560 posee cuatro puertos UART de hardware para comunicación serie TTL (5V). Un ATmega8U2 en la placa convierte el puerto físico serial 0 a uno virtual para que el Arduino se conecte con la computadora (las máquinas que utilizan Windows necesitarán un archivo .inf, pero las máquinas OSX y Linux reconocen la placa como un puerto COM automáticamente.

El software Arduino incluye un monitor en serie que permite enviar datos de texto simples hacia y desde la placa. Los leds RX y TX en la placa parpadean cuando los datos se transmiten a través del chip ATmega8U2 hacia la computadora. Una biblioteca SoftwareSerial permite la comunicación en serie en cualquiera de los pines digitales.

2.3.3.3. Programación

El Arduino Mega se puede programar en el entorno de desarrollo integrado "Arduino" o IDE (Integrated Development Environment) por sus siglas en inglés. Es un programa informático compuesto por un conjunto de herramientas de programación como: un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI), incorpora las herramientas para cargar el programa ya compilado en la memoria flash del hardware. La Figura 9-2, muestra el entorno gráfico. (Crespo E. , 2017).

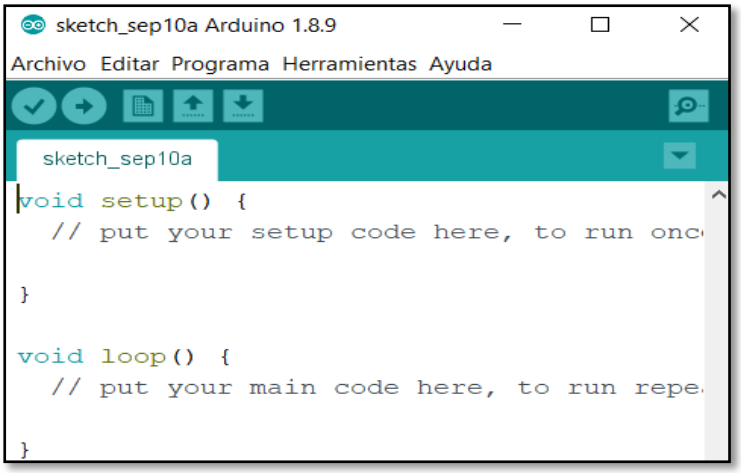


Figura 9-2. Entorno de desarrollo integrado Arduino.
Realizado por: Salazar Byron, 2019

2.3.4. Pantalla Nextion TFT-LCD NX4827T043

Las siglas TFT significan ("Thin Film Transistor") ó transistor de película delgada, es una pantalla que opera como interfaz hombre maquina (HMI), es resistiva formada por varias capas, cuando se presiona sobre la pantalla estas capas entran en contacto y se produce un cambio en la corriente eléctrica lo que permite detectar la pulsación (Oscar B, 2009). Nextion incluye una parte de hardware y una parte de software (Nextion Editor). La pantalla Nextion TFT utiliza el puerto serie para comunicarse con otros dispositivos. La Figura 10-2, muestra la pantalla Nextion.

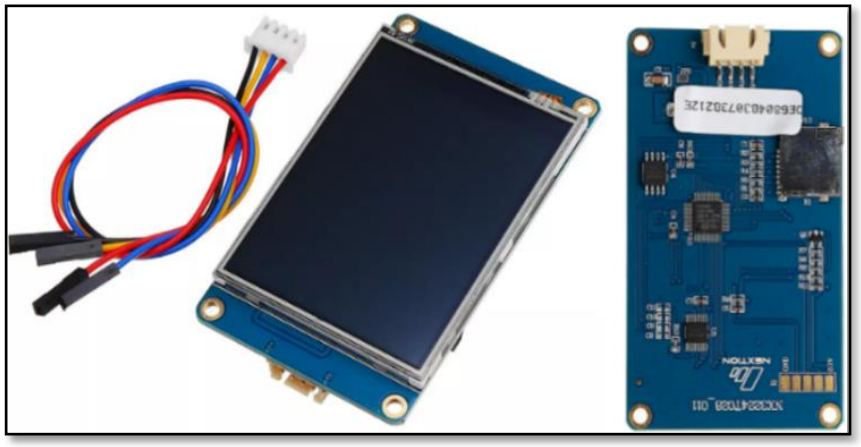


Figura 10-2. Pantalla TFT-LCD Nextion NX4827T043
Fuente: (banggood.com, s.f.)

2.3.4.1. Características:

La Tabla 3-2, muestra las características técnicas de la Pantalla Nextion.

Tabla 3-2: Características Técnicas de la Pantalla Nextion NX4827T043.

Pantalla Nextion NX4827T043	
Tamaño:	4,3”
Resolución:	480 x 272
Pantalla TFT:	Panel resistivo táctil
Interfaz de 4 pines TTL:	Transmisor, Receptor, GND, VCC
Memoria Flash:	16MB para Código de usuario y datos
Ranura para tarjeta micro-SD:	Para actualización de firmware
Core:	ARM 7 48 MHZ
Área de visualización:	95.04mm(L)×53.86mm(W)
Consumo de energía:	5V, 250mA
Corriente de operación (sleep mode)	15 mA
Brillo ajustable:	0~230 nit, intervalo de ajuste de 1%

Fuente: (banggood.com, s.f.)
Realizado por: Salazar Byron, 2019
Nota. Nit=cantidad de luz que es capaz de emitir una pantalla, equivale a una candela por metro cuadrado, a mayor cantidad, mayor luz emitirá una pantalla

2.3.4.2. Editor Nextion

El editor de Nextion, tiene componentes masivos para enriquecer el diseño de la interfaz (Lara, 2015), está dividido en seis secciones. En la columna de la izquierda se encuentran dos secciones, arriba esta una lista con elementos como: textos, números, botones, pulsadores, imágenes, barras de progreso, sliders que se puede introducir en la pantalla, en la parte inferior esta la lista de fuentes e imágenes.

En la columna central se encuentra una representación visual que muestra como quedarán los elementos en pantalla, en la parte inferior estás dos ventanas que permiten programar todos los controles utilizados. La columna de la derecha tiene un primer cuadro con todas las páginas que se muestran en pantalla y en la parte inferior está la tabla de atributos con la cual se configura los elementos empleados. Finalmente, la Figura 11-2, muestra el Editor de Pantallas Nextion.

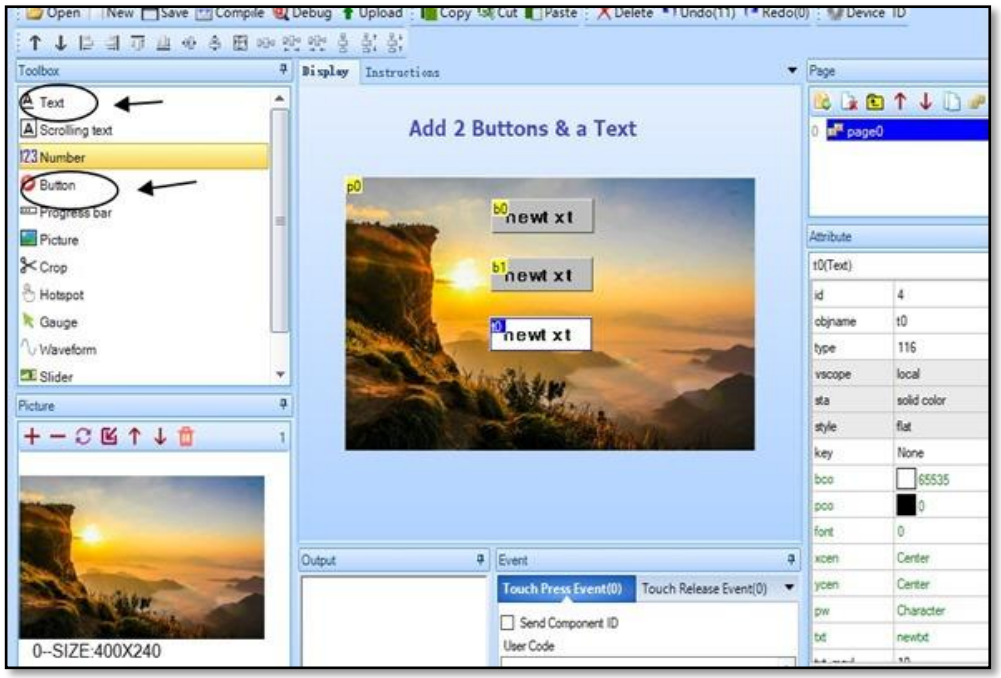


Figura 11-2. Editor de Pantallas TFT-LCD Nextion.
Fuente: (alselectro, 2019)

2.3.5. Módulo SIM800L

El SIM800L es un módulo celular de bandas liberadas, permite la transmisión GPRS, enviar y recibir SMS, hacer y recibir llamadas de voz. Después de conectar el módulo, este busca la red celular e inicia sesión automáticamente. Mediante el Arduino se envía y recibe instrucciones, con el fin de transmitir información de forma inalámbrica utilizando comandos AT. La Figura 12-2, muestra el módulo SIM800L.

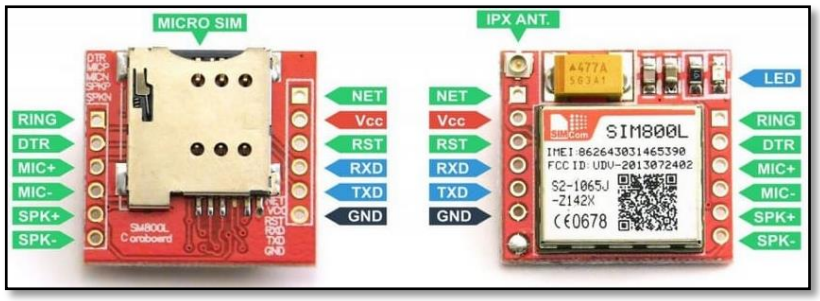


Figura 12-2. Módulo inalámbrico SIM800L.
Fuente: (Descubrearduino, 2014)

GPRS (General Packet Radio Service) o servicio general de paquetes vía radio, es una extensión mejorada del GSM. Permite la mensajería instantánea, servicio de mensajes cortos o SMS (Short Message Service), servicio de mensajes multimedia o MMS (Multimedia Messaging Service) y servicio de correo electrónico (Blasco, 2016).

2.3.5.1. Especificaciones Técnicas

La Tabla 4-2, muestra las características técnicas del módulo inalámbrico SIM800L.

Tabla 4-2: Características técnicas del módulo SIM800L.

Módulo SIM800L	
Voltaje de Operación:	3.4V ~ 4.4V DC
Nivel Lógico:	3V a 5V
Consumo de corriente (máximo):	500 mA
Consumo de corriente (modo sleep):	0,7 mA
Interfaz:	Serial UART
Velocidades de transmisión serial:	1200bps hasta 115200bps
Frecuencia de operación (Cuatribanda):	850/900/1800/1900MHz
Envía y recibe mensajes SMS:	Modo texto y PDU.
Envía y recibe:	Datos GPRS (TCP/IP, HTTP)
Controlado por:	Comandos AT
Velocidad máxima de transmisión:	85.6 Kbps
Chip:	Protocolo TCP/IP
Tamaño de la SIM:	Micro SIM

Fuente: (Descubrearduino, 2014)
Realizado por: Salazar Byron, 2019
Nota. Detección "automática" de velocidad de transmisión. Soporta Reloj en tiempo real (RTC). Audio Cancelación de Eco y supresión de ruido

2.3.6. Módulo de identificación de abonado (SIM)

El módulo SIM (Subscriber Identity Module) o módulo de identificación de abonado, es una tarjeta inteligente removible, se usa en teléfonos móviles y módems, estos dispositivos se conectan por medio de una ranura o lector SIM. Las tarjetas SIM contienen la clave de servicio del suscriptor usada para identificarse en la red, de esta forma es posible cambiar la suscripción del cliente de un terminal a otro únicamente cambiando la tarjeta.

El uso de la tarjeta SIM es obligatorio en las redes GSM. Un teléfono móvil sin una tarjeta SIM no funciona. La capacidad de almacenamiento más común es de entre los 16 y 32 Kb (Maugard, 2015). El chip posee ocho terminales metálicos debidamente estandarizados. Por medio de estos contactos, el lector alimenta eléctricamente a la tarjeta y transmite los datos para operar con ella. La Figura 13-2, muestra el módulo SIM.

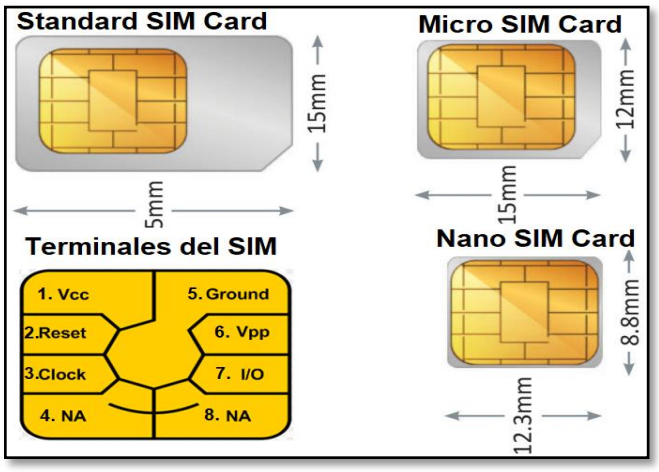


Figura 13-2. Terminales y Tipos del SIM Card.
Realizado por: Salazar Byron, 2019

2.3.7. Servidores

Un servidor es un ordenador u otro tipo de equipo informático encargado de suministrar información a una serie de clientes, que pueden ser tanto personas como otros dispositivos conectados a él. La información que puede transmitir es múltiple y variada: desde archivos de texto, imágenes, vídeo y hasta programas informáticos.

2.3.7.1. Servidor Xampp

Es un servidor de plataforma libre que integra en una sola aplicación varios paquetes, que se configuran en forma automática, lo que permite ejecutar un servidor web completo. Xampp permite instalar los siguientes paquetes:

- Apache Web Server
- Base de datos MariaDB
- Administrador de base de datos phpMyAdmin

2.3.7.2. Apache

Apache es un servidor HTTP de código abierto utilizado en plataformas Unix (BSD, GNU/Linux, etc.), Windows, Macintosh especialmente. Este software es necesario para montar un servidor local. (AppServ, 2018)

2.3.7.3. MariaDB

MariaDB es un sistema de gestión de base de datos relacional de código abierto (RDBMS por sus siglas en inglés), está basado en lenguaje de consulta estructurado (SQL), se ejecuta prácticamente en todas las plataformas, como Linux, UNIX y Windows. MySQL está asociado con las aplicaciones basadas en la web y publicaciones en línea. (Rouse, 2005-2018)

2.3.7.4. PHPMyAdmin

PHPMyAdmin facilita el desarrollo de un sitio web dinámico asociado a una base de datos, permite administrar dicha base sin la necesidad de recurrir a la escritura de líneas de comandos en la consola del equipo, ya que proporciona un entorno gráfico fácil de usar y muy intuitivo. (AppServ, 2018)

2.3.8. Seguridad de base de datos

La mayoría de información sensible del mundo está almacenada en sistemas gestores de bases de datos como MySQL, MariaDB, Oracle, Microsoft SQL Server entre otros. Toda esa información es la que hace que los hackers centren su esfuerzo en poder acceder a los mismos por medio de las vulnerabilidades que se puede encontrar referente a estos gestores.

Las vulnerabilidades pueden ser debidos a softwares no actualizados a la última versión, ya que estos pueden poseer problemas de seguridad, a la forma en la que está configurado su acceso o bien problemas en la programación de la aplicación, problemas que pueden causar el conocido ataque SQL Injection, uno de los ataques más comunes cuando de bases de datos se trata.

Hasta este momento, gran parte del esfuerzo para mejorar la seguridad de cualquier servicio informático se centra en asegurar los perímetros de las redes por medio de firewalls, IDS / IPS y antivirus. A continuación, se describe las vulnerabilidades más habituales que se puede encontrar a la hora de trabajar con bases de datos.

2.3.8.1. Contraseña/password

Existe poca seguridad al utilizar un password en blanco o bien al hacer uso de uno débil, hoy en día no es raro encontrarnos pares de datos usuario/password del tipo admin/12345 o similar. Esta es la primera línea de defensa de entrada a la información, por lo tanto, se debe optar por el uso de algo más complejo que sea complicado de conseguir por parte de cualquier atacante.

2.3.8.2. Preferencia de privilegios de usuario

En ocasiones muchos usuarios reciben más privilegios sobre la base de datos de los que realmente necesitan, lo que a la larga ocasiona un problema. Es recomendable modificar los privilegios otorgados a los usuarios que estarán en contacto con la información con el fin de que no puedan realizar modificaciones más allá de las autorizadas. Por ejemplo, sí un usuario sólo realizará

consultas a la base de datos, pero no podrá modificar ningún registro, no tiene sentido que tenga privilegios para modificar ya que lo que se hace es abrir una puerta para un eventual ataque.

2.3.8.3. Características de bases de datos innecesariamente habilitadas

Cada instalación de base de datos viene con una serie de paquetes o módulos adicionales de distintas formas y tamaños que en pocas ocasiones son utilizadas, lo que las convierten en una posible puerta de entrada para sufrir algún tipo de ataque si en esos paquetes se descubre cualquier problema de seguridad. Para reducir riesgos, es recomendable detectar paquetes que no se utilicen y desactivarlos del servidor donde estén instalados. Esto no sólo reduce los riesgos de ataques, sino que también simplifica la gestión de parches ya que únicamente será de máxima urgencia actualizar aquellos que hagan referencia a un módulo que estemos utilizando.

2.3.8.4. Bases de datos sin actualizar

Como ocurre con cualquier tipo de aplicación instalada en los equipos, es necesario ir actualizando la versión de la base de datos con las últimas versiones lanzadas al mercado, ya que en ellas se solucionan aquellos problemas de seguridad detectados, esto pone más barreras a los posibles atacantes.

2.3.8.5. Datos sensibles sin cifrar

No todo el mundo cifra la información más importante que se almacena en base de datos. Esto es una buena práctica para que, en caso de hackeo, sea complicado para el atacante recuperar la información. Las contraseñas de acceso a un sitio por parte de los usuarios pueden ser cifradas utilizando el algoritmo MD5. De esta forma una contraseña del tipo "YUghd73j%" en base de datos se almacenaría con el siguiente valor "993e65b24451e0241617d6810849c824", como se observa, se trata de un valor que poco o nada tiene que ver con el original

2.3.8.6. Inyecciones SQL

Un ataque de este tipo da acceso a alguien a una base de datos completa sin ningún tipo de restricción, pudiendo llegar incluso a copiar y modificar los datos. El funcionamiento de este tipo de vulnerabilidades es similar al que puede ocurrir en los portales web, donde una mala limpieza de los datos de entrada puede hace que ejecuten código no deseado en la base de datos, pudiendo coger el control de la plataforma. Muchos proveedores ofrecen soluciones para este tipo de situaciones, pero para que surjan efecto es necesario realizar la actualización de la aplicación a la última versión estable disponible que exista en cada momento.

2.3.9. LabVIEW

LabVIEW es un entorno de programación gráfica (G) que utiliza íconos, terminales y cables reemplazando el texto para programar (NationalInstrument, 2014). Permite visualizar cada

aspecto de una aplicación, incluyendo configuración de hardware, datos de medidas. Esta visualización hace que sea más fácil integrar cualquier tipo de hardware, representar una lógica compleja en el diagrama, desarrollar algoritmos de análisis, reconocimiento de datos y diseñar interfaces personalizadas.

2.3.10. Inkscape

Inkscape es un software de vectores gráficos para Windows, Mac OS X y GNU/Linux. Es usado para crear una gran variedad de gráficos como ilustraciones, iconos, logos, diagramas, mapas y diseños web. Inkscape es un software libre y de código abierto, puede exportar e importar varios formatos de archivo, incluyendo SVG, AI, EPS, PDF, PS y PNG (zigzagmlt, 2018).

2.3.11. SolidWorks

SolidWorks es un software de diseño asistido por computadora (CAD 3D), ofrece la posibilidad de crear, diseñar, simular, fabricar, publicar y gestionar datos del proceso de diseño, es utilizado específicamente para modelar piezas, desarrollar ensamblajes en 3D y planos en 2D (SolidBi, s.f.).

CAPÍTULO III

3. DISEÑO DE LA INVESTIGACIÓN

3.1. Tipos de Investigación

La investigación propuesta es de tipo descriptiva y experimental, se realizó un diagnóstico para conocer las particularidades del problema y sus posibles soluciones. Es descriptiva, porque analiza la situación y detalla los posibles problemas y soluciones, mide las características y observa la configuración y los procesos que componen la solución. Es experimental porque comprueba de forma práctica cada teoría analizada con las posibles soluciones.

3.2. Métodos de Investigación

Durante el proceso de implementación del prototipo, del sistema biométrico de transmisión inalámbrica para la identificación de personas a través de su historial médico, se utilizó el método deductivo, puesto que se realizó el análisis de los equipos más apropiados con el fin de obtener los mejores resultados. Así mismo se trabajó con el método analítico porque se exploró cada bloque del sistema, finalmente se utilizó el método experimental y de campo porque se implementó el sistema.

3.3. Enfoque de la Investigación

El enfoque de investigación es cualitativo y cuantitativo, porque se utilizó datos obtenidos de manera experimental realizando un análisis de los mismos, para determinar el desempeño del sistema.

3.4. Alcance de la Investigación

3.4.1. Descriptivo

Se describe el diseño del sistema, la arquitectura básica necesaria, así como también las estrategias de comunicación y transmisión para el funcionamiento más adecuado.

3.4.2. Correlacional

Relaciona la información obtenida mediante un sensor biométrico con los datos almacenados en un servidor, de forma rápida y eficiente vía inalámbrica.

3.4.3. Explicativo:

Explica el proceso y funcionamiento del sistema biométrico inalámbrico para la identificación y reconocimiento de personas a través de sus datos médicos y personales, esto es mediante un algoritmo que detalla las operaciones lógicas a realizar.

3.5. Unidad de Análisis

3.5.1. Identificación de personas

Para llevar a efecto la identificación de personas se utiliza el procesamiento digital de imágenes para extraer los rasgos característicos de las huellas digitales y realizar la comparar entre plantillas, otra forma es emplear las consultas SQL utilizando el número de la cédula.

3.5.2. Comunicación inalámbrica

El envío de la información se realiza a través de mensajes de texto utilizando el sistema global de comunicaciones móviles o sistema GSM. Desde el equipo terminal hacia la estación base se envía la plantilla con los rasgos característicos y desde la estación central se envía los datos que corresponden a la persona identificada.

3.5.3. Procesamiento Digital de Imágenes

La interfaz gráfica captura la imagen, procesa la información mediante un tratamiento digital de imágenes, con el fin de obtener los rasgos característicos o minucias de las huellas digitales.

3.5.4. Bases de datos

La base de datos ubicada en un servidor, almacena la información de cada persona registrada, junto con su historial médico.

3.6. Diseño e Implementación

3.6.1. Comparación de las Tecnologías Biométricas

Para la comparación de las tecnologías se van a tomar diferentes criterios de calificación, los cuales son:

- Universalidad: indica que tan común es encontrar la medida biométrica.
- Singularidad: indica que tan único o diferenciable es una medida biométrica con respecto a otra.
- Permanencia: indica el tiempo de permanencia de la medida biométrica en la persona sin sufrir cambios.
- Colectividad: la facilidad para adquirir, almacenar y medir el patrón biométrico.
- Rendimiento: indica que tan preciso, rápido y robusto es el sistema en el manejo de la medida biométrica.
- Aceptabilidad: indica la aprobación de la tecnología biométrica en el público.
- Fiabilidad: indica que tan fácil es engañar al sistema biométrico.

La Tabla 1-3, muestra un resumen de las ventajas y desventajas de las principales tecnologías biométricas, se puede ver que las huellas dactilares tienen un buen balance, cabe destacar que el rendimiento es muy deseable por las empresas, que suelen utilizar una tecnología biométrica para

controlar el acceso a personas. También el desempeño general del reconocimiento del ADN y el iris es bueno y en algunos casos es superior al de las huellas dactilares, debido a la cantidad de puntos característicos que se analizan.

Tabla 1-3: Comparación entre diversas tecnologías biométricas

Sistema Biométrico	Universalidad	Singularidad	Permanencia	Colectividad	Rendimiento	Aceptabilidad	Fiabilidad
ADN	Alto	Alto	Alto	Bajo	Alto	Bajo	Bajo
Huella dactilar	Medio	Alto	Alto	Medio	Alto	Medio	Medio
Rostro	Alto	Bajo	Medio	Alto	Bajo	Alto	Alto
Retina	Alto	Alto	Medio	Bajo	Alto	Bajo	Bajo
Iris	Alto	Alto	Alto	Medio	Alto	Bajo	Bajo
Voz	Medio	Bajo	Bajo	Medio	Bajo	Alto	Alto

Fuente: (Gualberto, Sánchez , Toscano, Nakano, & Pérez, 2008)
Realizado por: Salazar Byron, 2019
Nota. El sistema más fiable es el de huellas dactilares.

El método de identificación mediante huella dactilar es uno de los más fiables que actualmente se conocen. En estos últimos años la biometría dactilar se ha acercado al público en general y casi no resulta extraño ver en algunas instalaciones la utilización de detectores de huella dactilar para el acceso de personas, incluso en los ordenadores portátiles o teléfonos incluyen detectores de huella dactilar para que un usuario previamente registrado pueda iniciar una sesión.

3.6.2. Comparación de las Tarjetas de desarrollo.

Son tarjetas electrónicas cuyo núcleo principal consta de un microcontrolador o circuito integrado programable, capaz de ejecutar las órdenes grabadas en su memoria. Está compuesto de varios bloques funcionales, puertos de entrada/salida digitales y analógicos, los cuales cumplen una tarea específica dependiendo de las instrucciones establecidas. Entre las tarjetas de desarrollo más comunes se puede encontrar Arduino y Rasberry, la Figura 1-3, muestra el Arduino Mega 2560 y la Rasberry Pi.

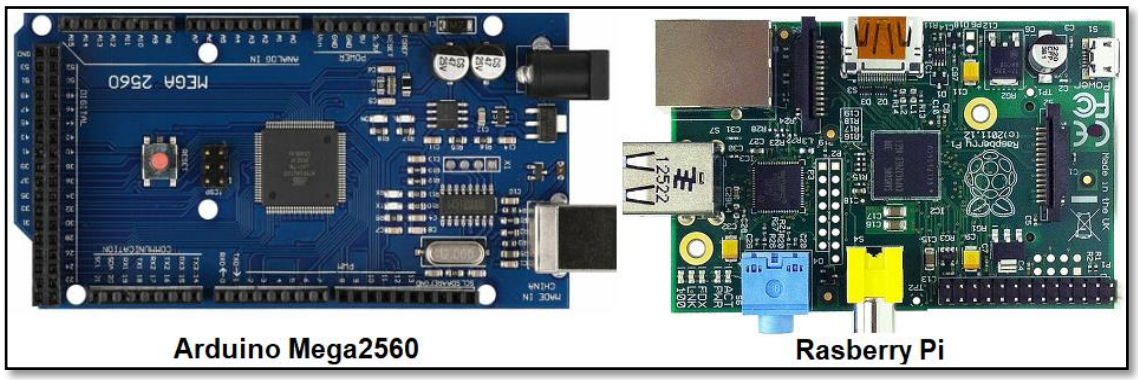


Figura 1-3. Tarjetas de Desarrollo Electrónicas.
Realizado por: Salazar Byron, 2019

La Tabla 2-3, muestra la comparación entre los módulos que considerados para construir el equipo terminal y la estación central.

Tabla 2-3: Tabla comparativa entre Arduino Y Raspberry Pi

Características	Arduino MEGA2560	Raspberry Pi
Core	ATmega 2560	4xARM Cortex-A53
Reloj principal	16MHz	1,2GHz
Tamaño de registros	8 bits	32 bits
RAM	2,5KB	1GB
Flash	256 KB	16GB, 32GB (SD)
Tiempo real	Si	No
Consumo	0,125 W	1,75 W
Precio	10 USD	55 USD
Facilidad de integración	Alta	Baja
Sistema operativo	Posible	Obligatorio
Puestos Serie	4	1

Fuente: (Crespo E. , 2017)
Realizado por: Salazar Byron, 2019
Nota. La tarjeta más adecuada para cumplir con los requerimientos establecidos es el Arduino Mega.

El módulo Raspberry Pi esta sobre dimensionado, ya que no se requiere hacer procesamiento en el equipo terminal, el Arduino consume baja potencia, es más barato, posee 4 puertos de comunicación serie y gestiona sin problemas el tráfico de información. Por esto se ha utilizado el Arduino Mega2560 para este proyecto.

3.6.3. Análisis de los Módulos Inalámbricos

3.6.3.1. Módulos Sim GSM

El módulo GSM es un terminal de comunicación inalámbrica, utiliza una tarjeta SIM (tarjeta de telefonía móvil), su función es similar a la de un teléfono móvil, y consiste básicamente en realizar llamadas y enviar mensajes de texto.

3.6.3.1.1. Arquitectura de la red GSM

La red GSM permite la interconexión con la red fija de telefonía móvil, es también conocida como la red de segunda generación o 2G. La banda de frecuencia en la que opera la red GSM es de (850/900/1800/1900) MHz y se compone de los siguientes elementos:

- **Estación móvil (MS):** es el dispositivo con el que el usuario se conecta a la red, estableciendo un enlace radio entre la estación móvil y la estación base de su área.
- **Área de Localización:** es un grupo de celdas, una celda es la unidad básica de cobertura de una estación base de radio frecuencia y cubre una determinada área.
- **Estación Base (BTS):** contiene los equipos de transmisión y recepción (antenas y amplificadores), además proporciona el enlace vía radio entre la red y las MS.

- **Controlador de estaciones base (BSC):** constituye el primer nivel de concentración de tráfico para minimizar los costes de transmisión.
- **Centro de conmutación de servicios móviles (MSC):** se encarga de realizar labores de conmutación dentro de la red y señalización básicas.
- **Registro de localización local (HLR):** es una base de datos que contiene información relativa a todos los usuarios abonados de la red móvil.
- **Registro de localización de visitantes (VLR):** es una base de datos de ubicaciones, actualiza el HLR periódicamente con la última ubicación de un MS.
- **Puerta de Enlace (SMS-GMSC):** es un nodo de unión de la red GSM con otras redes externas, gestiona los SMS he interroga al HLR para obtener información de encaminamiento para realizar llamadas.
- **Registro de identidad del Equipo (EIR):** su función es evitar que se utilicen equipos móviles no autorizados.

Los elementos descritos se observan en la Figura **2-3**, hacen posible la comunicación a través de la red GSM.

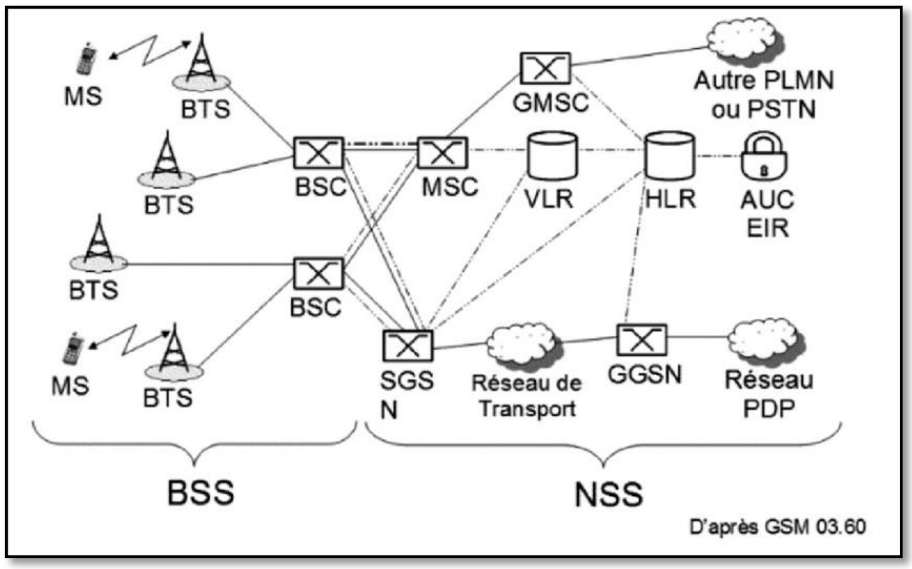


Figura 2-3. Arquitectura de la red GSM.
Fuente: (Dailly & Collet, 2018)

3.6.3.1.2. Estructura de los canales

GSM 850 que es la banda en la que opera Claro, tiene asignados los canales 128 a 251, iniciando la frecuencia de UPLINK en 824,2 MHZ. Existe una separación de 45 MHz entre los canales UPLINK y los canales DOWNLINK, por lo tanto, los canales de bajada estarán situados a partir de los 869,2 MHZ (Escalona & Paradels, 2002).

El canal 129 UPLINK, inicia en 824,4 MHz, y su equivalente DOWNLINK 45 MHz después o 869,4 MHz. Y así sucesivamente para todos los canales ARFCN de GSM 850, hasta el canal 251 que concluirá en 848,8 MHz, y su equivalente en DOWNLINK en 893,8 MHz (Escalona & Paradels, 2002).

- **Canales físicos:** GSM es un sistema multiportadora, para acceder al medio utiliza una combinación de TDMA (Time Division Multiple Access) y FDMA (Frequency Division Multiplex Access). El espacio entre portadora es de 200 KHz, permitiendo 124 canales (para el caso GSM-850) por cada enlace (ascendente y descendente) y una banda de guarda de 200 KHz en los extremos. Cada canal está dividido en 8 ranuras en el tiempo, denominadas time slots, con una duración de 0.577 ms. En la **Figura 3-3**, se observa su esquema.

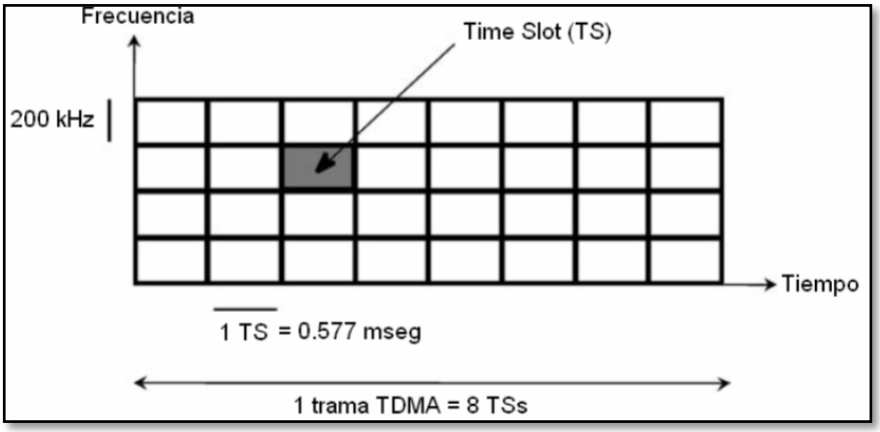


Figura 3-3. Esquema de acceso al medio en GSM.
Fuente: (Botella, 2017)

- **Canales lógicos:** El canal lógico es la información portada por los time-slot. Los canales lógicos son mapeados o multiplexados en uno o diferentes time-slots para transmitir información. Existen dos tipos de canales en GSM: los canales de tráfico (TCH – Traffic CHannels), que transportan información (voz o datos) del usuario y los canales de control (CCH – Control CHannels), que transportan señalización y sincronización entre la estación base y la estación móvil (Botella, 2017).

3.6.3.2. Módulos de comunicación XBee PRO S3B

Los módulos XBee S3B son dispositivos de radio frecuencia, diseñados para tramos de largo alcance, en zonas urbanas hasta 600 metros, en exteriores se puede tener un alcance de hasta 15 millas (24km) con antena de alta ganancia y de hasta 6 millas (9.6km) con antena dipolo (Xbee).

3.6.3.2.1. Protocolo de Comunicación ZigBee

ZigBee es un protocolo inalámbrico que adopta el estándar IEEE 802.15.4, trabaja en las bandas libres ISM (Industrial, Scientific and Medical). Posee tasas de transmisión de hasta 250kbps con

2.4 GHz, 40kbps en 915 Mhz y 20kbps en 868Mhz, se puede conectar hasta 65.534 dispositivos por red. El protocolo ZigBee permite topologías como: malla, árbol o estrella. La Figura **4-3**, muestra las topologías que se pueden configurar y el tipo de nodos que forman las redes.

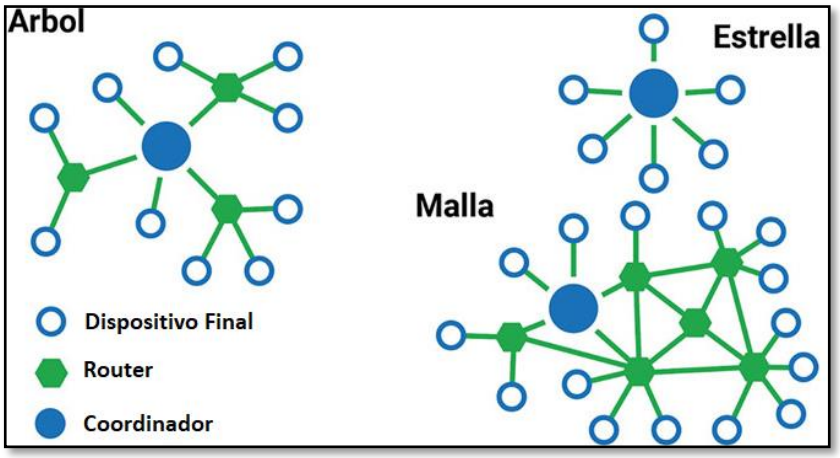


Figura 4-3. Topología de Red ZigBee.
Fuente: (LEDBOX, 2021)

Los nodos pueden asumir tres roles distintos:

- **Dispositivo final:** Son todos los dispositivos que de una forma activa o pasiva interactúan con nosotros, se comunican únicamente con el router o el coordinador y no tienen la capacidad de enrutar paquetes.
- **Coordinador:** solo puede existir uno, situado en el centro de una red en estrella o en la raíz de una red en árbol, realiza las funciones de inicio, control y enrutamiento. Ejerce un papel clave en la gestión de la seguridad de las comunicaciones actuando como Centro de Confianza, encargado de realizar la distribución de las claves de seguridad utilizadas en el cifrado de las comunicaciones.
- **Router:** Su función es la de gestionar las rutas de comunicación entre los dispositivos. Una vez dentro de una red ZigBee, el Router tiene la capacidad de retransmitir paquetes provenientes de otros Routers o de Dispositivos Finales.

3.6.3.2.2. Canales

El estándar IEEE 802.15.4 define 27 canales de comunicación diferentes por lo que el dispositivo y la capa física deben seleccionar la frecuencia y sintonizar al dispositivo dentro del canal a utilizarse. La asignación del canal se da de manera dinámica y se define a través de una combinación de números y páginas de canal. La Figura 5-3, muestra la asignación de canales.

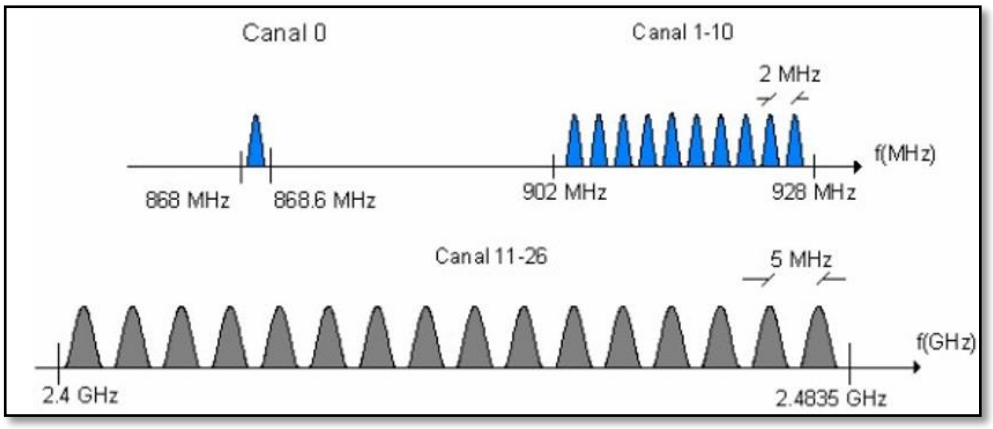


Figura 5-3. Canales de Frecuencias usadas por IEEE 802.15.4-2006.
Fuente: (Villacrés, 2009)

3.6.3.2.3. Numeración de canales para 868 MHz, 915 MHz y 2450 MHz.

- La banda de 868 MHz soporta un solo canal entre las frecuencias de 868 y 868,6 MHz.
- La banda de 915 MHz soporta 10 canales entre las frecuencias de 902.2 y 908.0 MHz, con un espaciamiento entre cada canal de 2MHz.
- La banda de 2,4 GHz soporta 16 canales entre las frecuencias de 2.4 y 2.4835 GHz, con un espaciamiento entre cada canal de 5MHz.

La frecuencia central en cada uno de los canales se calcula con las siguientes ecuaciónes (*1*), (*2*) y (*3*).

$$fc[Mhz] = 868{,}3, para\ k = 0 \quad (1)$$

$$fc[Mhz] = 906\ + 2(k-1), para\ k = 1{,}2,\ldots 10 \quad (2)$$

$$fc[Mhz] = 2405\ + 5(k-1), para\ k = 11{,}12,\ldots 26 \quad (3)$$

Dónde:

- k es el número del canal

3.6.3.3. Selección de los módulos inalámbricos

La Figura 6-3, muestra los módulos de comunicación inalámbrica analizados para seleccionar la mejor opción e implementar el sistema biométrico. Los módulos SIM utilizan la red de telefonía móvil para enviar la información, sin embargo, los módulos Xbee utilizan topologías como la de estrella o árbol, por los que se reduce la distancia de transmisión de datos, por tal motivo se ha escogido los módulos SIM GSM.

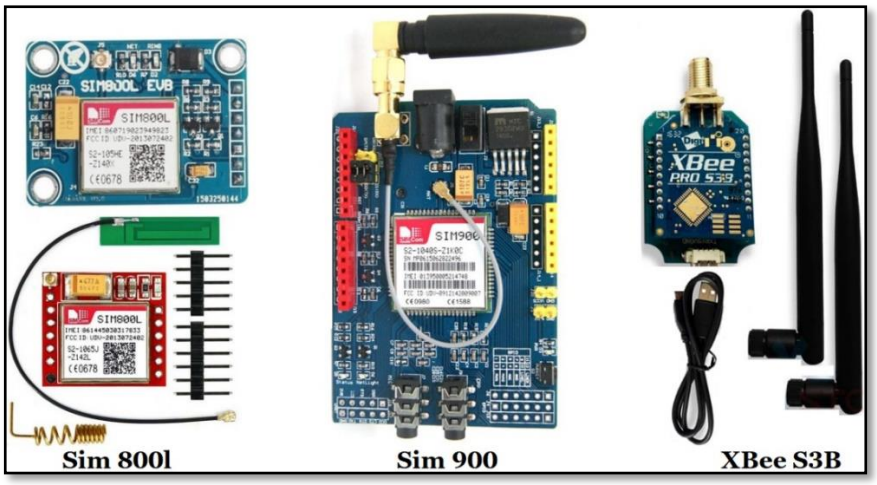


Figura 6-3. Módulos de Comunicación Inalámbrica.
Realizado por: Salazar Byron, 2019.

Los módulos inalámbricos Sim800L y Sim900 poseen diferentes características como se ve en la Tabla 3-3. Por cuestiones de consumo, tamaño y adaptabilidad, el módulo más eficiente es el Sim800L, ya que simplifica la transmisión de la información, consume menos corriente y voltaje; esto incrementa el tiempo de autonomía del equipo terminal.

Tabla 3-3: Tabla comparativa entre Módulos Sim

Características	SIM800L	SIM900
Voltaje de Operación:	3.4V ~ 4.4V	5 V ~ 12 V
Corriente (máximo):	500 mA	2A
Corriente (modo sleep):	0,7 mA	1,5 mA
Interfaz:	Serial UART	Serial UART
Velocidades de transmisión:	1200bps hasta 115200bps	1200bps hasta 115200bps
Frecuencia de operación:	850/900/1800/1900MHz	850/900/1800/1900MHz
Envía y recibe mensajes SMS:	Modo texto y PDU.	Modo texto y PDU.
Envía y recibe:	Datos GPRS (TCP/IP, HTTP)	Datos GPRS (TCP/IP, HTTP)
Controlado por:	Comandos AT	Comandos AT
Velocidad máxima de tx:	85.6 Kbps	85,6 Kbps hasta 42,8 Kbps
Chip:	Protocolo TCP/IP	Protocolo TCP/IP
Tamaño de la SIM:	Micro SIM	Micro SIM

Fuente: (Dailly & Collet, 2018)
Realizado por: Salazar Byron, 2019
Nota. La tarjeta más adecuada para cumplir con los requerimientos establecidos es el Sim800L

3.6.4. Seguridad de base de datos

La seguridad es un aspecto importante pare evitar que la información sea robada o modificada por los hackers, por tal motivo a continuación se describe los aspectos tomados en consideración para cuidar los datos.

3.6.4.1. Contraseña Segura

Una contraseña segura debe tener las siguientes características:

- **Larga:** Cuanto más larga es una contraseña, más segura es, debe tener al menos, 12 caracteres.
- **Aleatoria:** Se emplea una combinación de letras, números, mayúsculas y símbolos para formar una cadena impredecible de caracteres que no se asemejen a nombres.
- **Única:** Debe ser única para cada cuenta a fin de reducir su vulnerabilidad en caso de hackeo.

3.6.4.2. Prevención de amenazas

- Se previene ataques de ransomware aplicando antivirus, en este caso el "Malwarebytes". El ransomware, es un tipo de malware que impide a los usuarios acceder a su sistema o archivos personales, se exige un pago de rescate para poder acceder de nuevo a ellos.
- Se realiza copias de seguridad de los datos en la nube y en un dispositivo local.
- Se realiza actualizaciones constantes del sistema operativo y la base de datos con las últimas versiones ya que en ellas se solucionan problemas de seguridad detectados, por lo que se pone más barreras a los posibles atacantes.

3.6.4.3. Inyecciones SQL

Es un método de infiltración de código intruso. Estas vulnerabilidades afectan a los sitios web que utilicen bases de datos SQL como lo son MariaDB, MySQL, Oracle, SQL Server, entre otros. Según el Proyecto de seguridad de aplicaciones web abiertas (OWASP), las inyecciones SQL están catalogadas como la amenaza número uno para la seguridad de las aplicaciones.

Se produce una inyección SQL cuando, se inserta o "inyecta" código SQL invasor dentro del código SQL programado, a fin de alterar el funcionamiento normal del programa y lograr así que se ejecute la porción de código "invasor" incrustado en la base de datos, este tipo de intrusión es de carácter malicioso, dañino o espía. Un programa elaborado con descuido resultar ser vulnerable, y la seguridad del sistema (base de datos) puede quedar eventualmente comprometida.

La vulnerabilidad se efectúa cuando se "arma descuidadamente" una sentencia SQL en tiempo de ejecución, o bien durante la fase de desarrollo, cuando el programador crea la sentencia SQL a ejecutar en forma desprotegida. Al ejecutarse la consulta en la base de datos, el código SQL inyectado también se ejecutará y puede hacer cosas, como insertar registros, modificar o eliminar datos, autorizar accesos e, incluso, ejecutar otro tipo de código malicioso en el computador.

Por ejemplo, en el sistema principal en la sección consulta de usuarios se puede realizar una búsqueda mediante el parámetro "Cedula" de las personas registradas, una inyección SQL se provoca de la siguiente forma:

consulta: = “SELECT * FROM usuarios WHERE cedula = ' ”+ Cedula + “ ';”

Si el operador escribe un número de cédula, nada anormal sucede y la aplicación genera una sentencia SQL similar a la siguiente, que es correcta:

SELECT * FROM usuarios WHERE cedula = '1803725066';

Pero si un operador malintencionado inyecta código SQL como el siguiente:

1803725066'; DROP TABLE usuarios; SELECT * FROM datos WHERE nombre LIKE '%

Se genera la siguiente consulta SQL, (el color verde es lo que pretende el programador, el azul es el dato, y el rojo, el código SQL inyectado):

SELECT * FROM usuarios WHERE cedula = '1803725066';

DROP TABLE usuarios;

SELECT * FROM datos WHERE nombre LIKE '%';

En la base de datos se ejecuta la consulta en el orden dado, se selecciona el registro con el número de cédula '1803725066', se borra la tabla 'usuarios' y finalmente se seleccionaría toda la tabla "datos", que no debería estar disponible para los usuarios comunes. En resumen, cualquier dato de la base de datos puede quedar disponible para ser leído o modificado por un usuario malintencionado.

3.6.4.3.1. Prevención inyecciones SQL

Para corregir este problema, se desinfecta la entrada del usuario, básicamente, lo que se hace es eliminar todos los caracteres especiales de una cadena para que pierdan su significado cuando los utilice la base de datos o limitando el número de caracteres que se puede ingresar para el número de cédula en la entrada de texto. La Figura 7-3, muestra la manera de evitar inyecciones SQL cuando se realiza una consulta.

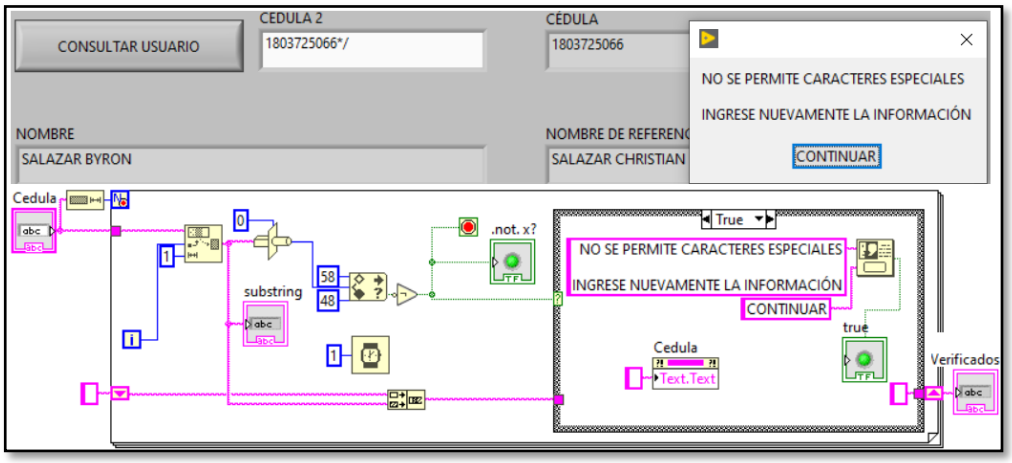


Figura 7-3. Prevención de inyecciones SQL.
Realizado por: Salazar Byron, 2019.

3.6.5. Requerimientos alcanzados

El sistema biométrico de transmisión inalámbrica es capaz de identificar a una persona por la huella digital o su número de cédula siempre y cuando haya sido previamente registrada, cuenta con una base de datos de prueba en la que se almacena la información personal, médica, plantillas de minucias e imágenes de las huellas digitales. El sistema biométrico consta de los siguientes bloques:

- Equipo Terminal
- Estación Central

3.6.6. Equipo terminal

El equipo terminal es un sistema de transmisión inalámbrica, permite identificar personas mediante la huella digital o el número de cédula. Fue diseñado y probado en las unidades móviles en el Hospital General Docente Ambato, Centros de salud número 1 y 2, Centro de Salud de Totoras, tiene una autonomía de 5 horas gracia a la batería incluida, opera en cualquier zona siempre y cuando exista cobertura. El diagrama de flujo de la Figura 8-3, explica el funcionamiento del equipo terminal.

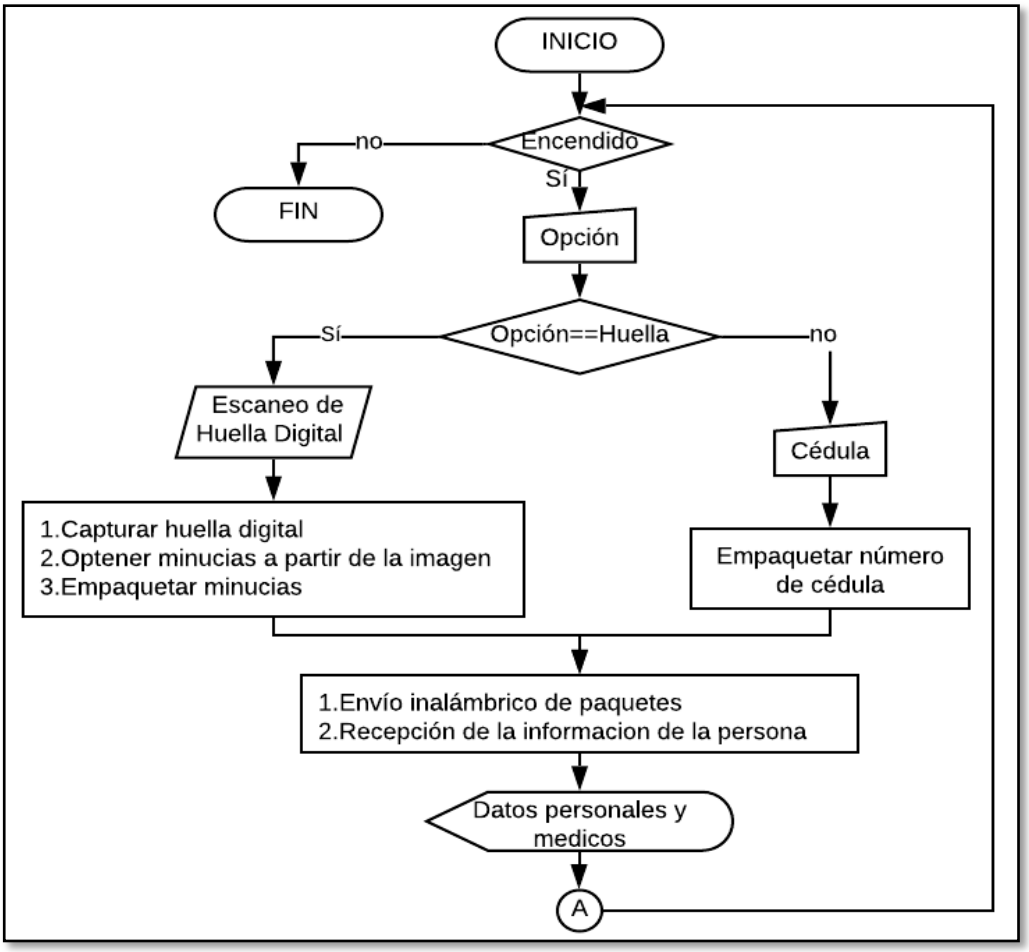


Figura 8-3. Diagrama de flujo del Equipo Terminal.
Realizado por: Salazar Byron, 2019

En la pantalla principal del Equipo Terminal, están dos botones, uno activa el sensor biométrico para digitalizar la huella, obtener la plantilla de minucias a partir de la imagen y descarga la información; el segundo botón, ejecuta una pantalla donde se puede ingresar los 10 dígitos. Finalmente, estos datos son empaquetados y enviados hacia la estación central.

La estación central recibe la información, la procesa y compara con los datos previamente almacenados. Si identifica a alguien, reenvía sus datos al Equipo Terminal, el cual proyecta la información en pantalla, esto permite darse cuenta de quien se trata y qué medidas tomar en base a su historial médico. A continuación, se describe el diseño y los elementos que componen el equipo terminal.

3.6.6.1. Placa Principal

La placa principal sirve para integrar todos los módulos del equipo terminal, permite alimentar eléctricamente cada etapa del sistema además de la comunicación entre el Arduino, la pantalla, el sensor biométrico y el módulo de comunicación inalámbrico. La placa fue diseñada en Proteus.

El Arduino Mega 2560 o tarjeta de control se comunica con todos los dispositivos a través de sus puertos serie, la forma de comunicarse es mediante tramas de datos, estas tramas se detallan más adelante. La Figura 9-3, muestra la tarjeta principal, el ANEXO 1 contiene las pistas de las placas del equipo terminal y la estación central.

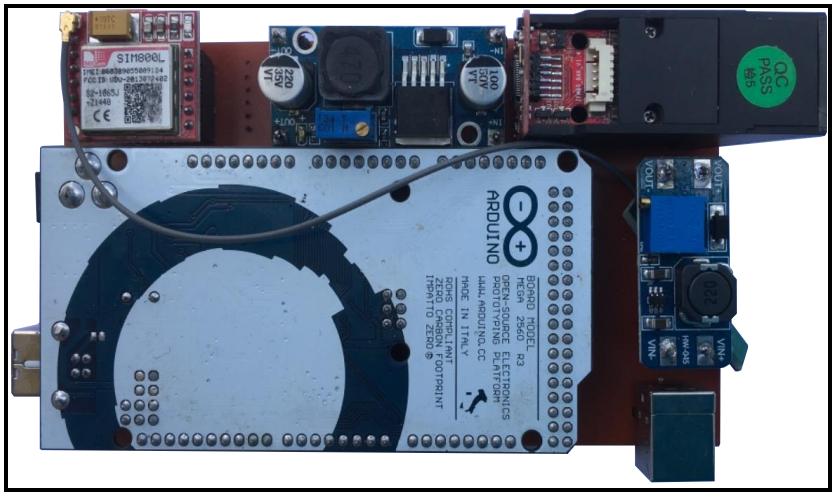


Figura 9-3. Tarjeta Principal.
Realizado por: Salazar Byron, 2019

3.6.6.2. Etapa de alimentación

La etapa de alimentación consta de una batería Samsun de 3,85 Voltio y 2800 miliamperios, un elevador de voltaje XL6009 que puede variar de 7 a 32 voltios y un reductor de voltaje LM2596 que varía de 1,23 a 27 voltios. La Figura 10-3, muestra la etapa de alimentación, conjuntamente con los diagramas de cada módulo.

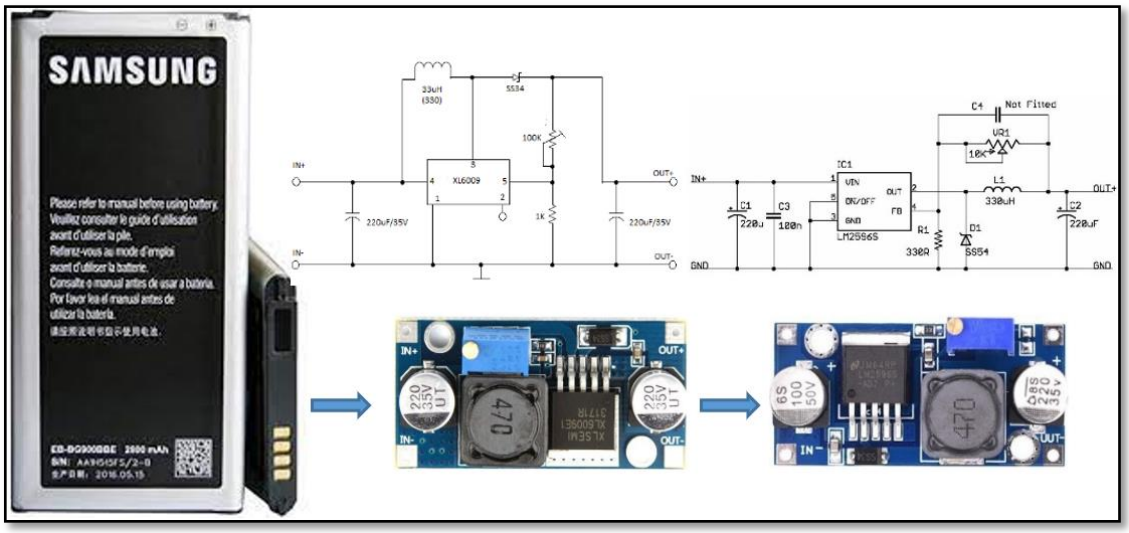


Figura 10-3. Etapa de alimentación (Batería, Elevador y Reductor de Voltaje).
Realizado por: Salazar Byron, 2019

La batería genera de 3,85 a 4,3 Voltios, pero este valor no es suficiente para el circuito electrónico; por lo que se coloca un elevador para regular y mantener constante 6 voltios así se descargue la batería, este módulo alimenta la tarjeta de desarrollo y la pantalla nextion. Finalmente, a la salida se conecta un reductor de voltaje que mantenga una tensión de 4,1 Voltios para proteger al sensor biométrico y al módulo SIM800L.

3.6.6.3. Interfaz gráfica o (HMI)

HMI son las siglas en ingles de Human Machine Interfaz o Interfaz hombre máquina, permite interactuar y controlar el sistema biométrico para llevar a efecto el reconocimiento de personas. El control es a través de tramas que contienen los datos o las instrucciones que debe ejecutar el Arduino mega.

3.6.6.3.1. Diseño gráfico de la interfaz

La interfaz fue diseñada en Inkscape un editor de gráficos de código abierto. Las imágenes permiten que la interfaz sea intuitiva y fácil de utilizar. Esta interfaz cuenta con cuatro páginas, dos para control he ingreso de información y dos para mostrar los datos consultados. A continuación, se describen la estructura de cada página.

La figura de la página principal cuenta con tres espacios en negro que representan el lugar para colocar botones de control, la primera imagen con teclado muestra el color de los botones sin presionar y la segunda el color de los botones presionados, la cuarta figura sirve para proyectar la información, cuenta con tres botones para navegar entre pantallas. En la Figura 11-3, se observa las imágenes diseñadas para la interfaz gráfica.

Figura 11-3. Imágenes para las páginas de la Interfaz gráfica.
Realizado por: Salazar Byron, 2019.

3.6.6.3.2. Programación y configuración de la interfaz gráfica

Para programar la interfaz gráfica se utilizó el programa Nextion Editor. Se crea un nuevo proyecto, se asigna un nombre y se selecciona el modelo de la pantalla, "NX4827T043". Finalmente, se escoge la orientación del diseño a noventa grados y se presiona Ok para comenzar. La Figura 12-3, muestra el proceso de creación del proyecto.

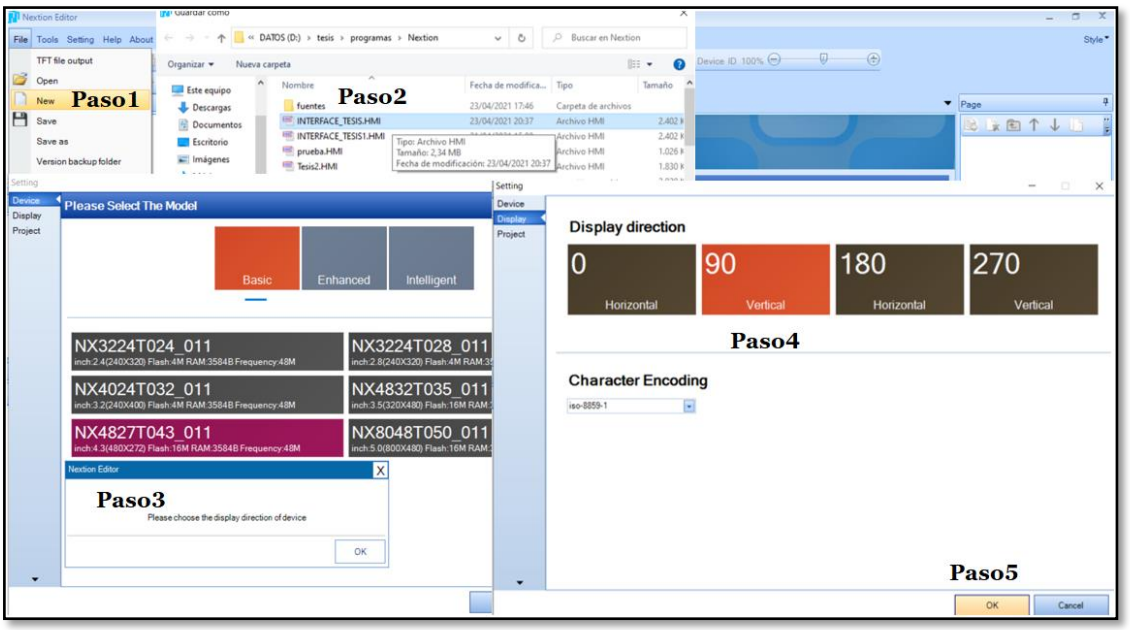


Figura 12-3. Creación de proyecto de la Interfaz Gráfica.
Realizado por: Salazar Byron, 2019.

En la parte inferior izquierda del programa Nextion editor en la ventana Picture se presiona en la imagen del signo más y se añade las imágenes diseñadas en Inkscape, en la parte superior derecha, en la ventana “Page” se añade tres nuevas páginas, se selecciona la página uno y se carga la imagen que corresponde a la página principal, se selecciona la página dos y se carga la imagen con teclado, en las páginas tres y cuatro se carga la misma imagen para mostrar los datos y se procede a programar cada página.

En la parte superior de cada página esta una barra que contiene dos imágenes de control, una muestra el estado de la batería y la otra muestra el nivel de recepción de señal del módulo inalámbrico, el botón “ESPOCH” de cada página sirve para colocar la pantalla en estado de reposo para ahorrar energía. Cada herramienta posee atributos que permiten identificar los elementos utilizados. Las abreviaturas utilizadas para representar los controles son (btn) para Botones, (txt) para texto y (img) para imágenes.

La pantalla principal que se muestra en la Figura 13-3, es con la que arranca el sistema, permite escoger la forma de operación para identificar a una persona ya sea por su huella digital o su número de cédula. Consta de cuatro botones y un control de texto para mostrar el estado de la operación.

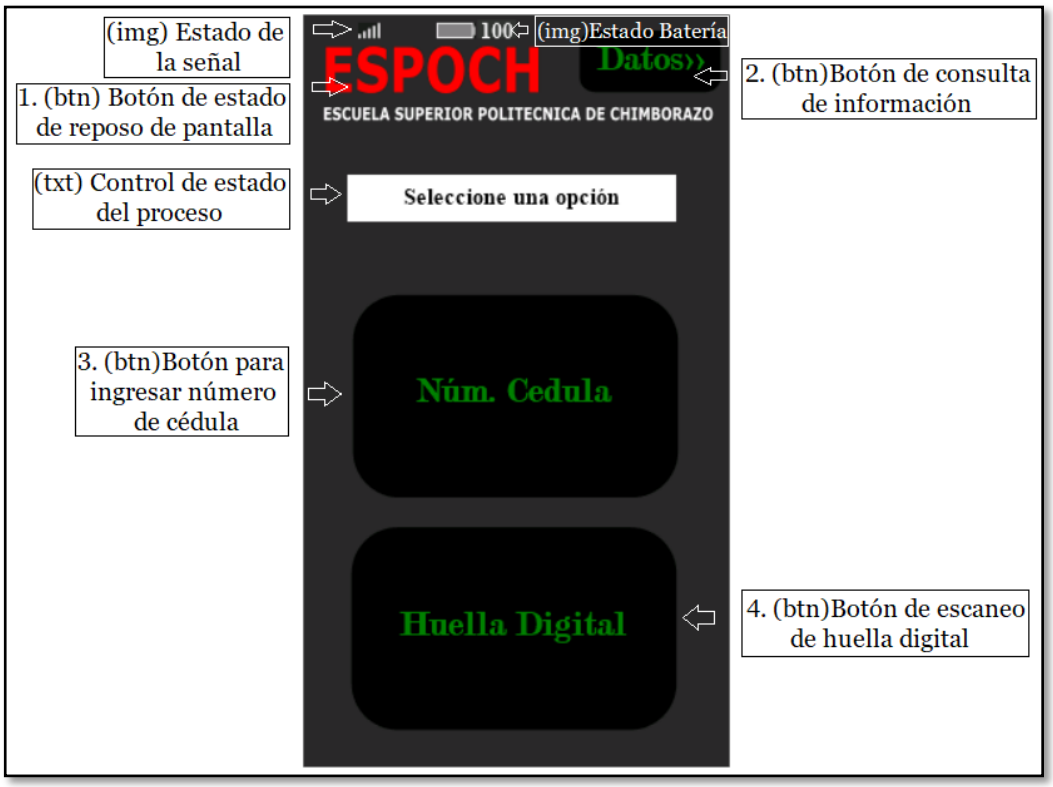


Figura 13-3. Pantalla principal de control del Sistema Biométrico.
Realizado por: Salazar Byron, 2019.

Cada control ya sea botón, texto o imagen poseen atributos, estos atributos permiten identificar a las herramientas utilizadas en la interfaz para programarlos y controlarlos ya sea desde la propia pantalla o desde el Arduino a través del puerto de comunicación serie. La Tabla 4-3, muestra los parámetros que identifican cada elemento.

Tabla 4-3: Identificadores de los controles de la página principal.

Controles	Identificador	Objeto (Objname)	Descripción
Datos (btn)	3	b2	Datos>>
Espoch (btn)	8	b3	----
Núm. Cédula (btn)	1	b0	Núm. Cédula
Huella Digital (btn)	2	b1	Huella Digita
Estatus (txt)	4	tpp: texto pantalla principal	Seleccione una Opción
Nivel de Señal (img)	5	ps1: señal placa Sim	
Nivel batería (img)	7	n0	

Realizado por: Salazar Byron, 2019
Nota. Botón (btn), control de texto (txt), control de imagen (img).

El botón número uno "ESPOCH" coloca la pantalla en estado de reposo para ahorrar energía, para habilitar nuevamente la pantalla se toca en cualquier lugar de la misma dos veces. El botón número dos traslada a otra página donde se observa la información consultada. El botón tres redirecciona a otra página donde se ingresa los números de cédula. Finalmente, el botón cuatro activa el sensor biométrico para escanear la huella digital. La Tabla 5-3, muestra el código de programación de cada botón.

Tabla 5-3: Código de los botones de la página principal.

Control	Evento	Código	Descripción
"ESPOCH"	Touch Release	thsp=3	Pone la pantalla en modo reposo en 3 segundos
"Datos"	Touch Release	page page3	Muestra la página 3 de datos
		printh EF 01 03 00 00 00 00 00 00 00 00 00 00 FF FF FF	
"Num. Cedula"	Touch Release	page page2	Envía a la página 2 para ingresar datos de la cédula
"Huella Digital"	Touch Release	printh EF 01 01 00 00 00 00 00 00 00 00 00 00 FF FF FF	
		tpp.txt="Coloque el dedo en el Sensor Biométrico"	
Pantalla	Preinitialize	baud=9600	Velocidad de transmisión de datos
		thup=1	Activa la pantalla al tocar
	Touch press	thsp=0	Deshabilita suspensión de pantalla

Realizado por: Salazar Byron, 2019
Nota. Los Eventos son las acciones que realiza un elemento.

La página dos, cuenta con un control de texto para mostrar los números ingresados y otro para indicar el estado de la operación, un teclado numérico (0-9) para ingresar los números de cédula, una tecla de borrado, una tecla aceptar para enviar la información, una tecla de datos para mostrar la información consultada y finalmente una tecla en forma de casa que redirecciona a la página principal. En la Figura 14-3, se observa la página utilizada para ingresar los números de cédula.

Figura 14-3. Página de ingreso de números de cédula.
Realizado por: Salazar Byron, 2019

Al igual que en la página principal, cada control de texto, imagen o botón, posee atributos, estos atributos permiten identificar a las herramientas utilizadas en cada página para programarlos y controlarlos ya sea desde la propia Pantalla Nextion o desde el Arduino a través del puerto de comunicación serie. La Tabla 6-3, muestra los parámetros que identifican cada elemento de la segunda página utilizada para el ingreso de números de cédula.

Tabla 6-3: Identificadores de los controles de la segunda página.

Controles	Identificador	Objeto (Objname)	Descripción
(btn) 0,1,2,3,4,5,6,7,8,9	2,3,4,5,6,7,8,9,10,11	b0,b1,b2,b3,b4,b5,b6,b7,b8,b9	0,1,2,3,4,5,6,7,8,9
(btn) Borrar	12	b12	Borrar
(btn) Aceptar	13	b13	Aceptar
(txt)Dígitos cédula	1	t1	Dígitos
(txt)Estatus operación	15	tps	

Realizado por: Salazar Byron, 2019
Nota. tps=texto página secundaria.

Al presionar los botones, estos cambian de color como en la Figura 14-3, en los números 3, 6 y 9, en caso de equivocaciones el botón "borrar" permite corregir errores, el botón "ACEPTAR" envía los números al Arduino mediante una trama, si son 10 dígitos se empaqueta la información y se reenvía a la estación base para identificar a la persona y en caso de ser menos de 10, el

Arduino emite el mensaje "¡La cedula debe contener 10 Dígitos!", La Tabla 7-3, muestra el código de programación de cada botón.

Tabla 7-3: Código de los controles de la segunda página.

Control	Evento	Código	Descripción
“0”	Touch Release	t1.txt=t1.txt+"0"	Adiciona el cero a t1.txt
“1”	Touch Release	t1.txt=t1.txt+"1"	Adiciona el uno a t1.txt
“2”	Touch Release	t1.txt=t1.txt+"2"	Adiciona el dos a t1.txt
“3”	Touch Release	t1.txt=t1.txt+"3"	Adiciona el tres a t1.txt
“4”	Touch Release	t1.txt=t1.txt+"4"	Adiciona el cuatro a t1.txt
“5”	Touch Release	t1.txt=t1.txt+"5"	Adiciona el cinco a t1.txt
“6”	Touch Release	t1.txt=t1.txt+"6"	Adiciona el seis a t1.txt
“7”	Touch Release	t1.txt=t1.txt+"7"	Adiciona el siete a t1.txt
“8”	Touch Release	t1.txt=t1.txt+"8"	Adiciona el ocho a t1.txt
“9”	Touch Release	t1.txt=t1.txt+"9"	Adiciona el nueve a t1.txt
“Borrar”	Touch Release	t1.txt=t1.txt-1	Resta un digito a t1.txt o lo borra
“Aceptar”	Touch Release	printh EF 01 02 //print t1.txt //printh ff ff ff	

Realizado por: Salazar Byron, 2019

Nota. Los controles “Datos”, “ESPOCH”, ya fueron descritos en la página principal.

La página tres posee seis controles de texto para mostrar la información, con seis etiquetas que indican el tipo de información consultada, en la parte inferior hay tres botones para navegar entre pantallas. La página cuatro posee controles de texto con sus respectivas etiquetas para mostrar el nombre de los datos consultados, en la parte inferior hay tres botones para navegar. La Figura 15-3, muestra las páginas tres y cuatro.

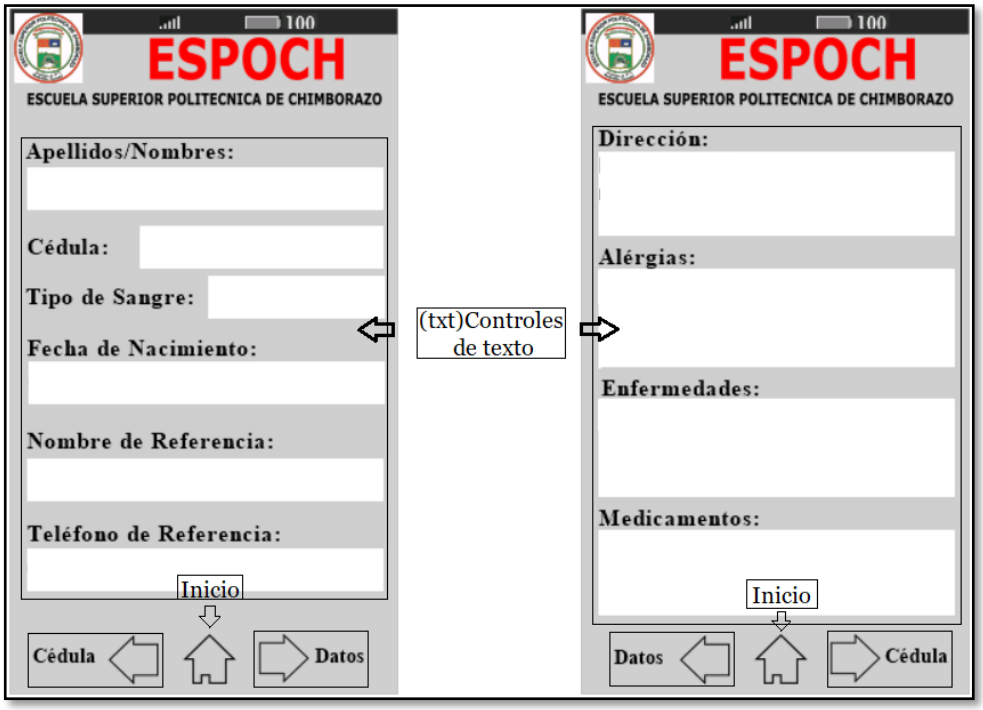


Figura 15-3. Página tres y cuatro de visualización de datos.

Realizado por: Salazar Byron, 2019.

Todos los elementos poseen atributos que identifican a las herramientas utilizadas en cada página para programarlos y controlarlos desde la Pantalla o el Arduino a través del puerto serie. La Tabla 8-3, muestra los parámetros que identifican cada elemento de la tercera y cuarta página.

Tabla 8-3: Identificadores de los controles de las páginas de visualización.

Controles	Identificador	Objeto (Objname)	Descripción
(btn)Cédula, Casa (inicio), Datos	1, 2, 4	b0, b1, b2	Cédula, Casa (inicio), Datos
(txt)Apellidos/Nombres	6	tn	
(txt) Cédula	8	tc	
(txt)Tipo de Sangre	10	tts	
(txt)Fecha de Nacimiento	12	tfn	
(txt) Nombre de referencia	14	tnr	
(txt)Teléfono de referencia	16	ttr	
(btn)Datos, Casa (inicio), Cédula	1, 3, 4	b0, b1, b2	Datos, Casa (inicio), Cédula
(txt)Dirección	6	td	
(txt)Alergias	8	ta	
(txt) Enfermedades	10	te	
(txt)Medicamentos	12	tm	

Realizado por: Salazar Byron, 2019

Nota. tn=(txt)nombre, tc=(txt)cédula, tts=(txt)tipo sangre, tfn=(txt)teléfono, tnr=(txt)nombre de referencia, ttr=(txt)teléfono de referencia, td=(txt)dirección, ta=(txt)alergias, te=(txt)enfermedades tm=(txt)medicamentos

La página tres muestra los datos personales, el botón inferior con la flecha hacia la izquierda redirecciona a la pantalla con teclado, el botón inferior con la flecha hacia la derecha redirecciona a la página cuatro, la cual muestra los datos médicos, el botón inferior con la flecha hacia la izquierda redirecciona a la pantalla tres y el botón con la flecha hacia la derecha abre la página dos. La Tabla 9-3, muestra el código de programación de cada control.

Tabla 9-3: Código de los controles de la tercera y cuarta página

Control	Evento	Código	Descripción
Cédula	Touch Release	page page2	Abre la página para ingresar números
Inicio	Touch Release	page page1	Redirecciona a la página principal
Datos	Touch Release	page page4 page page3	Si está en la página 3 abre la página 4, si está en la página 4 abre la página 3 para consultar datos.
		printh EF 01 03 00 00 00 00 00 00 00 00 00 FF FF F	

Realizado por: Salazar Byron, 2019

Nota. Las páginas tres y cuatro poseen los mismos botones por eso se describen una sola vez.

3.6.6.3.3. Implementación de la interfaz en la Pantalla Nextion

Una vez diseñada la interfaz gráfica, se compila el programa para comprobar que no existan errores y verificar su funcionamiento. Se generan dos archivos uno es de extensión *.HMI y otro de tipo *.tft, el nombre del proyecto es INTERFAZ_TESIS.HMI. El archivo de extensión "*.tft" se copia en una tarjeta de memoria microSD, en este caso se utilizó una tarjeta KINGSTON de 32 Giga bytes.

Una vez copiado el archivo, se inserta la memoria en la ranura de la pantalla “Nextion NX4827T043” al estar desconectada, para cargar el archivo en la memoria interna de la pantalla se conecta los terminales de alimentación a GND y VCC de 3,3 a 5 voltio, al finalizar la carga del archivo aparecerá un mensaje indicando carga exitosa 100%, para finalizar se debe desconectar la alimentación y retirar la memoria.

Una vez cargada la interfaz, se conecta la pantalla con el Arduino. La comunicación entre estos módulos es a través del puerto serie, la pantalla Nextion tiene cuatro terminales dos para la alimentación (GND y VCC) y dos para la transmisión y recepción (TX y RX). Los terminales TX y RX se conectan al puerto Serial 1 del Arduino, y los terminales de alimentación se conectan a GND y a VCC. La Figura 16-3, muestra la ranura de la tarjeta de memoria y la conexión entre la pantalla y el Arduino.

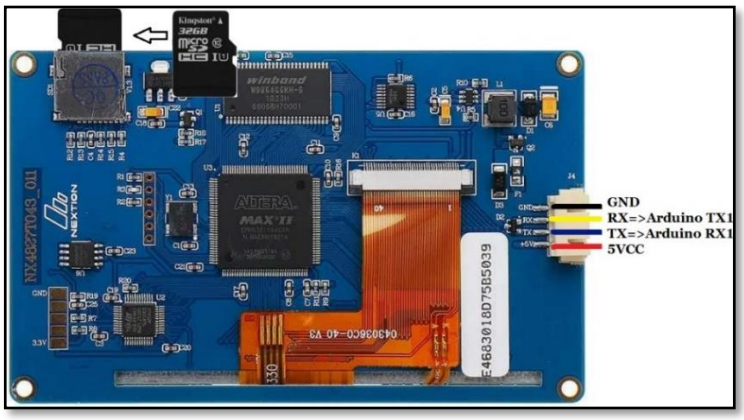


Figura 16-3. Pines de conexión hacia el Arduino.
Realizado por: Salazar Byron, 2019.

Vale la pena indicar que, para que exista comunicación entre la pantalla y el Arduino, el terminal de transmisión de la pantalla debe estar conectado al pin de recepción del Arduino (RX1=Pin 19) y el terminal de recepción de la pantalla debe estar conectado al pin de transmisión del Arduino (TX1=18) ya que, si el Arduino transmite, la pantalla recibe y cuando la pantalla transmita, el Arduino recibe.

3.6.6.4. Implementación Arduino y Pantalla Nextion

El Arduino, es el cerebro del sistema biométrico, interconecta a todos los dispositivos a través de los puertos serie mediante el uso de tramas de datos. La pantalla Nextion se conecta con el Arduino a una velocidad de 9600 baudios. La Figura 17-3, muestra la conexión entre el Arduino y la pantalla nextion.

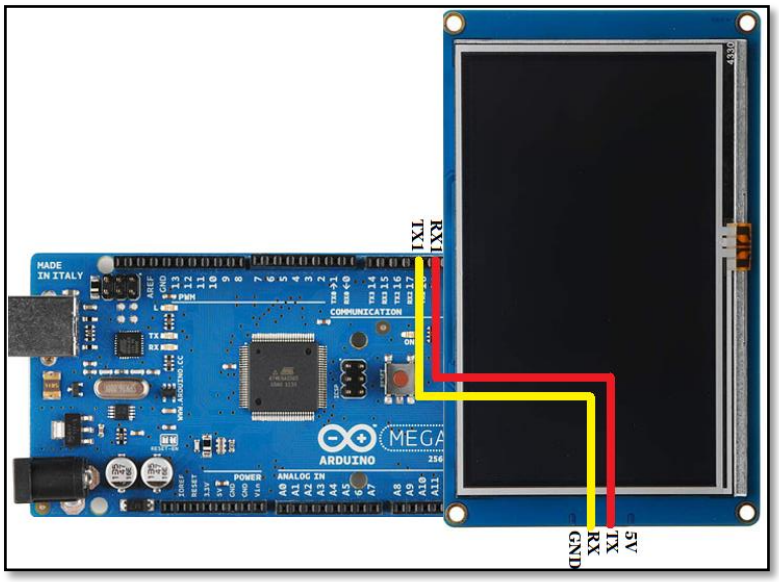


Figura 17-3. Conexión entre Arduino Pantalla Nextion.
Realizado por: Salazar Byron, 2019

3.6.6.4.1. Proyección de datos en pantalla

Una de las funciones del equipo terminal es mostrar la información consultada desde la estación central. Cuando el equipo terminal recibe la información de alguna persona, el Arduino procesa la trama y escribe de forma inmediata en los controles de texto de la pantalla tal como se ve en la Figura 18-3. Cada vez que se navega entre páginas, se borra la información de los controles, para eso existe el botón "Datos".

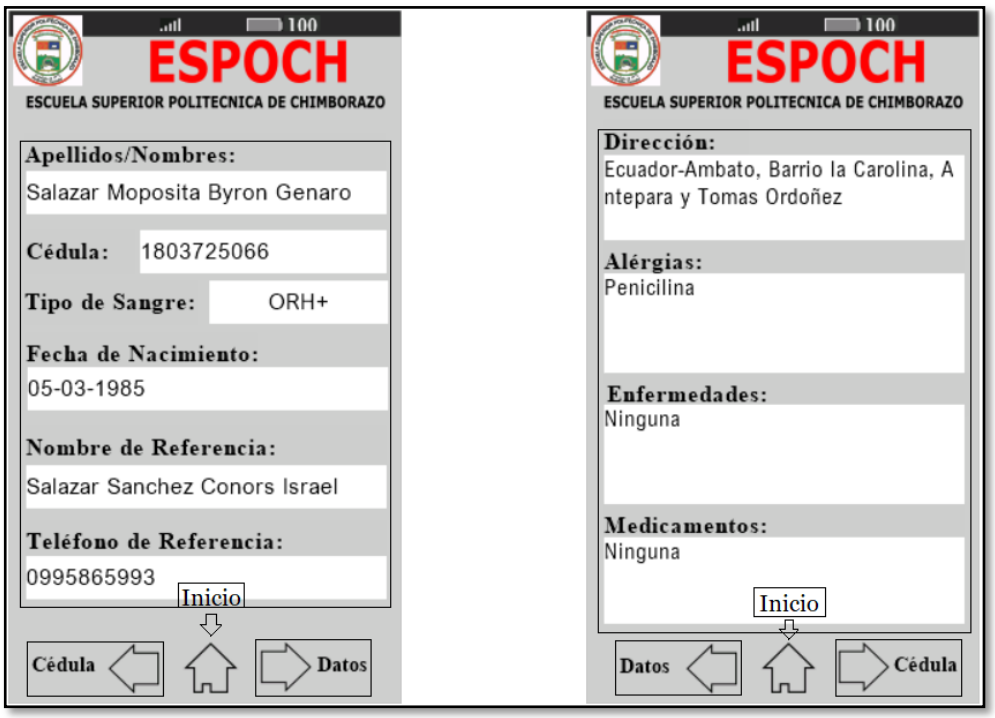


Figura 18-3. Interfaz gráfica con la información de identificación.
Realizado por: Salazar Byron, 2019.

El botón “Datos” genera una trama, la cual ordena al Arduino que escriba en los controles de texto la información consultada desde la estación central, los tres botones inferiores de las páginas tres y cuatro permiten navegar en la interfaz del sensor biométrico. La Tabla 10-3, muestra la estructura de la trama que indican al Arduino que vuelva a escriba en los controles de texto la información.

Tabla 10-3: Trama para escribir los datos consultados en los controles txt.

	2 bytes	**1 byte**	**10 bytes**	**3 bytes**
Nombre	Cabecera	Instrucción	Complemento de trama	Fin
Datos	EF 01	03	00 00 00 00 00 00 00 00 00 00	FFFFFF

Realizado por: Salazar Byron, 2019
Nota. Instrucción (03), indica al Arduino que escriba en los controles de texto la información consultada desde la estación base.

Para utilizar los elementos de la pantalla o escribir sobre los controles de texto, lo primero es incluir la librería “Nextion.h” e instanciar los objetos. Se debe crear un objeto por cada elemento que se quiere controlar desde el Arduino Mega. La Tabla 11-3, indica los objetos instanciados para utilizar los elementos de la pantalla nextion.

Tabla 11-3: Objetos instanciados para controlar desde el Arduino.

Controles de texto	**Instancia de objetos**
Estatus del proceso realizado	NexText ObjPantallaPrincipal = NexText(0, 4, "tpp");
Porcentaje de la batería	NexNumber ObjBateria = NexNumber(0, 7, "n0");
Número de cédula	NexText ObjPantallaCedula = NexText(1, 15, "tpc");
Apellidos/Nombre	NexText ObjNombre = NexText(2,6,"tn");
Cédula	NexText ObjCedula = NexText(2,8,"tc");
Tipo de Sangre	NexText ObjTipodeSangre = NexText(2,10,"tts");
Fecha de nacimiento	NexText ObjFechadeNacimiento = NexText(2,12,"tfn");
Nombre de Referencia	NexText ObjNombredeReferencia = NexText(2,14,"tnr");
Teléfono de referencia	NexText ObjTelefonodeReferencia = NexText(2,16,"ttr");
Dirección	NexText ObjDireccion = NexText(3,6,"td");
Alergias	NexText ObjAlergias = NexText(3,8,"ta");
Enfermedades	NexText ObjEnfermedades = NexText(3,10,"te");
Medicamentos	NexText ObjMedicamentos = NexText(3,12,"tm");

Realizado por: Salazar Byron, 2019
Nota. Instanciar es crear un objeto o un recurso para controlar los elementos de la pantalla.

Los parámetros utilizados para la instanciación de objetos son:

1. El primer parámetro indica el número de página en la que se encuentra el control.
2. El segundo paramero indica el valor del identificador (id) del elemento.
3. El tercer parámetro es el nombre del objeto de los controles de texto de la pantalla.

Según pruebas realizadas basta que se especifique el nombre del objeto para poder manipularlo desde el Arduino, no importa los valores del primero y segundo parámetro.

3.6.6.4.2. Control de la Interfaz Gráfica

Existen dos formar de operación para llevar a cabo el reconocimiento de personas, uno es a través de la cédula y otro el mediante procesamiento de imágenes. Cada método posee su propia trama para especificar al Arduino cuál es el procedimiento a realizar. Cuando se ejecuta algún proceso, el Arduino escribe en los controles de texto de notificación de la pantalla Nextion lo que realizó.

Para identificar a una persona mediante el número de cédula, a través de la interfaz gráfica se introduce los diez dígitos, una vez ingresados, se presiona el botón "Aceptar" y la pantalla envía la trama de datos con los números de la cédula. La Tabla 12-3, muestra la estructura de la trama.

Tabla 12-3: Trama de datos del botón "ACEPTAR" con los números de la cédula.

2 bytes	1 byte	1 byte	10 bytes	3 bytes
Cabecera	Instrucción	tamaño	Datos Hexadecimal (0-9)	Fin
EF 01	02	0A	HEX (31 38 30 33 37 32 35 30 36 36) DEC (1 8 0 3 7 2 5 0 6 6)	FFFFFF

Realizado por: Salazar Byron, 2019

Nota. Instrucción (02), indica al Arduino que la trama está compuesta por los dígitos de la cédula.

La trama está en formato hexadecimal "EF 01 02 31 38 30 33 37 32 35 30 36 36 FF FF FF". Una vez que el Arduino recibe la información, verifica que el número de dígitos estén completos, de ser así, se empaqueta la información y se envía a la estación base para identificar a la persona, caso contrario se muestra en pantalla un mensaje indicando que son 10 dígitos, el siguiente código permite mostrar en pantalla el mensaje de estado del proceso.

ObjPantallaCedula.setText("La Cédula debe contener 10 Digitos");

Otra forma de identificar a una persona es mediante el procesamiento de imágenes. Para esto en la pantalla principal al presiona el botón "Huella Digital" se envía una trama de datos hacia el Arduino como la que se observa en la Tabla 13-3, indicándole que active el sensor biométrico para escanear, almacenar y obtener las minutas de la huella digital, finalmente el Arduino empaqueta y envía la información hacia la estación central para llevar a cabo el proceso de reconocimiento de huellas digitales.

Tabla 13-3: Trama de datos de los botones "Datos" y "Huella Digital"

	2 bytes	1 byte	10 bytes	3 bytes
Nombre	Cabecera	Instrucción	Complemento	Fin
Huella Digital	EF 01	01	00 00 00 00 00 00 00 00 00 00	FFFFFF

Realizado por: Salazar Byron, 2019

Nota. Instrucción (01), activa el sensor biométrico, capturar imagen, convertir en minutas y enviar la información.

3.6.6.5. Implementación Arduino y Sensor Biométrico

El sensor biométrico está conectado con el Arduino a través del puerto Serial 2 como se observa en la Figura 19-3. Maneja grandes cantidades de información ya que la imagen está compuesta

por 38460 bytes razón por lo cual, para comunicar los módulos se utiliza la tasa de transferencia a 57600 baudio, una tasa menor provoca que se pierda la información.

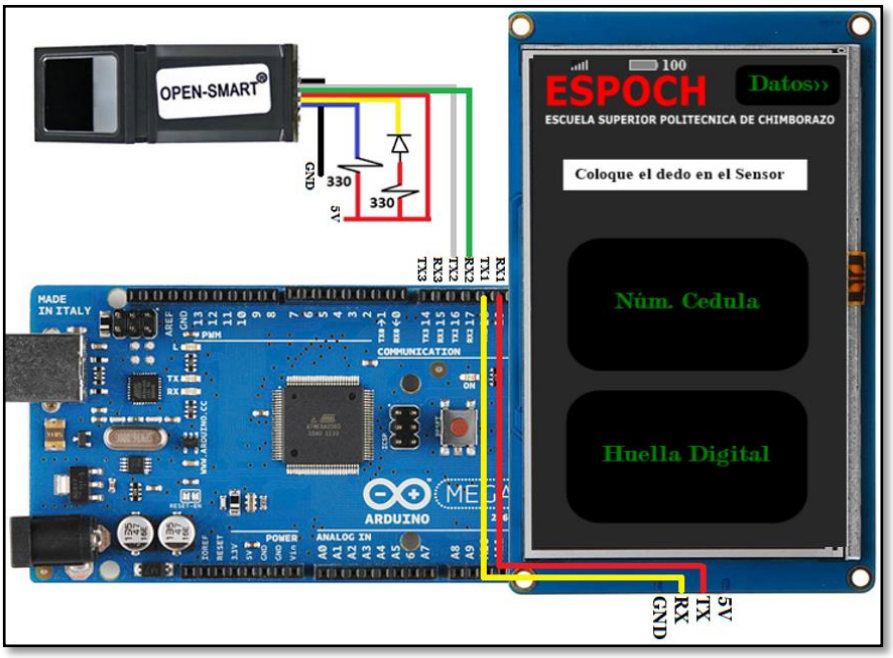


Figura 19-3. Conexión Arduino, Sensor biométrico y Pantalla.
Fuente: "Innova Domotics", por APM, 2018

3.6.6.5.1. Confirmación de comunicación entre el Arduino y el Lector Biométrico

Una vez que se alimenta el equipo terminal, el Arduino envía un paquete de comandos el cual se muestra en la Tabla 14-3, esta tabla contiene la trama de datos que sirve para confirmar si existe comunicación entre el Sensor Biométrico y el Arduino. El formato del paquete que contiene la trama de datos de acuse de recibo se muestra en la Tabla 15-3, esta trama indica si existe o no conexión entre ambos módulos.

Tabla 14-3: Trama de confirmación de conexión entre el Sensor y Arduino

2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes
Cabecera	Dirección del módulo	Identificador de paquete	Longitud del paquete	Código de instrucciones	Código de control	Checksum
EF01 H	FFFFFFFF H	01 H	0004 H	17 H	00 H	001C H

Fuente: (Grow, 2015)
Realizado por: Salazar Byron, 2019
Nota. Trama de datos para confirmar si existe comunicación entre el Sensor Biométrico y el Arduino

Tabla 15-3: Acuse de recibo de la confirmación de conexión.

2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
Cabecera	Dirección del módulo	Identificador de paquete	Longitud del paquete	Código de control	Checksum
EF01 H	FFFFFFFF H	07 H	0003 H	xx H	SUMA

Fuente: (Grow, 2015)
Realizado por: Salazar Byron, 2019
Nota. Trama de acuse de recibo, indica si existe comunicación entre el Arduino y el Sensor

El código de confirmación puede tomar los siguientes valores que significan:

- 00H: operación de puerto completa;
- 01H: error al recibir el paquete;
- 1dH: no funciona el puerto de comunicación

Si el Biométrico le indica al Arduino que existe conexión mediante la trama de acuse de recibo, el usuario ya puede operar con el equipo terminal. A continuación, se muestra los formatos de paquetes de datos para, capturar la imagen de la huella digital, y generar la plantilla de minucias con los rasgos característicos.

3.6.6.5.2. Captura de la imagen de la huella Dactilar

Para adquirir la imagen después de presiona el botón "Huella Digital", el Arduino reenvía la trama de datos de la Tabla 16-3, la cual ordena al lector biométrico que capture la imagen de la huella digital y la almacene en la memoria interna, el sensor parpadea para indicarle al usuario que coloque el dedo en pantalla. Finalmente, el sensor devuelve el código de confirmación.

Tabla 16-3: Trama de datos para capturar la imagen de la huella digital.

2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
Cabecera	Dirección del módulo	Identificador de paquete	Longitud del paquete	Código de instrucciones	Checksum
EF 01 H	FFFF FFFF H	01 H	00 03 H	01 H	0005 H

Fuente: (Grow, 2015)
Realizado por: Salazar Byron, 2019
Nota. Trama de captura de la imagen de la huella digital.

El formato del paquete que contiene la trama de datos del estatus de la captura de la huella se muestra en la Tabla 17-3. Esta trama muestra si la captura fue exitosa, existió error al recibir el paquete, no se puede detectar el dedo o falló al detectar la huella. La pantalla parpadea por un lapso de 8 segundos hasta que el usuario coloque el dedo.

Tabla 17-3: Acuse de recibo de la captura de la imagen

2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
Cabecera	Dirección del módulo	Identificador de paquete	Longitud del paquete	Código de confirmación	Checksum
EF01 H	FFFFFFFF H	07 H	0003 H	xx H	SUMA

Fuente: (Grow, 2015)
Realizado por: Salazar Byron, 2019
Nota. Trama de datos del estatus de la captura de la huella, indica si la operación tuvo o no éxito

El código de confirmación puede tomar los siguientes valores que significan:

- 00H: Captura exitosa del dedo;
- 01H: Error al recibir el paquete;
- 02H: No se puede detectar el dedo;
- 03H: Falla al capturar el dedo;

3.6.6.5.3. Generación de un archivo de caracteres de la imagen.

Si la trama de confirmación indica que la captura de la huella digital fue exitosa, el Arduino envía una trama de datos como la que se observa en la Tabla 18-3, para generar un archivo de caracteres a partir de la imagen y almacenar la información en los buffers (CharBuffer1 o CharBuffer2) este archivo posee las minutas o rasgos característicos de la huella digital.

Tabla 18-3: Trama para generar archivo de caracteres a partir de la imagen.

2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes
Cabecera	Dirección del módulo	Identificador de paquete	Longitud del paquete	Código de instrucciones	Número de Buffer	Checksum
EF 01 H	FFFF FFFF H	01 H	00 04 H	02 H	01H	00 08 H

Fuente: (Grow, 2015)
Realizado por: Salazar Byron, 2019
Nota. Trama para formar archivo de caracteres a partir de la imagen.

Luego de generar el archivo de caracteres, el sensor biométrico envía la trama de acuse de recibo para indicar el estado de la operación, si el codigo de confirmación es "00" en hexadecimal, significa que la operación fue exitosa, esta trama se puede observar en la Tabla 19-3.

Tabla 19-3: Acuse de recibo de la generación de archivo de caracteres de la Huella.

2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
Cabecera	Dirección del módulo	Identificador de paquete	Longitud del paquete	Código de confirmación	Checksum
EF 01 H	FFFF FFFF H	07 H	00 03 H	xx H	SUMA

Fuente: (Grow, 2015)
Realizado por: Salazar Byron, 2019
Nota. Trama de datos del estatus de la captura de la huella, indica si la operación tuvo o no éxito.

El código de confirmación puede tomar los siguientes valores que significan:

- 00H: se ha generado un archivo de caracteres completo;
- 01H: error al recibir el paquete;
- 06H: no se puede generar el archivo de caracteres debido a la imagen de huella digital desordenada;
- 07H: no se puede generar el archivo de caracteres debido a la falta de puntos de caracteres o al tamaño excesivo de la imagen de la huella digital;
- 15H: no se puede generar la imagen debido a la falta de una imagen primaria válida;

3.6.6.5.4. Descarga de la plantilla de minutas.

Si la trama de confirmación indica que la generación del archivo de caracteres fue exitosa, el Arduino vuelve a enviar una trama de datos como la que se observa en la Tabla 20-3, para descargar la información almacenada en el Buffer 1, la trama de datos especifica de que Buffer se desea extraer la plantilla.

Tabla 20-3: Formato de comandos para descargar la plantilla del Buffer 1.

2 bytes	4 bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes
Cabecera	Dirección del módulo	Identificador de paquete	Longitud del paquete	Código de instrucciones	Número de Buffer	Checksum
EF 01 H	FFFF FFFF H	01 H	00 04 H	08 H	01H	00 0E H

Fuente: (Grow, 2015)
Realizado por: Salazar Byron, 2019
Nota. Trama para descargar archivo de caracteres.

El formato de la trama de acuse de recibo, que corresponde a la descargar de la plantilla de minucias, se observa en la Tabla 21-3, la trama indica el estado de la operación, si el codigo de confirmación es "00" el sensor envia la trama de datos con las minucias obtenidas a partir de la imagen de la huella digital.

Tabla 21-3: Acuse de recibo de la Descarga de la Plantilla de minucias.

2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
Cabecera	Dirección del módulo	Identificador de paquete	Longitud del paquete	Código de confirmación	Checksum
EF 01 H	FFFF FFFF H	07 H	00 03 H	xx H	SUMA

Fuente: (Grow, 2015)
Realizado por: Salazar Byron, 2019
Nota. Trama de datos del estatus de la descarga de la plantilla de minucias.

El código de confirmación puede tomar los siguientes valores que significan:

- 00H: listo para transferir el siguiente paquete de datos;
- 01H: error al recibir el paquete;
- 0dH: error al cargar la plantilla;

Después de responderle al Arduino, el módulo descargará dos tramas, la primera está compuesta por paquetes de datos que constituyen la información de los rasgos característicos de la huella digital y la segunda trama indica el final de la transferencia de la información. La Tabla 22-3, muestra el formato de los dos paquetes que descarga el sensor biométrico.

Tabla 22-3: trama de datos y trama de fin de paquete de datos.

Nombre	Bytes	Trama 1 (Hex)	Trama 2 (Hex)
Cabecera	2 bytes	EF 01	EF 01
Dirección	4 bytes	FF FF FF FF	FF FF FF FF
Identificador	1 byte	02	08
Tamaño	2 bytes	01 02	01 02
Datos	256 bytes	Datos, minucias	Datos
Checksum	2 bytes	Suma de bytes	Suma de bytes
Descripción		Trama de datos.	Fin de paquete de datos

Realizado por: Salazar Byron, 2019
Nota. Identificados (02) Datos, Identificador (08) Fin de paquete de datos.

El tamaño del archivo de características es de 256 bytes, el número de minucias de dicho archivo no es más de 50. Los primeros 56 bytes corresponden al encabezado, utilizado para la información general; Los últimos 200 bytes son para almacenar información de los rasgos característicos, la Tabla 23-3, muestra la estructura de la trama de datos de los rasgos característicos.

Tabla 23-3: Formato de la trama de datos de los rasgos característicos.

Nombre	Bytes	Descripción
Bandera	0	Bandera de archivo de característica. "0": el archivo de características no es válido o eliminado
Tipo	1	Tipo de archivo, características.
Calidad	2	Calidad de característica, 0 ~ 100.
Número	3	Número de Minutas dentro del rango de 5 ~ 50.
SN	4 ~ 5	Asistente de búsqueda.
Información de fondo	6 ~ 39	Información de la tabla de fondo
Punto de singularidad	40 ~ 43	(x, y) información de coordinación de los dos puntos centrales.
Reservado	44 ~ 55	Reservado
Minutas	56 ~ 255	Minutas

Realizado por: Salazar Byron, 2019

Nota. Formato de la trama de rasgos característicos.

Una vez descargada la información el Arduino verifica la integridad de los paquetes realizando una suma de comprobación. Si la suma es correcta, el Arduino empaqueta la información tomando únicamente el tercer byte que indica el número de minucias y todos los bytes a partir del 56 que conforman dichas minucias. La Tabla 24-3 muestra la nueva trama empaquetada.

Tabla 24-3: Trama de datos para procesamiento de Imágenes.

Nombre	Bytes	Datos	Descripción
Cabecera	2	EF 01	Inicio de trama
Instrucción	1	01	Procesamiento de imágenes.
Número de Minucias	1	Cantidad	Tamaño del paquete
Datos de Minucias	20 ~ 200	Datos	Minucias (Posición (x; y), ángulo, tipo)
Fin de trama	3	FF FF FF	Final de trama.

Realizado por: Salazar Byron, 2019

Nota. Formato de la trama para procesamiento de imágenes.

Una vez empaquetada la información ya sea con los dígitos de la cédula o los rasgos característicos, se envía la información de forma inalámbrica hacia la estación central a través del módulo inalámbrico SIM800L para realizar el proceso de reconocimiento de personas.

3.6.6.6. Implementación Arduino y Módulo SIM800L

El Sim800L permite la comunicación inalámbrica entre la estación base y el equipo terminal a través de mensajes de texto. Es cuatri banda o de bandas liberadas, trabaja en las frecuencias de (850/950/1800/1900 MHz), para conectarse a la red se utiliza una tarjeta micro Sim de la Compañía Claro; pero se puede usar cualquier tarjeta que trabaje en las frecuencias indicadas.

3.6.6.6.1. Envió de mensajes SMS.

El Sim800L está conectado al puerto Serie 3 del Arduino a una velocidad de 57600 baudios, ya que trabaja en sincronía con el puerto Serial 2 a través del cual se conecta el Biométrico y el Arduino. Si los puertos 2 y 3 trabajan a distintas velocidades, se pierde información ya que la tasa de transferencia es diferente. La Figura 20-3, muestra el acople de todos los módulos.

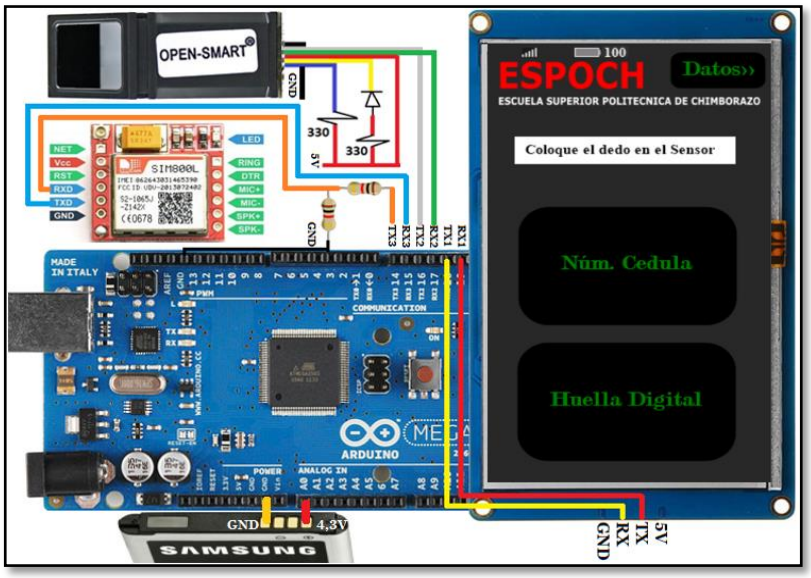


Figura 20-3. Conexión completa del Equipo Terminal.
Realizado por: Salazar Byron, 2019.

El Arduino se comunica con el módulo Sim mediante comandos AT para configurarlo, controlarlo o enviar mensajes. Los mensajes contienen las minucias o los números de cédula, estos datos son almacenados en un array para ir enviando uno por uno. Los dígitos de la cédula se envían únicamente en un mensaje; sin embargo, las minucias se envían en varios ya que solo se puede escribir hasta 140 caracteres. La Tabla 25-3, contiene el código para enviar mensajes.

Tabla 25-3: Envió de mensajes SMS mediante comandos AT

Programa	Descripción
Serial3.print("AT\r");	Es el comando más básico, inicializa la detección automática de baudios.
Serial3.print("AT+CNMI=2,2,0,0,0\r");	especifica cómo se deben manejar los mensajes SMS recién llegados.
Serial3.write("AT+CMGF=0\r");	Selecciona el formato del mensaje: Unidad de Datos de Protocolo (PDU)
Serial3.write("AT+CMGS=\"+593986502852\"\r");	Número de teléfono al cual se envía el mensaje.

For(int i; i<=sizeof(ContenidodelMensaje); i++){
Serial3.print(ContenidodelMensaje[i]);
}
	Contenido, dígitos de cédula, minucias. Se envía uno a uno cada byte.

Serial3.write((char)26);	Enviar el carácter de control ASCII 26 indicando el final del cuerpo del mensaje
Serial3.print("AT+CSCLK=2\r");	ingresa al modo de suspensión para ahorrar energía

Fuente: (Conel, 2012)
Realizado por: Salazar Byron, 2019
Nota. La velocidad de transferencia es de 57600 baudios.

El comando AT+CNMI=2,2,0,0,0 posee cinco parámetros a continuación se describe cada uno de ellos (<mode>,<mt>,<bm>,<ds>,<bfr>) :

- <mode>: controla cómo notificar al Equipo Terminal (TE)
  (2) Cuando la línea de datos está inactiva, notifique a TE directamente; de lo contrario, la notificación se almacenará en caché primero y luego se enviará cuando la línea de datos esté activa.
- <mt>: Establece el contenido del TE de notificación y almacenamiento de mensajes cortos.
  (2) Para mensajes cortos, guárdelos en la tarjeta SIM y notifique a TE.
- <bm>: Configura la transmisión celular
  (0) No se envían notificaciones al Equipo Terminal.
- <ds>: informe de estado
  (0) Sin notificación.
- <bfr>: Parámetro relacionado con el buffer del terminal.
  (0) Sin notificación.

3.6.6.7. Monitoreo de la señal inalámbrica y de la batería.

El Arduino monitorea a cada segundo los niveles de la señal inalámbrica recibida y de la carga de la batería. El nivel de la señal inalámbrica se representa de forma gráfica y el porcentaje de carga de la batería en forma numérica, estos valores se muestran en una barra en la parte superior de la interfaz.

3.6.6.7.1. Monitoreo del nivel de señal recibida.

Para llevar a cabo el monitoreo de la señal, no es necesario realizar ninguna conexión adicional. El monitoreo se realiza mediante el envío de comandos AT (AT+CSQ) hacia el módulo Sim. Este comando devuelve la fuerza de la señal de la red registrada. La respuesta tiene el formato + CSQ: <rssi>, <ber>, donde <rssi> indica la intensidad de la señal recibida y <ber> es la tasa de error de bits del canal (Conel, 2012), solo se usa el parámetro RSSI para el monitoreo.

RSSI ayuda a determinar si una señal es suficiente para establecer una conexión inalámbrica. RSSI es un índice relativo, mientras que dBm se considera un número absoluto que representa los niveles de potencia en mW (miliwatts). Por lo tanto, es necesario transformar RSSI a dBm para verificar la potencia de la señal de recepción. Para convertir el índice RSSI a dBm se utiliza la Ecuación (4).

$$dBm = Indice(rssi) \times 2 - 113 \quad (4)$$

La escala tiende al valor cero (0) como centro; y representa 0 RSSI o 0 dBm. Generalmente la escala se expresa dentro de valores negativos; cuanto más negativo es el resultado, existe mayor pérdida de señal (Admired). La Tabla 26-3, muestra la calidad de la señal inalámbrica.

Tabla 26-3: Potencia de la señal de recepción inalámbrica

Índice (RSSI)	Potencia (dBm)	Estado de la señal
0 ~ 1; 99	-113 ~ -111; 85	Sin cobertura
2 ~ 9	-109 ~ -95	Bajísima cobertura
10 ~ 14	-93 ~ -85	Baja cobertura
15 ~ 19	-83 ~ -75	Buena/Media cobertura
20 ~ 30	-73 ~ -53	Muy buena
31 ~ 56	-51 ~ -1	Excelente

Realizado por: Salazar Byron, 2019
Nota. cuanto más se acerque el valor a 0 (cero) dBm, más fuerte será la señal.

La calidad de la señal es presentada en la interfaz de forma gráfica mediante celdas, cada gráfica va acorde al nivel de la potencia según se observa en la Tabla 26-3, cuanto más se acerque a cero dBm, más fuerte es la señal y más barras se observan, cuanto más negativa es la señal menor cobertura y menos barras se ven. La Figura 21-3, muestra de forma gráfica la potencia de la señal recibida.

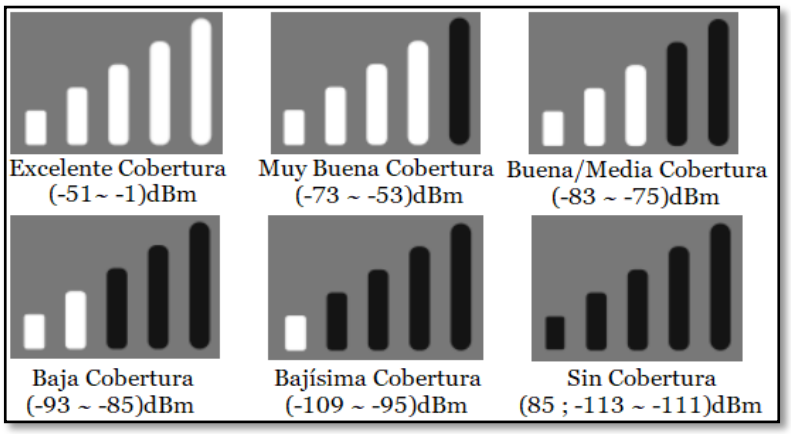


Figura 21-3. Representación gráfica de la intensidad de la señal recibida.
Realizado por: Salazar Byron, 2019.

3.6.6.7.2. Monitoreo de la Carga de la Batería

El monitoreo de la batería se realiza a cada segundo a través de la entrada analógica A0 del Arduino. Una señal analógica es una magnitud que puede tomar cualquier valor dentro de un intervalo -Vcc y +Vcc en el caso de la batería Samsun entre (3,3 ~ 4,3) Voltios, ya que por recomendaciones del fabricante se debe mantener por arriba del valor mínimo para cuidar su vida útil; por lo tanto, si la batería alcanza los 3,3 V la interfaz gráfica muestra cero por ciento lo que significa que se debe volver a cargar.

No se mide el valor analógico con todos sus decimales, sino que se clasifica dentro de 2^N niveles, que en intervalos se refiere a $(2^N - 1)$. El ancho de este intervalo medido en mili Voltios es la precisión de la señal. Cuanto mayor sea el número de bits, mayor será el número de intervalos; por lo tanto, mejor será la precisión de la medición de los niveles de voltaje (Llamas, 2014).

El Arduino Mega, posee entradas analógicas de 10 bits de resolución, esto proporciona 1024 niveles digitales; por lo tanto, si 5V se divide entre 1024, se obtiene una precisión de medida de 4,88mV por intervalo (Llamas, 2014). Para mostrar en la interfaz, se escala 3,3V a 0% y 4,3V a 100%, para esto se utiliza la ecuación (5):

$$\frac{1024}{Nivel?} \times \frac{5V}{Voltaje} \tag{5}$$

Dónde:

$$\boldsymbol{Nivel}\ (\mathbf{675}) = 3{,}3V = 0\% \ \cap\ \boldsymbol{Nivel}\ (\mathbf{860}) = 4{,}4V = 100\%$$

Teóricamente los 3,3 Voltios se encuentran en el nivel 675 lo que equivale al 0% de la carga de la batería y los 4,3 Voltios se encuentran en el nivel 860 lo que equivale 100% de la carga; pero en la práctica existe ligeras variaciones, la Tabla 27-3, muestra el código utilizado para trabajar con el conversor análogo digital.

Tabla 27-3: Conversor análogo Digital para monitorear la carga de la batería.

Programa	Descripción
int sensorValue = 0;	Declaración de variable para almacenar datos de la entrada A0
int outputValue = 0;	Declaración de variable para almacenar datos escalados
sensorValue = analogRead (sensorPin);	Lee el puerto analógico A0
outputValue = map (sensorValue, 725, 876, 0, 100);	Escalamiento de valores: (0%=3,3V=725); (100%=4,3V=876)
ObjBateria.setValue(outputValue);	Impresión en pantalla nextion el porcentaje de carga de la batería

Fuente: (Crespo, 2018)
Realizado por: Salazar Byron, 2019
Nota. 0% equivale a 3,3V para cargar la batería y cuidar la vida útil sin sobrepasar límites.

Una vez que se monitorea el estado de la señal de recepción y el nivel de carga de la batería, se imprimen en pantalla, en las cuatro páginas que conforman la interfaz gráfica, de esta manera el usuario determina si tiene señal o si el nivel de batería es apto para operar. La Figura 22-3, muestra las señales impresas en pantalla.

Figura 22-3. Intensidad de la señal y nivel de la batería.
Realizado por: Salazar Byron, 2019

3.6.6.8. Diseño del chasis del Equipo Terminal en SOLIDWORKS

Las partes fueron modeladas en el programa de diseño SOLIDWORKS, el chasis está compuesto por 9 como se ve en el ANEXO 2. La Figura 23-3, muestra el equipo terminal en su chasis.

Figura 23-3. Equipo Terminal ensamblado en la caja 3D.
Realizado por: Salazar Byron, 2019.

3.6.7. Estación central

La estación central es un sistema embebido compuesto por varios elementos que permiten comunicarse de forma inalámbrica con el equipo terminal y llevar a cabo el reconocimiento de personas.

La estación central recepta la trama de datos enviada desde el equipo terminal, procesa la información para identificar personas y en caso de reconocer algún individuo, arma un paquete y lo reenvía. Los datos recibidos corresponden al número de cédula o plantilla de minucias la cual contiene rasgos característicos de la huella digital, estos parámetros ayudan a identificar a una persona mediante procesamiento de imágenes o consultas SQL.

El diagrama de flujo de la Figura 24-3, muestra la estructura y explica el funcionamiento de la estación central para llevar a cabo el reconocimiento de personas, a continuación de describe cada elemento que conforma la estación central.

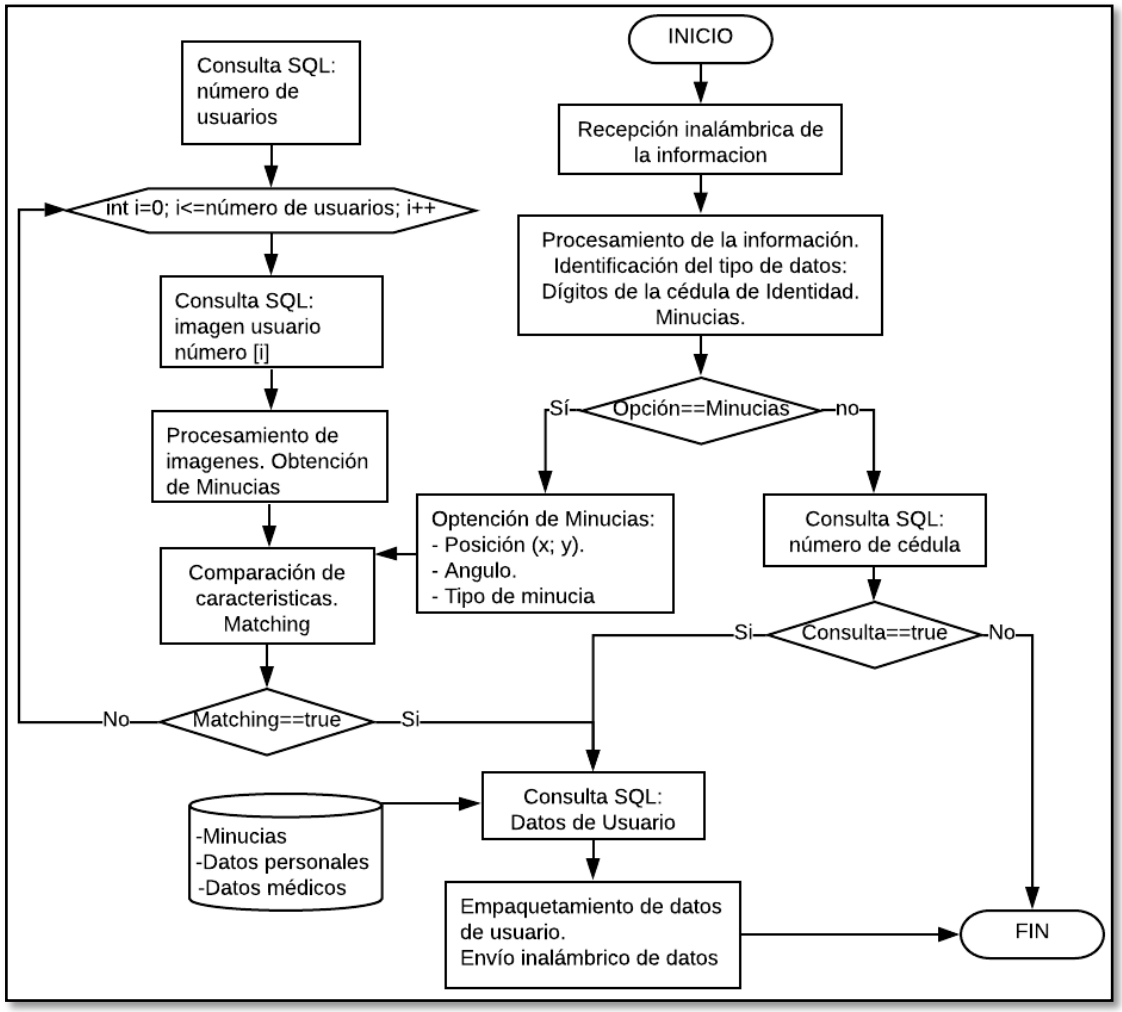


Figura 24-3. Diagrama de la Estación Base.
Realizado por: Salazar Byron, 2019

La estación central consta de los siguientes elementos de Hardware y Software:

- Computador hp, core i7
- Tarjeta de transmisión inalámbrica
- Servidor XAMPP: phpMyAdmin + base de datos MariaDB.
- Programa principal de procesamiento

3.6.7.1. Tarjeta de transmisión inalámbrica

La tarjeta de transmisión inalámbrica permite la comunicación entre la estación base y el equipo terminal. Esta tarjeta está compuesta por una placa cuyas pistas fueron diseñadas en Proteus como se observa en el ANEXO 1, esta placa se encarga de integrar los módulos, para receptar la

información de forma inalámbrica y enviar al computador con el fin de procesar los datos he identificar personas, una vez identificada la persona se reenvía sus datos al equipo terminal. El ANEXO 5 contiene los programas de Arduino utilizados.

El Arduino se conecta con el Módulo SIM800L a través del puerto Serial 2 a una velocidad de 57600 baudios, ya que debe ser la misma a la que trabaja el módulo inalámbrico en el equipo terminal, si no están en sincronía, no se pueden comunicar. Los datos recibidos por el puerto 2 del Arduino, son reenviados automáticamente al computador a través del Serial 0, por tal motivo deben estar sincronizados a la misma velocidad de 57600 baudios. La Figura 25-3, muestra la tarjeta inalámbrica ensamblada.

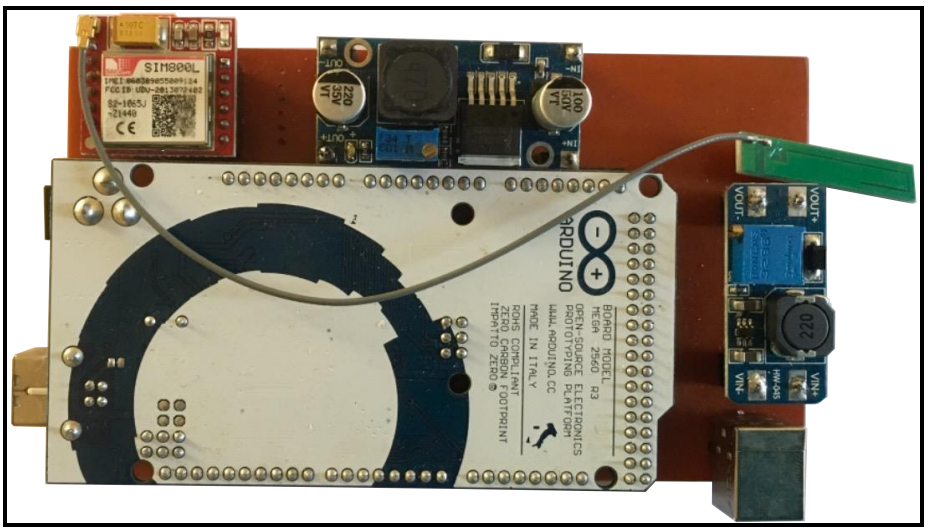


Figura 25-3. Tarjeta de transmisión Inalámbrica.
Realizado por: Salazar Byron, 2019.

3.6.7.2. Base de datos MariaDB

Se necesita almacenar información y realizar consultas, por esta razón se utilizó el paquete de Software Xampp que contiene el sistema relacional de bases de datos MariaDB, el servidor Web Apache y el programa de administración de bases de datos phpMyAdmin. En combinación con el servidor web Apache y el lenguaje PHP, MySQL sirve para el almacenamiento de datos.

Una vez instalado Xampp, se ejecuta el panel de control y se levanta los servicios de Apache y Mysql. Cuando se levantan los servicios, los módulos de Apache y Mysql se tornan de color verde y sus correspondientes botones dicen stop. La Figura 26-3, muestra el panel de control con los servicios levantados.

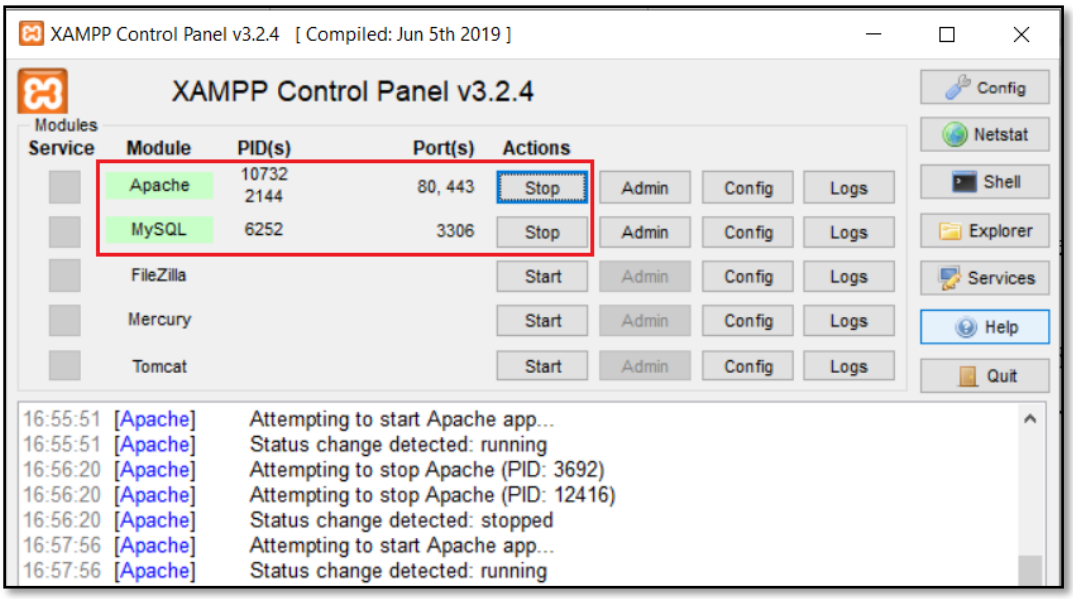


Figura 26-3. Servicios Apache y Mysql levantados.
Realizado por: Salazar Byron, 2019

Una vez levantados los servicios de Apache y Mysql y con la ayuda de cualquier navegador, en la barra de búsqueda se escribe "http://localhost/phpmyadmin/". Esta dirección permite ingresar a phpmyadmin para administrar y crear bases de datos. Para crear una base de datos, se selecciona la pestaña nueva y se asigna un nombre, en este caso se le ha asignado el nombre de "historial_medico".

La base de datos creada consta de cuatro tablas: usuarios_registrados, imágenes_plantillas, datos_personales y datos_medicos. Estas tablas están relacionadas mediante campos comunes denominados clave primaria y clave foránea. La Tabla 28-3, denominada usuarios registrados, contiene los siguientes campos: id_usuarios, apellidos y nombres. El campo id_usuarios es la clave primaria y permite crear relaciones con otras tablas para realizar consultas.

Tabla 28-3: Usuarios Registrados

ID_USUARIOS	APELLIDOS	NOMBRES
1	SALAZAR MOPOSITA	BYRON
2	ARROBA POVEDA	CUMANDA
3	SALAZAR MOPOSITA	CHRISTIAN

Realizado por: Salazar Byron, 2019
Nota. id_usuario es la clave primaria y permite relacionarse con las tablas

La Tabla 29-3, denominada "datos_personales" contiene los siguientes campos: id_datos_personales, id_usuarios, cédula, dirección, fecha_nacimiento, teléfono, nombre_referencia, teléfono_referencia. El parámetro "id_usuarios", está establecido como clave foránea y a través de este campo se realiza la relación con la clave primaria de la tabla "usuarios_registrados".

Tabla 29-3: Datos Personales

CAMPOS	USUARIO 1	USUARIO 2	USUARIO 3
ID_DATOS_PERSONALES	1	2	3
ID_USUARIOS	1	2	3
CÉDULA	1803725066	1803963691	1804431458
DIRECCIÓN	AMBATO	AMBATO	AMBATO
FECHA_NACIMIENTO	1985-03-05	1986-09-24	1998-01-14
TELÉFONO	0979159888	0987357153	0995865993
NOMBRE_REFERENCIA	SALAZAR CONORS	ARROBA FRANCISCO	FRANKLIN VALENCIA
TELÉFONO_REFERENCIA	0034670531371	032415032	032415479

Realizado por: Salazar Byron, 2019
Nota. Id_usuarios es la clave foránea y se relaciona con la clave primaria de la tabla "Usuarios_registrados".

La Tabla 30-3, denominada "datos_médicos" contiene los siguientes campos: id_datos_médicos, id_datos_personales, tipo_sangre, alergias, enfermedades, medicamentos. La columna "id_datos_personales" está establecida como clave foránea y a través de este campo se realiza la relación con la tabla "datos_personales" a través de la columna que lleva su mismo nombre.

Tabla 30-3: Datos Médicos.

CAMPOS	USUARIO 1	USUARIO 2	USUARIO 3
ID_DATOS_MÉDICOS	1	2	3
ID_DATOS_PERSONALES	1	2	3
TIPO_SANGRE	ORH+	ORH+	ORH+
ALERGIAS	PENICILINA	NO	NO
ENFERMEDADES	NO	HIPOTIROIDISMO	NO
MEDICAMENTOS	NO	LEVOTIROXINA	NO

Realizado por: Salazar Byron, 2019
Nota. Id_datos_personales es la clave foránea y se relaciona con la clave primaria de la tabla "Datos_personales".

La Tabla 31-3, denominada "imágenes_plantillas", posee los siguientes campos: id_usuarios, imágen1, imágen2, plantilla1, plantilla2. La columna "id_usuarios" está establecida como clave foránea y a través de este campo se realiza la relación con la clave primaria de la tabla "usuarios_registrados". Las relaciones se realizan para llevar a cabo consultas a través del lenguaje SQL.

Tabla 31-3: Imágenes y Plantillas de Minucias.

CAMPOS	USUARIO 1	USUARIO 2	USUARIO 3
ID_USUARIOS	1	2	3
IMAGEN 1	38460 bytes conforman la imagen 1		
IMAGEN 2	38460 bytes conforman la imagen 2		
PLANTILLA 1	256 bytes conforman la plantilla 1		
PLANTILLA 2	256 bytes conforman la plantilla 2		

Realizado por: Salazar Byron, 2019
Nota. Id_usuarios es la clave foránea y a través de este campo se relaciona con la clave primaria de la tabla "usuarios_registrados"

La Figura 27-3, muestra la relación creada entre las tablas que contienen las imágenes y la información de los usuarios. La relación se realiza entre la clave primaria y la clave foránea de diferentes tablas, cada tabla puede tener una sola clave primaria y varias claves foráneas, antes de crear cada campo, se especifica el tipo de dato a manejar y el número de caracteres que pueden almacenar.

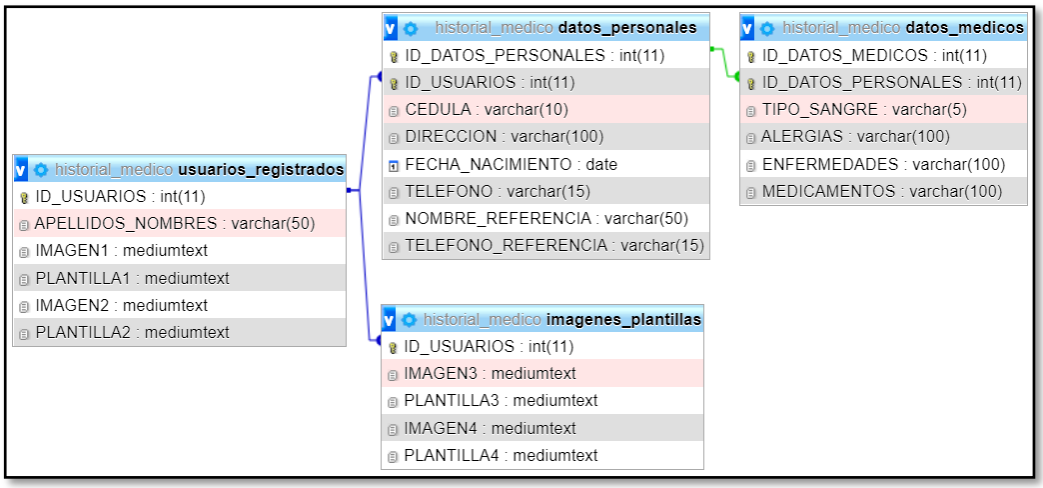


Figura 27-3. Relación entre Tablas de Datos e Imágenes.
Realizado por: Salazar Byron, 2019

3.6.7.3. Conexión con la base de datos

Una vez creada la base de datos se realizar la conexión para hacer posible el acceso a cualquier dato desde la interfaz. Para esto se descarga e instala el conector de base de datos ODBC (Open DataBase Connectivity). Una vez instalado el conector, se ingresa a la siguiente ruta (Panel de control\Todos los elementos de Panel de control\Herramientas administrativas) y se abre ODBC, realizado esto se despliega la ventana del administrador de origen de datos como se observa en la Figura 28-3.

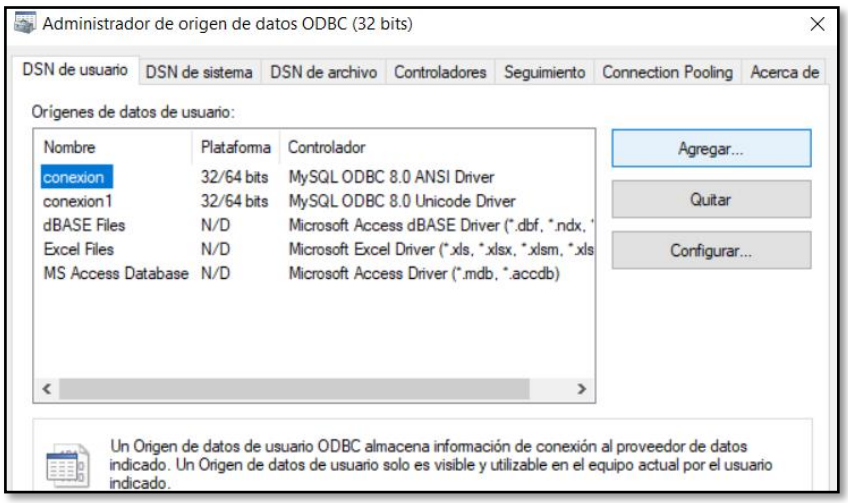


Figura 28-3. Administrador de origen de datos ODBC.
Realizado por: Salazar Byron, 2019

En la pestaña “DSN de usuario”, pulsar en “Agregar” para crear una conexión y acceder a la información. En “Data Source Name” se asigna el nombre de la conexión “conexión”, en “TCP/IP Server” se especifica el servidor “localhost”, el puerto por defecto es el “3306”, en “User” se escribe el nombre del usuario “root” y en “Database” se indica el nombre de la base de datos con la que se realizara la conexión “historial_medico”. Finalmente, al presionar en test se verifica si la conexión fue exitosa como se muestra en la Figura 29-3.

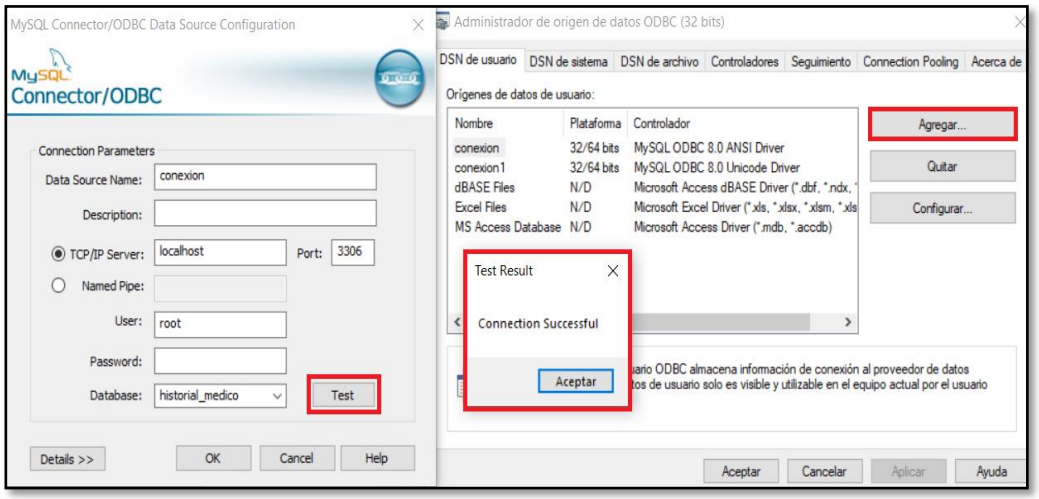


Figura 29-3. Conexión con la base de Datos.
Realizado por: Salazar Byron, 2019

3.6.7.4. Diseño del programa principal.

El programa principal de la estación central está diseñado en LabVIEW y corre en el computador, su propósito es recibir, almacenar, procesar y consultar la información. Para comenzar a trabajar con el programa es necesario habilitar el puerto de comunicación serie para enviar y recibir información desde las tarjetas electrónicas.

3.6.7.4.1. Comunicación serial para la captura y envío de la información

La comunicación entre el Arduino Mega2560 y el computador se da a través del puerto Serie. Los parámetros a configurar son:

- Selección del Puerto virtual VISA para la comunicación serie (Variable).
- Tasa de transferencia (57600).
- Bits de datos (8).

La Figura 30-3, muestra la conexión para comunicar el computador con el Arduino. Este bloque representa la comunicación serie que se encarga de recibir y enviar la información desde y hacia el equipo terminal. Vale la pena aclarar que la tasa de transferencia está fijada a 57600 por que a esa velocidad trabaja el módulo inalámbrico, en el terminal VISA se selecciona el puerto de comunicación (COM).

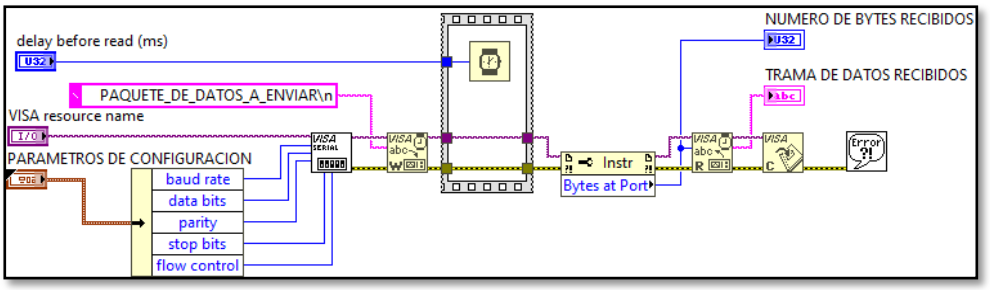


Figura 30-3. Comunicación Serie
Realizado por: Salazar Byron, 2019

3.6.7.5. Programa Principal, almacenamiento de la información

Antes de comenzar con el reconocimiento de personas, es necesario tener registrados usuarios en la base de datos para realizar las pruebas. El programa principal posee varias pestañas como: Principal, Registro de Huellas Digítales, Registro de Usuarios y Consulta de Usuarios, a continuación, se describe cada pestaña y cómo funciona.

3.6.7.5.1. Registros de Huellas Digítales.

Una vez establecida la comunicación entre el computador y los módulos electrónicos, se puede comenzar a trabajar en el registro de huellas digitales. Para el registro, se utiliza el equipo terminal con un programa diferente para aprovechar el sensor biométrico. La Figura 31-3, muestra la ventana de registro y El ANEXO 3 contiene parte de las huellas digitales almacenadas.

Figura 31-3. Ventana de registro de Huellas Digítales.
Realizado por: Salazar Byron, 2019.

Al presionar el botón “Capturar Huella Digital”, se envía la trama de datos “EF 01 FF FF FF FF 01 00 03 0100 05” para activar el sensor biométrico y capturar la huella digital, el botón “Verificar Huella” revisa el estado de la operación, si esta fue exitosa, se envía la trama “EF 01

FF FF FF FF 01 00 03 0A 00 0E" para descarga la imagen, finalmente se procesan los datos para obtener las minucias y almacenarlas.

Una vez capturada la imagen, esta es almacenada en el buffer del sensor con una resolución de 256 x 288 pixeles. Cuando se descarga a través de UART, para acelerar la velocidad, solo se transfieren los 4 bits más significativos del valor de cada píxel (es decir, 16 grados de gris) y se los agrupa en un solo byte, esto reduce a la mitad la cantidad de datos transmitidos, la Tabla 32-3, muestra el proceso utilizado para comprimir la información. Al reconstruir la imagen el pixel de 16 grados de gris se ampliará al formato de 256 grados de gris.

Tabla 32-3: Comprimiendo dos bytes en uno solo

E0								**FF**							
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
1	1	1	0	0	0	0	0	1	1	1	1	1	1	1	1
1	1	1	0	No utilizados				1	1	1	1	No utilizados			
1		1		1		0		1		1		1		1	
E								F							

Realizado por: Salazar Byron, 2019
Nota. Técnica para comprimir la imagen

Para reconstruir la huella digital es necesario descomprimir la información y obtener la resolución completa de 256 x 288 pixeles, esto permite comenzar con el procesamiento de la imagen para encontrar los rasgos característicos y de esta manera comparar con las minucias enviadas desde el equipo terminal.

3.6.7.5.2. Registro de Usuarios en la Base de Datos

Antes de llevar a cabo el registro de usuarios, es necesario establecer la conexión con la base de datos, para esto se implementa el diagrama de la Figura 32-3, donde a través de la conexión creada se puede realizar consultas o ingresar datos, para utilizar automáticamente la conexión creada, a la entrada de la función se asigna la constante "falso" caso contrario se debe indicar de forma manual que conexión utilizar.

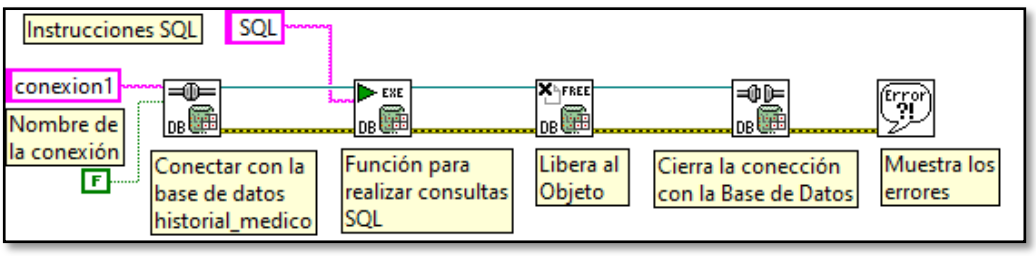


Figura 32-3. Conexión con la base de datos.
Realizado por: Salazar Byron, 2019.

Para ingresar información, se abre la ventana “Registro de Usuarios”, se llena los campos en base a la información establecida como se observa en la Figura 33-3, y se presiona el botón “Registrar Usuario”. Al presionar el botón, se almacena la información ingresada, las imágenes y las plantillas de minucias mediante sentencias SQL. Las imágenes y minucias son almacenadas en formato numérico.

Figura 33-3. Ventana de Registro de Usuarios.
Realizado por: Salazar Byron, 2019

La Tabla 33-3, muestra los comandos SQL utilizados para almacenar la información en la base de datos.

Tabla 33-3: Sentencias SQL para Registro de Datos de Usuarios.

Tabla	Sentencias SQL
usuarios_registrados	INSERT INTO usuarios_registrados (ID_USUARIOS, APELLIDOS _NOMBRES) VALUES (‘Auto incremento’, ‘Apellido y Nombre’)
datos_personales	INSERT INTO datos_personales (ID_DATOS_PERSONALES, ID_ USUARIOS, CEDULA, DIRECCION, FECHA_NACIMIENTO, TELEFONO, NOMBRE_ REFERENCIA, TELEFONO_REFERENCIA) VALUES (‘Datos de los campos’, ..)
datos_medicos	INSERT INTO datos_medicos (ID_DATOS_MEDICOS, ID_DATOS _PERSONALES, TIPO_SANGRE, ALERGIAS, ENFERMEDADES, MEDICAMENTOS) VALUES (‘Datos de los Campos’...)
imagenes_plantillas	INSERT INTO imagenes_plantillas (ID_USUARIOS, IMAGEN1, IMAGEN2, PLANTILLA1, PLANTILLA2) VALUES (‘Datos de los Campos’...)

Realizado por: Salazar Byron, 2019
Nota. Sentencias SQL para almacenamiento de la información.

Para llevar a cabo el reconocimiento de personas mediante procesamiento de imágenes, se utilizan dos huellas digitales, se pueden utilizar más para aumentar la efectividad del reconocimiento, pero también aumenta el tiempo del proceso. Con el objetivo de reducir el tiempo en el reconocimiento de personas, se procesa la información y se almacena la plantilla de minucias.

3.6.7.6. Programa Principal, procesamiento de imágenes

Una vez descargada la información, se realiza el procesamiento de la imagen para obtener la plantilla de minucias y luego almacenarla. Las etapas no se muestran en el programa principal, pero se llevan a cabo con la finalidad de extraer los rasgos característicos; por lo tanto, si el equipo terminal solicita identificar alguna persona, el sistema consulta las plantillas almacenadas y realiza el cotejo entre minucias. La Figura 34-3, muestra el procedimiento utilizado.

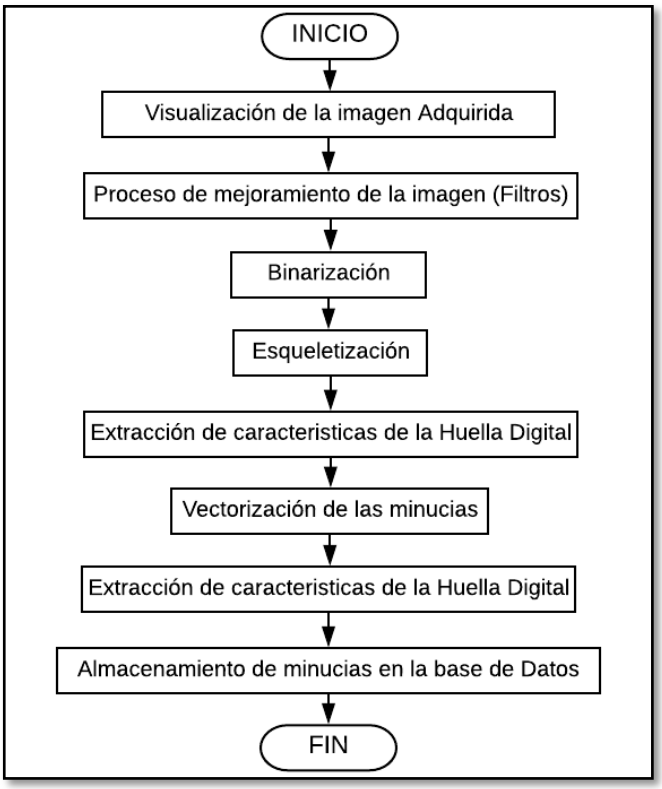


Figura 34-3. Proceso Digital de imágenes para la extracción características.
Realizado por: Salazar Byron, 2019

3.6.7.6.1. Lectura y construcción de la imagen

Para iniciar con el procesamiento de las huellas, se construye la imagen a partir de los datos recibidos. La información está compuesta por 144 tramas de 267 bytes. 256 bytes de cada paquete corresponden al valor de los pixeles. La imagen esta comprimida, ya que cada byte contiene el valor de 2 pixeles; por lo tanto, al descomprimir se tiene el doble de bytes y 288 filas de pixeles.

Para construir la imagen, cada byte se transforma a código binario y se divide en dos partes, cuyos valores corresponden a los bits más significativos de los 2 nuevos bytes generados, para esto se añade 4 ceros los cuales corresponden a los bits menos significativos, completando así dos bytes. La Tabla 34-3, muestra el procedimiento de descompresión utilizado para construir la imagen.

Tabla 34-3: Descompresión de Datos para la Construcción de la imagen.

E (14)								F (15)							
1		1		1		0		1		1		1		1	
1	1	1	0	0	0	0	0	1	1	1	1	0	0	0	0
E				0				F				0			

Realizado por: Salazar Byron, 2019

Nota. Técnica para descomprimir la imagen.

Una vez que se descomprime la información se obtiene 288 tramas de datos de 256 bytes que representan los pixeles. Primero se crear un espacio de memoria y se asigna un nombre que sirve para visualizar la imagen, este espacio se libera cuando finaliza el proceso. La Figura 35-3, muestra la huella digital construida con una resolución de 256 por 288 pixeles, a partir de esta figura se obtiene los rasgos característicos.

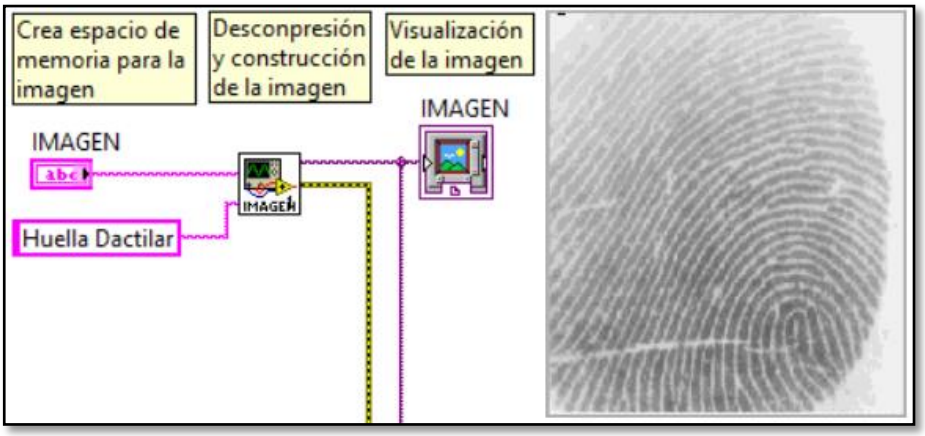


Figura 35-3. Descompresión y Construcción de la Imagen.

Realizado por: Salazar Byron, 2019.

3.6.7.6.2. Elección de la zona de interés

En caso de utilizar algún dispositivo con resolución diferente, este proceso se encarga de seleccionar la zona de interés o (ROI), recorta la imagen a un tamaño de 256x288 pixeles a través de un nodo de propiedad del visualizador de la imagen original, como se observa en la Figura 36-3. Esto permite enfocarse en la zona que tiene mayor cantidad de información.

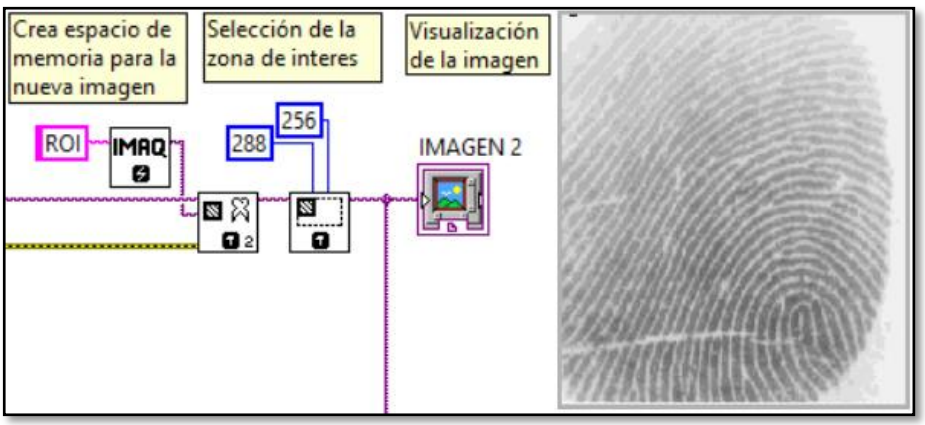


Figura 36-3. Selección de la Zona de Interés.

Realizado por: Salazar Byron, 2019.

3.6.7.6.3. Normalización

La etapa de Normalización, resalta las crestas en una imagen de huella dactilar en escala de grises. Las intensidades en escala de grises varían de 0 a 255 siendo el valor 0 el elemento más oscuro y 255 el más claro, dependen de la claridad y calidad con que se captura una imagen; por lo tanto, el objetivo de esta etapa es fijar los valores de la media y la varianza de la imagen a un valor establecido previamente por el usuario, esto dependen de factores tales como:

- Grado de sequedad o humedad de la piel de la yema del dedo
- Nivel de presión que se ejerce sobre la superficie del sensor

Varios de los procesos que intervienen en el análisis de las huellas dactilares, terminan comparando el resultado con un umbral que depende del valor medio de la intensidad de la imagen, la media se determina por medio de la ecuación (6).

$$\vec{X} = \frac{1}{n \times m} \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} I(i,j) \tag{6}$$

La varianza calculada (Vc) de la imagen de n x m pixeles se define con la ecuación (*7*):

$$\mathrm{Vc} = \frac{1}{n \times m} \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} \left(I(i,j) - \vec{X}\right)^2 \tag{7}$$

Dónde:

- I (i, j), es la intensidad asociada con el pixel (i, j)
- $\vec{X}$, es la media de la intensidad de la imagen.

Luego de calcular la media y la varianza, se halla para cada pixel la intensidad IC con la ecuación (*8*). Se toma una media deseada de 145 y una varianza de 120:

$$IC = \sqrt{\frac{vd \times \left(I - \vec{X}\right)^2}{Vc}} \tag{8}$$

Dónde:

- Vd, es la varianza deseada
- Md, es la media deseada

La intensidad de cada pixel se modifica en función del resultado de la ecuación (*9*). La Figura 37-3, muestra la Imagen original a la izquierda y a la derecha la imagen normalizada, aquí se puede observar de qué manera el proceso de normalización varía la intensidad de los píxeles en base a los valores de la media y varianza.

$$I(i,j) = \begin{cases} md - \sqrt{\frac{vd \times (I - \vec{X})}{Vc}} & si\ I < \vec{X} \\ md + \sqrt{\frac{vd \times (I - \vec{X})}{Vc}} & si\ I > \vec{X} \end{cases} \tag{9}$$

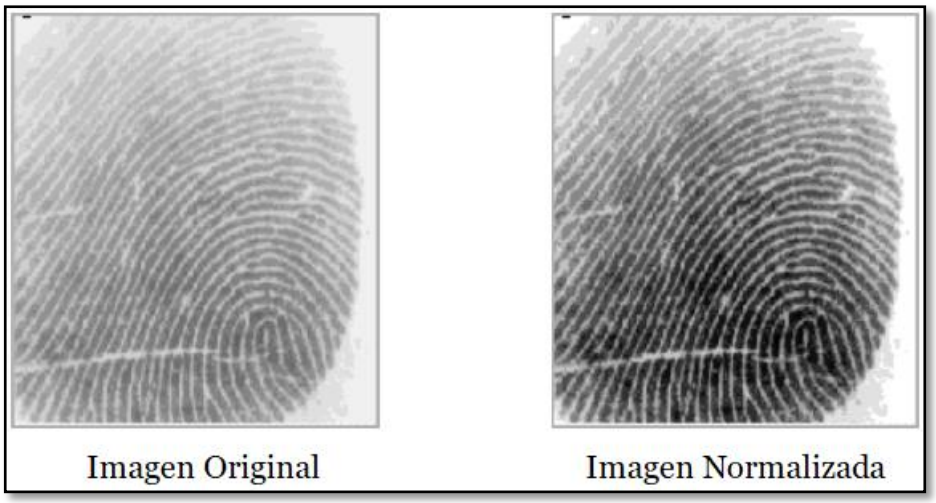


Figura 37-3. Imagen Original y Normalizada.
Realizado por: Salazar Byron, 2019.

3.6.7.6.4. Segmentación

La fase de segmentación consiste en separar el fondo y segmentar el área de interés, donde están contenidas las crestas y valles de la huella dactilar. Para ello se analizan los valores de la varianza local de la imagen I(i, j), se divide en bloques k de N x N, se analiza la varianza de cada bloque y se fija un umbral global, si el valor es menor que este umbral se establece como fondo de la imagen (blanco),de lo contrario se establece como imagen de interés y mantiene el valor de la imagen normalizada.

La varianza para cada bloque k de dimensiones N x N se define como la ecuación (*10*):

$$\mathrm{V}(k) = \frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}(I(i,j) - M(k))^2 \qquad (10)$$

Dónde:

- $\mathrm{V}(k)$, es la varianza del bloque k.
- $I(i,j)$, es la intensidad del pixel.
- $M(k)$, es la media del bloque k.

El fondo representa zonas homogéneas en términos de intensidad de píxel cuya varianza es muy pequeña. Para el cálculo de la varianza local, la imagen se divide en bloques de 16 x 16, de modo que si el valor está por debajo de 100 (umbral tomado de forma empírica) quiere decir que este bloque no contiene huella, por el contrario, si la varianza es mayor o igual a 100 ese valor indica que hay huella, por lo tanto, no debe segmentarse el bloque (López, s.f.). La Figura 38-3, muestra la imagen segmentada.

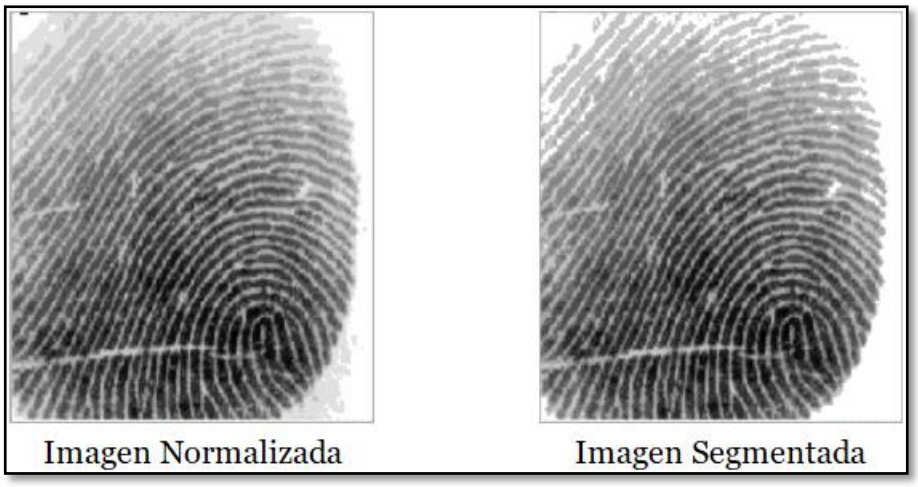


Figura 38-3. Imagen Segmentada.
Realizado por: Salazar Byron, 2019.

3.6.7.6.5. Orientación

Luego de ejecutar el proceso de segmentación, se realiza este proceso cuyo objetivo es determinar la orientación de las crestas (ridges) dentro de la imagen de la huella dactilar. Esto permite establecer la orientación de las minucias. Para calcular la orientación del pixel central (i, j), se siguen los siguientes pasos:

- Primero se calcula para todos los gradientes en X e Y, aplicando las máscaras Gx e Gy que se observan en la Figura 39-3, para ello se realiza la convolución de cada pixel con una máscara de 3 x 3, conocido en procesamiento digital como operador de Sobel, Gx y Gy son mascarilla para detección de bordes horizontal y vertical.

Mascara Gx

-1	0	1
-2	0	2
-1	0	1

Mascara Gy

-1	-2	-1
0	0	0
1	2	1

Figura 39-3. Operador de Sobel Gx e Gy.
Fuente: (López, s.f.)

- Luego de hallar el gradiente en X e Y de cada píxel se procede a calcular la orientación del bloque de acuerdo a su píxel central, con la ecuación (*11*):

$$\theta(i,j) = 0{,}5 \times arctg\left[\frac{V_x(i,j)}{V_y(i,j)}\right] \quad (11)$$

Dónde la ecuación (*12*), corresponde a la covarianza en X y la ecuación (*13*) corresponde a la covarianza en Y:

$$V_x(i,j) = \sum_{u=i-\frac{N}{2}}^{i+\frac{N}{2}} \sum_{v=j-\frac{N}{2}}^{j+\frac{N}{2}} 2G_x(u,v)G_y(u,v) \tag{12}$$

$$V_y(i,j) = \sum_{u=i-\frac{N}{2}}^{i+\frac{N}{2}} \sum_{v=j-\frac{N}{2}}^{j+\frac{N}{2}} \left(G_x^2(u,v)G_y^2(u,v)\right) \tag{13}$$

La orientación calculada con las expresiones (11), (12) y (13) da buenos resultados; Sin embargo, se ha demostrado que existen bloques donde el cálculo de la orientación no es correcto. Conocida la estructura particular de los patrones de huella, es sabido que la direccionalidad de los pixeles no experimenta cambios bruscos entre un bloque y sus vecinos, de modo que el mapa de orientación a nivel local es bastante homogéneo.

Para eliminar errores se utiliza un filtro pasa bajos sobre el resultado obtenido en la expresión (*11*). Para esto se calculan dos matrices, una corresponde a $Sin2\theta$ y la otra a $Cos2\theta$. A cada una de estas matrices se les aplica un filtro pasa bajos con una máscara 3x3; la Figura 40-3, muestra la máscara y la figura filtrada. Finalmente, se calcula el valor de la orientación según la ecuación (*14*).

$$\theta = 0{,}5 \times arctg\frac{Sin2\theta}{Cos2\theta} \tag{14}$$

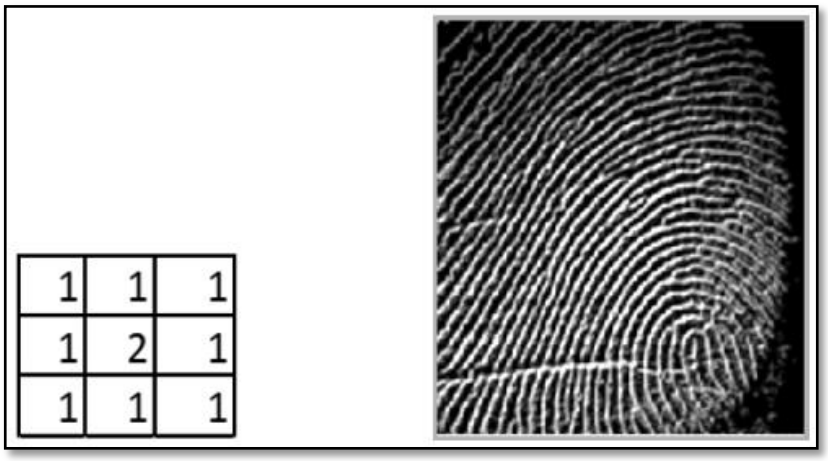


Figura 40-3. Mascara para aplicar un filtro pasa bajos.
Fuente: (López, s.f.)

3.6.7.6.6. Filtrado para el mejoramiento de la imagen

El objetivo de esta etapa es determinar qué píxeles forman parte de los valles (negros) y de las crestas (blancos). En general un filtro bidimensional matemáticamente requiere una señal de entrada y produce una señal de salida, la cual está relacionada con la entrada. La Figura 41-3, muestra las máscaras de filtrado. IMAQ Convolute, Filtra una imagen utilizando un filtro lineal.

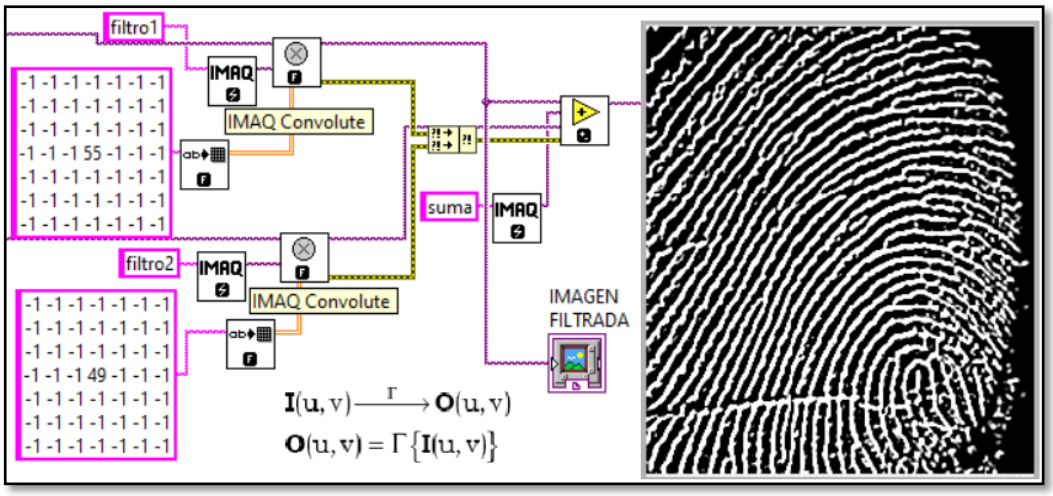


Figura 41-3. Filtros para mejorar la imagen.
Realizado por: Salazar Byron, 2019

3.6.7.6.7. Binarización

Esta etapa busca adecuar la imagen para facilitar el trabajo de reconocimiento, esta adecuación consiste en llevar la imagen de 255 posibles niveles de gris a solo dos posibles niveles de gris (0 y 255) o blanco y negro. Para iniciar este proceso, se verifica cada pixel y se compara con el umbral establecido 200, los datos menores a 200 serán cero y los superiores serán 255, la Figura 42-3, muestra la imagen binarizada.

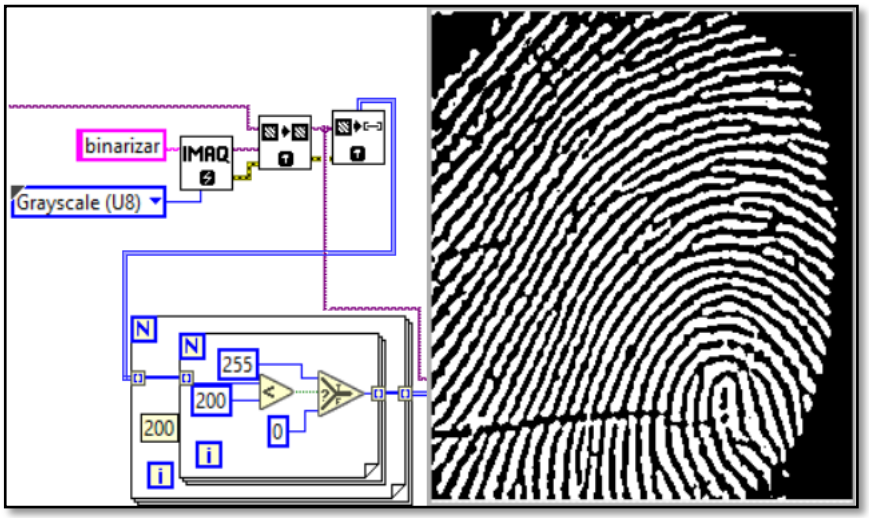


Figura 42-3. Binarización de la imagen
Realizado por: Salazar Byron, 2019

3.6.7.6.8. Adelgazamiento

Antes de la extracción de minucias, se aplica un algoritmo de adelgazamiento o también denominado esqueletización o "thinning", que reduce el ancho de las crestas, hasta que presenten un grosor igual a un pixel, facilitando el proceso de reconocimiento. Se utiliza el algoritmo

propuesto por “Zhang-Suen” en [Zhang 84], donde se define a los ocho vecinos de un pixel. Ver Figura 43-3.

P8 (x-1, y-1)	P1 (x-1, y)	P2 (x-1, y+1)
P7 (x, y-1)	P0 (x, y)	P3 (x, y+1)
P6 (x+1, y-1)	P5 (x+1, y)	P4 (x+1, y+1)

Figura 43-3. Pixel central y definición de sus vecinos
Fuente: (López, s.f.)

Un píxel P0 (x, y) es interno, si cuatro de sus vecinos (x+1,y), (x-1,y), (x,y+1) y (x,y-1) son igual a 0. Un píxel P0 es límite, si no es interno y si solamente uno de sus ocho vecinos es 1. Un píxel es de conexión si al ser eliminado se interrumpe una línea. El algoritmo de adelgazamiento consiste en encontrar píxeles internos en la imagen y después eliminar los píxeles límite. Este proceso se realiza hasta no encontrar más píxeles internos.

Para llevar a cabo el proceso de adelgazamiento, se aplican de forma iterativa dos conjuntos de condiciones en base a la Figura 43-3. Aquellas que cumplen todas las condiciones de la etapa A se cambiarán y se pondrán de color blanco:

Etapa A

- Si se cumple que el número de vecinos distintos de “0” es mayor o igual que dos y menor o igual que seis. (se asegura que los puntos finales se preservan).
- Que solamente una vez se pasa de valor “0” a valor “1” si se recorre el borde, (se preservan los puntos que se encuentran entre ellos).
- Que alguno de “P1”, “P3” y “P5” es un “0”.
- Que alguno de “P3”, “p5” ó “P7” es un “0”.

Una vez cambiados los pixeles que cumplan las condiciones de la “Etapa A”, se cambiarán aquellos que cumplan las cuatro condiciones siguientes de la “Etapa B”.

Etapa B

- Si el número de vecinos distinto de “0” es mayor o igual a “2” y menor o igual a “6”.
- Que solamente una vez se pasa del valor “0” a “1”, si se recorre el borde.
- Que alguno de “P1”,”P3” y “P7” es un “0”.
- Que alguno de “P1”,”P5” ó “P7” es un “0”.

Este algoritmo se realiza hasta que ningún pixel cambie su color de negro a blanco.

El primer paso de este algoritmo consiste en encontrar el total de píxeles internos que existen en la imagen. Después, se eliminan todos los píxeles que están en el límite, verificando que no se trate de un píxel de conexión. Este proceso se repite hasta que no existan más píxeles internos. Después del primer adelgazamiento de la imagen y al no encontrar más píxeles internos, el segundo paso consiste en encontrar únicamente píxeles internos con tres vecinos y después se eliminan los píxeles límite.

La tercera etapa de adelgazamiento consiste en realizar nuevamente la eliminación de píxeles internos. La eliminación de un píxel interno se realiza cuando no es posible eliminar un píxel límite, pero existen aún píxeles internos. El último paso consiste en eliminar píxeles internos que tienen únicamente 2 vecinos y teniendo cuidado de que no se trate de un píxel de conexión. La imagen adelgazada después de N repeticiones se muestra en la Figura 44-3, (Gualberto, Sánchez , Toscano, Nakano, & Pérez, 2008).

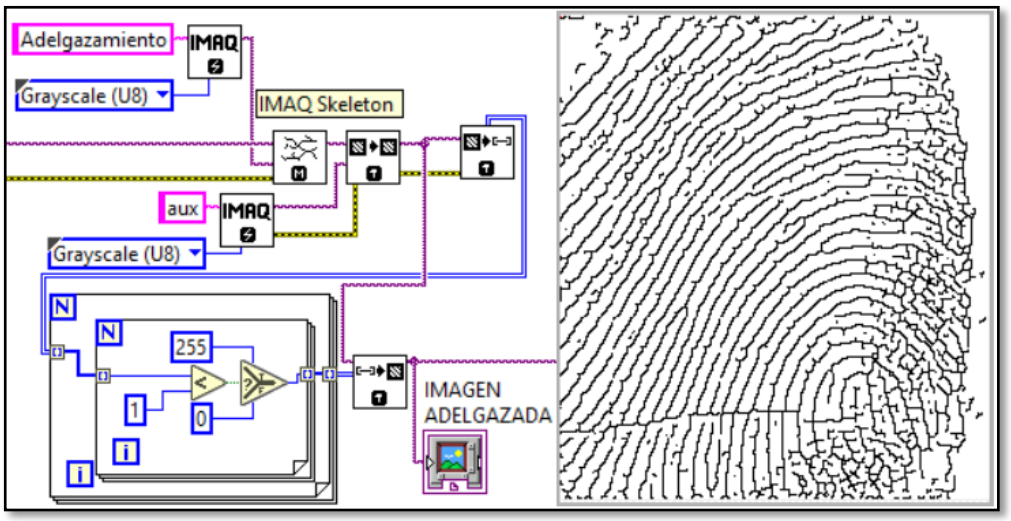


Figura 44-3. Imagen Adelgazada.
Fuente: (López, s.f.)

3.6.7.6.9. Extracción de minucias.

Luego del adelgazamiento, se realiza la búsqueda y detección de minucias. Para la extracción de minucias, se utiliza el algoritmo de Crossing Number que analiza los vecinos de cada píxel que forman parte de las crestas de la huella mediante una ventana de 3 x 3, esta ventana se observa en la **Figura 45-3**, la cual indica el orden de operación para encontrar las minucias de la imagen.

P4	***P3***	***P2***
P5	***Pc***	***P1***
P6	***P7***	***P8***

Figura 45-3. Ventana 3 x 3 para análisis de pixeles.
Realizado por: Salazar Byron, 2019.

Este proceso consiste en realizar operaciones algebraicas con la vecindad del píxel en análisis mediante la Formula de Crossing Number que se observa en la ecuación (*15*).

$$CN = \frac{1}{2}\sum_{i=1}^{8}|P_i - P_{i+1}|\,;\; P9 = P1 \qquad (15)$$

El resultado de aplicar el algoritmo de Crossing Number determina un tipo de minucia que se lo aprecia en la Tabla 35-3:

Tabla 35-3: Tipo de minucias obtenidas con el algoritmo.

Crossing Number	Minucias
1	Terminación
2	Continuación
3	Bifurcación

Fuente: (Edward, 1900)
Realizado por: Salazar Byron, 2019
Nota. Tipos de minucias presentes.

En la Figura 46-3, se muestra la configuración de las ventanas cuando estas simbolizan bifurcaciones.

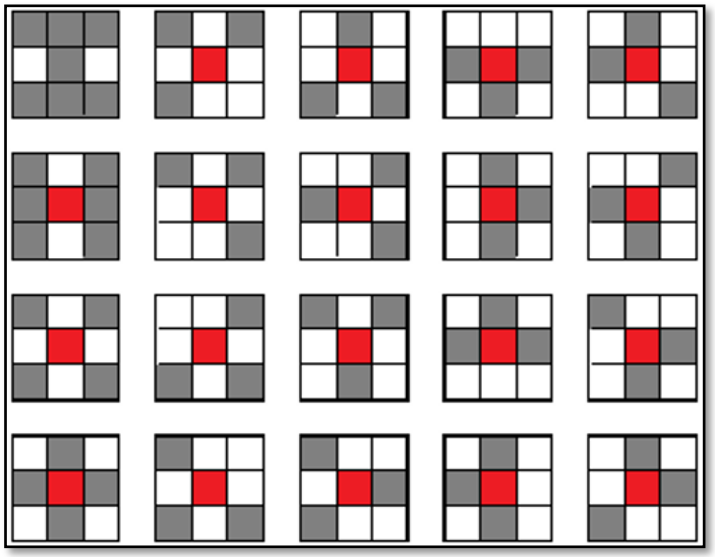


Figura 46-3. Condiciones para detectar bifurcaciones.
Realizado por: Salazar Byron, 2019.

Para el caso de las terminaciones, las consideraciones se resumen en la Figura 47-3, al igual que en las bifurcaciones, el recorrido es pixel a pixel, cuando un pixel cumple alguna de estas consideraciones, se dice que el pixel central o Pc es una terminación.

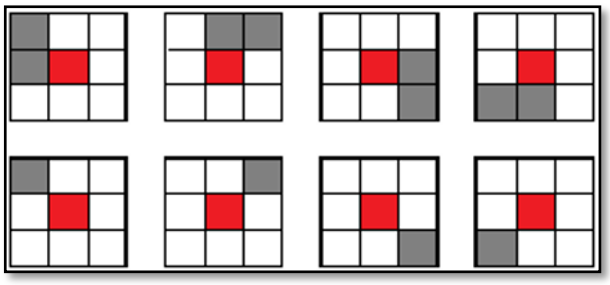


Figura 47-3. Condiciones para detectar terminaciones.
Realizado por: Salazar Byron, 2019.

En resumen, sí:

- Pc=7 pixeles blancos determinan un bloque con terminación
- Pc=6 pixeles blancos determinan un bloque sin minucias
- Pc≤5 pixeles blancos determinan un bloque con bifurcación

Este proceso se realiza en toda la imagen adelgazada aplicando una ventana de 3x3. La Figura 48-3, muestra las minucias localizadas, las terminaciones están señaladas de color rojo y las bifurcaciones están representadas con el color azul, cabe la pena indicar que toda terminación o bifurcación son marcadas; sin embargo, más adelante se eliminan las minucias falsas.

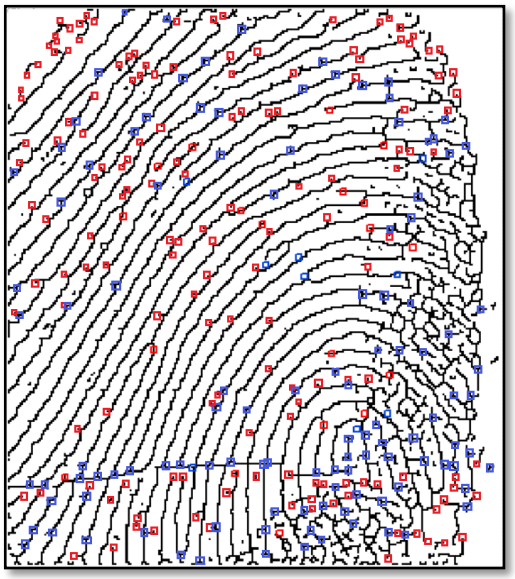


Figura 48-3. Minucias encontradas y marcadas.
Fuente: (López, s.f.)

3.6.7.6.10. Eliminación de minucias falsas

Es muy común encontrar falsas minucias, debido a sobre detecciones. Existen distintas formas que se atribuyen a falsas minucias, estas se resumen en la Figura 49-3:

A. Cuando una línea pierde continuidad cerca de ella, dos terminaciones serán detectadas, con ángulos desfasados ~180 grados.
B. Cuando dos líneas pierden continuidad cerca de ella, cuatro terminaciones serán detectadas, con ángulos desfasados ~180 grados.
C. Cuando una línea que debería llevar un flujo "natural" de dos líneas paralelas, se corta y se une repentinamente con una línea cercana a ella, y el ángulo entre la línea y la bifurcación están desfasado en ~90 grados.
D. Cuando hay dos bifurcaciones muy cercanas, puede ser un caso especial de doble bifurcación, pero si las líneas que forman las bifurcaciones están bastante cerca se trata de dos líneas paralelas. Sus ángulos son desfasados ~180 grados.
E. Cuando una bifurcación unida por una línea a una terminación está muy cerca, se deben eliminar las dos minucias y dejar una solo línea.
F. Cuando dos líneas paralelas son atravesadas por una línea, y se forman dos bifurcaciones con ángulos desfasados paralelos, se elimina la línea que las une.
G. Cuando en una bifurcación las dos líneas que se unen, son atravesadas por otra línea, se debe eliminar esta línea, y dejar una sola bifurcación.
H. Similar al caso F pero con dos líneas

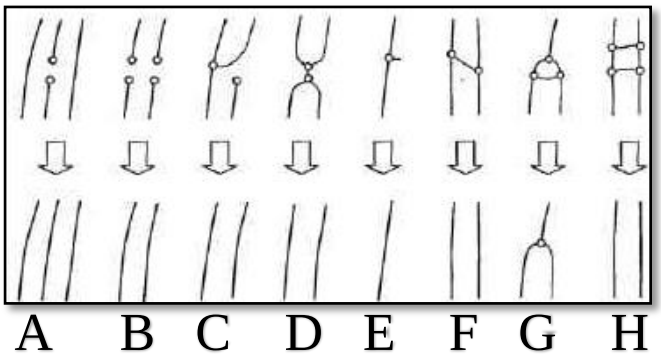


Figura 49-3. Estructuras comunes de falsas minucias.
Fuente: (Adrada, 2010)

En resumen, se mide las distancias entre las minucias y si la separación es menor a 20 pixeles, estas se eliminan ya que son errores que se producen al procesar la imagen, la Figura 50-3, muestra la imagen libre de minucias falsas. Una vez que se eliminan las minucias, se procede a vectorizar la información para almacenar en la base de dato.

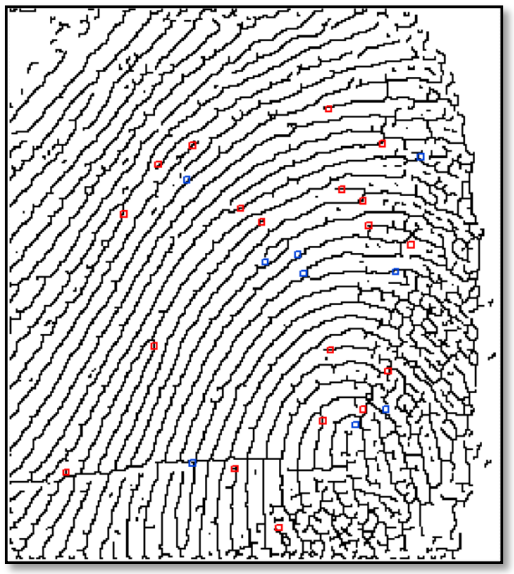


Figura 50-3. Imagen libre de Falsas Minucias.
Realizado por: Salazar Byron, 2019

3.6.7.6.11. Vectorización de las minucias

Después de encontrar todas las minucias presentes en la huella digital cuyos puntos varían de 5 a 50, comienza el proceso de vectorización o armado de trama con los datos de la posición, ángulo y tipo de minucia (x, y, θ, tipo (Terminación=0, Bifurcacion=1)), este vector se almacena en la base de datos conjuntamente con los datos de la persona registrada, para compararse con la plantilla recibida desde el equipo terminal.

3.6.7.7. Etapa de Identificación de Personas.

Esta etapa se encarga de procesar la información recibida desde el equipo terminal para identificar personas. La comunicación se da mediante tramas, estas contienen la plantilla de minucias o los dígitos de la cédula, para que puedan ser comparadas con la información almacenada en la base de datos. La Tabla 36-3, contiene la estructura de la trama enviada desde el equipo terminal, está compuesta por: cabecera de inicio, instrucción, tamaño, datos y el final de trama que indica al programa que comience a operar.

Tabla 36-3: Trama de Datos de entrada a la Estacón Base

Nombre	Bytes	Cédula	Minucias
Cabecera	2	EF 01	EF 01
Instrucción	1	02	01
Longitud de trama	1	0A	C8
Datos	10 ~ 200	Dígitos de la cédula	Rasgos característicos
Fin de trama	3	FF FF FF	FF FF FF

Realizado por: Salazar Byron, 2019
Nota. Trama de datos que contiene la plantilla de minucias y los dígitos de la cédula

A continuación de describe las dos maneras establecidas para identificar personas, la primera es mediante el número de la cédula, este método se lo realiza a través de consultas SQL, la segunda es mediante la plantilla de minucias y para esto se realiza un procesamiento de imágenes para establecer la similitud entre plantillas e identificar a la persona a la que corresponde la imagen.

3.6.7.7.1. Identificación mediante consulta SQL

La trama de datos indica al programa principal que tipo de información posee, mediante el byte de instrucción, ya que, si es 01 en hexadecimal, el programa sabe que la trama está compuesta por una plantilla de minucias, si es 02 el programa extrae los dígitos de la cédula y procede a realizar la consulta de la información mediante el uso de comandos SQL.

La consulta se realiza mediante los siguientes comandos: "SELECT ur. APELLIDOS_ NOMBRES, dp.CEDULA, dp.DIRECCION, dp.FECHA_ NACIMIENTO, dp. TELEFONO, dp. NOMBRE_REFERENCIA, dp.TELEFONO_ REFERENCIA, dm.TIPO _SANGRE, dm. ALERGIAS, dm.ENFERMEDADES, dm. MEDICAMENTOS, ur. IMAGEN1, ur.PLANTILLA1 FROM usuarios_registrados ur INNER JOIN datos_personales dp ON ur.ID_ USUARIOS =dp.ID_ USUARIOS INNER JOIN datos_medicos dm ON dp.ID_ DATOS_ PERSONALES=dm.ID_ DATOS_PERSONALES WHERE dp.CEDULA= 1803725066"

El resultado de una consulta SELECT devuelve una tabla lógica, es decir, los resultados son una relación de datos, que tiene filas o registros, con una serie de campos o columnas. Igual que cualquier tabla de la base de datos, a continuación, se describe los comandos utilizados:

- SELECT: indica las columnas que se van a mostrar y en qué orden.
- FROM: Esta cláusula indica las tablas de las cuales se va a obtener la información.
- INNER JOIN: Combina los registros de una o varias tablas en la base de datos.
- ON: Especificar los campos a través de los cuales se establece la relación de tablas.
- WHERE: Especifica la condición de filtro de las filas devueltas. En este caso se utiliza con el campo Cédula (dp.Cedula=Dígitos de la cédula)

Es necesario estableces una relación interna previamente, para llevar a cabo consultas en diferentes tablas. Una vez que se realiza la consulta y si existe coincidencia con alguna persona registrada, el programa principal obtiene la información, la empaqueta y la envía hacia el equipo terminal para mostrar en pantalla.

3.6.7.7.2. Identificación mediante procesamiento de imágenes

La trama de datos indica al programa principal mediante el byte de instrucción que tipo de información ha recibido. Si la instrucción es 01 en formato hexadecimal, el programa sabe que el paquete está compuesto por minucias y tiene que llevar a cabo el procesamiento de imágenes para

identificar a la persona. Cada minucia está compuesta por 4 bytes, la Tabla 37-3, muestra la estructura de las minucias.

Tabla 37-3: Estructura de las minucias (Tipo, calidad, ángulo, posición).

	Calidad				**Ángulo**								**Y**								**X**							
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Byte 60 (1E Hex)					Byte 59 (19 Hex)									Byte 58 (12 Hex)									Byte 57 (3C Hex)								
E				1				9				1				2				1				C				3			
0	1	1	1	1	0	0	0	1	0	0	1	1	0	0	0	0	1	0	0	1	0	0	0	0	0	1	1	1	1	0	0

Fuente: (Grow, 2015)
Realizado por: Salazar Byron, 2019
Nota. El byte de la posición 0 contiene el tipo de minucia. Terminación=0; Bifurcaciones=1.

Una vez que se extrae las minucias de la trama de datos, el sistema crea una matriz de 4 columnas por N número de filas, las filas dependen de la cantidad de minucias encontradas en la huella digital. La Figura 51-3, muestra el programa utilizado para extraer las minucias y almacenarlas en una matriz, los datos obtenidos sirven para ir comparándolos con cada matriz almacenadas en la base de datos para identificar a la persona a la que le corresponde la huella.

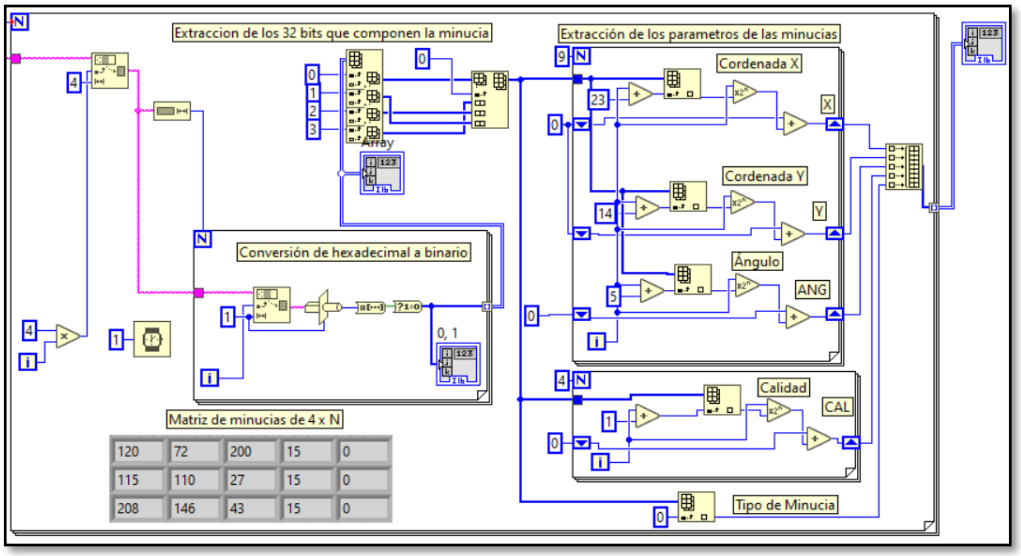


Figura 51-3. Obtención de Matriz de minucias.
Realizado por: Salazar Byron, 2019.

Una vez que se obtienen los rasgos característicos de las huellas digitales, para realizar el proceso de reconocimiento, primero se calcula las distancias de las rectas que se forman entre todas las minucias con la ecuación (*16*), segundo se determina el ángulo que forman dichas rectas con el plano horizontal, para esto se utiliza la ecuación (*17*). M1(x1; x2) y M2(y1; y2) representan los puntos de las minucias.

$$d = \sqrt{(x1 - x2)^2 + (y1 - y2)^2} \quad (16)$$

$$m = tg\theta = \frac{y2-y1}{x2-x1} \quad (17)$$

Una vez que se obtienen las distancias y los ángulos de las plantillas a comparar, primero se localizan las rectas con distancias similares. Aun perteneciendo a la misma huella, las distancias pueden variar ligeramente por lo tanto se ha establecido un rango de tolerancia de 10 pixeles, si sobrepasan este umbral son descartadas (Gualberto, Sánchez , Toscano, Nakano, & Pérez, 2008). La Figura 52-3, muestra la distancia y el ángulo de dos plantillas del mismo dedo.

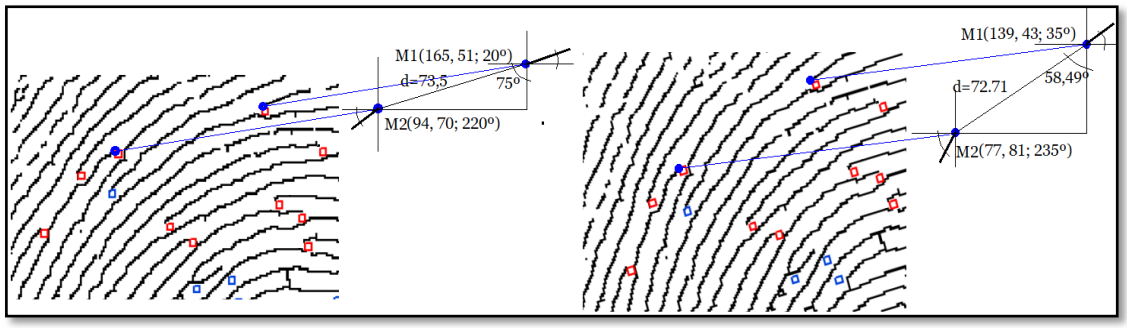


Figura 52-3. Distancias entre plantillas de minucias del mismo dedo.
Fuente: (Gualberto, Sánchez , Toscano, Nakano, & Pérez, 2008)

Cuando se comparan las distancias, estas pueden ser similares aun a pesar de no corresponder a las mismas minucias, esto produce errores en la identificación. Para evitar estos inconvenientes, se calcula el ángulo que se forma entre cada minucia con respecto a la recta con ayuda de la ecuación (*18*), de esta manera se obtienen dos parámetros (distancia y 2 ángulos) para realizar la comparación entre plantillas, la Figura 53-3, muestra los ángulos entre las minucias y la recta.

$$tg\theta = tg(\alpha - \beta) = \left|\frac{tg \propto -tg\beta}{1 + (tg\alpha \times tg\beta)}\right| \qquad (18)$$

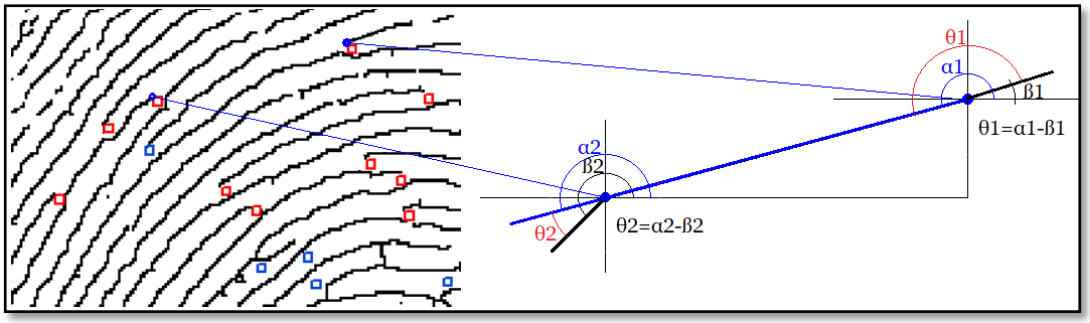


Figura 53-3. Ángulo de Minucias con respecto a la Recta.
Realizado por: Salazar Byron, 2019.

Al momento de tomar una huella digital para comparar con las almacenadas, los rasgos característicos pueden cambiar de posición, pero se mantienen las distancias entre minucias y los ángulos varían de forma proporcional. Por lo tanto, cuando se compara estos parámetros; si son similares, es muy probable que se trate de los mismos puntos de referencia; sin embargo, si algún elemento no coincide, estos son descartados, esto mejora el proceso de verificación.

La Figura 54-3, muestra alta coincidencia entre dos Huellas Digitales, un mayor número de rectas significa que existe mayor porcentaje de coincidencia. Se grafica unicamente las rectas que tienen las mismas distancias y ángulos.

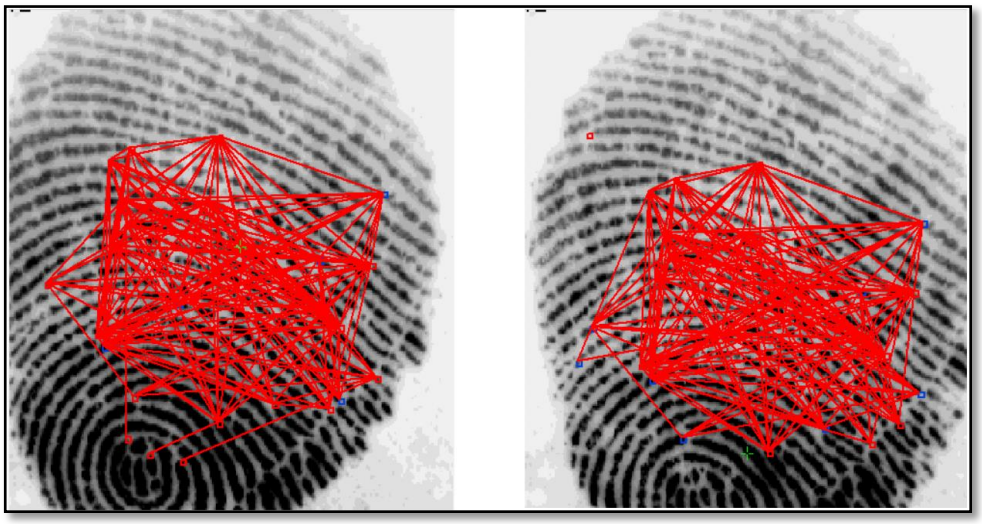


Figura 54-3. Alta coincidencia entre Huellas Digitales.
Realizado por: Salazar Byron, 2019

Un menor número de rectas significan baja coincidencia, como se ve en la Figura 55-3.

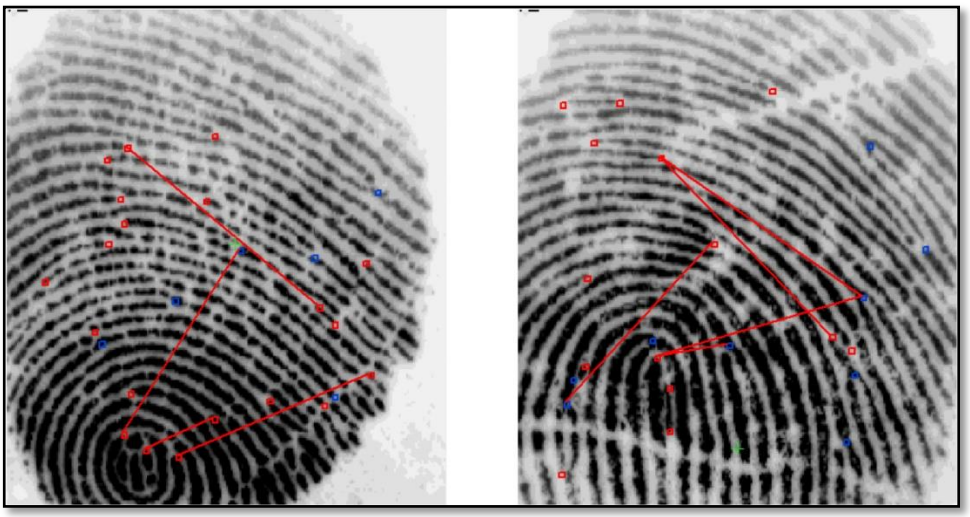


Figura 55-3. Baja coincidencia entre Huellas Digitales.
Realizado por: Salazar Byron, 2019

Las rectas que se obtienen, poseen las mismas distancias y ángulos, es decir son comunes en ambas imágenes. Cada línea representa un vector y a través de la cantidad de líneas que se tiene, se puede determinar el número de rasgos característicos en común que poseen ambas figuras, para esto se utiliza la ecuación (*19*).

$$M = \frac{1+\sqrt{1+8R}}{2} \qquad (19)$$

Dónde:

- M es el número de minucias
- R es el número de rectas

Se estima que, con un umbral mayor de 20 coincidencias en los rasgos característicos, se obtiene un buen reconocimiento; es decir, que la plantilla de entrada será declarada semejante solamente cuando su matriz contenga más de 20 minucias similares con alguna de las plantillas almacenadas en la base de datos; por lo tanto, una vez que se realiza el proceso de verificación, de existir coincidencia, el sistema consulta la información de la persona asociada a la huella digital identificada para empaquetarla y enviarla de forma inalámbrica al equipo terminal

3.6.7.8. Consulta de la información en la Base de Datos

La consulta de la información se realiza a través de comandos SQL "SELECT dp.APELLIDOS, dp.NOMBRES, dp.CEDULA, dp.EDAD, dp.TELEFONO, dp. DIRECCION, dm.TIPO_SANGRE, dm.ALERGIAS,dm.ENFERMEDADES, dm.MEDICAMENTOS FROM identificador id INNER JOIN datos_personales dp ON id.CEDULA= dp.CEDULA INNER JOIN datos_medicos dm ON id.IDENTIFICADOR=dm.IDENTIFICADOR WHERE id.IDENTIFICADOR= Número del identificador al cual corresponde la plantilla con mayor número de coincidencias". Los campos que se obtienen son:

- Apellidos y Nombres
- Cédula
- Edad
- Teléfono de referencia
- Dirección
- Tipo de Sangre
- Alergias
- Enfermedades
- Medicamentos

La Figura 56-3, muestra el diagrama para realizar la consulta en la base de datos, luego de identificar algún individuo. Una vez realizada la consulta se arma la trama de datos para ser enviada hacia el equipo terminal, con el objetivo de mostrar la información de la persona en pantalla como ya se indicó en la Figura 18-3, para que el personal médico tome la mejor decisión en base a su historial médico.

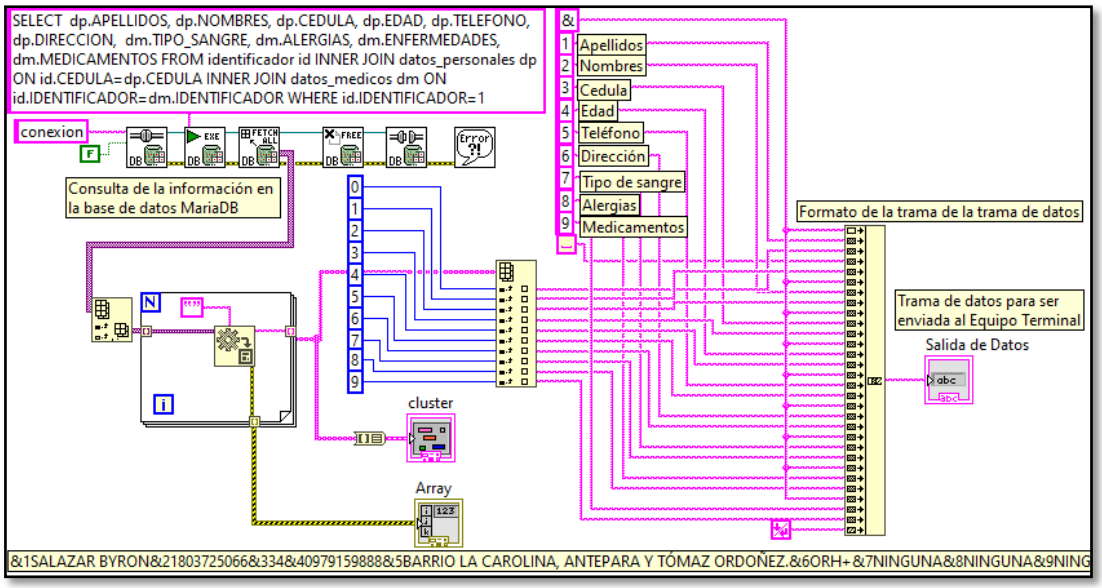


Figura 56-3. Consulta de la Información y armado de la trama de datos.
Realizado por: Salazar Byron, 2019

La trama de datos en la parte inferior de la Figura 56-3, está compuesta por el carácter especial "&" seguido de su número identificador. El carácter "&" y su número identificador precede a cada campo de información consultada, esta estructura indica al equipo terminal cuando comienza un nuevo campo de datos y en qué número de control de texto se debe imprimir en la pantalla ya que es necesario especificar el control en el que se desea colocar la información.

Finalmente, la interfaz gráfica también muestra en pantalla la información consultada de la persona identificada, esto es con el fin de informar al usuario del sistema que se ha llevado a cabo una nueva consulta. La Figura 57-3, muestra la pantalla principal de la interfaz gráfica donde se monitorea constantemente el funcionamiento del sistema biométrico de transmisión inalámbrica.

Figura 57-3. Pantalla Principal de la Interfaz Gráfica.
Realizado por: Salazar Byron, 2019

CAPÍTULO IV

4. ANÁLISIS Y RESULTADOS

Una vez implementado el sistema biométrico de transmisión inalámbrica, se realizó las pruebas de campo con el personal que opera las unidades móviles del ministerio de salud. Las pruebas realizadas permiten obtener los resultados del rendimiento del sistema, al calcular las tasas de falsa aceptación, falso rechazo y la tasa de error, estos resultados se obtienen en base al número de pruebas realizadas. El ANEXO 4 contiene imágenes de las pruebas realizadas con el personal de la salud.

4.1. Medidas de Errores en los Sistemas de Verificación

Para la evaluación del rendimiento de un sistema de verificación de huellas, se puede formular lo siguiente; Llámese a una huella de un nuevo usuario como S (Sospechosa) y la huella almacenada en los registros como C (Conocida), entonces la hipótesis nula y alternadas son:

$H_0: C \neq S$, La huella de entrada NO es la misma que la huella almacenada.
$H_1: C = S$, La huella de entrada SI es la misma que la huella almacenada.

Las decisiones asociadas a estos dos casos son:

$D_0: C \neq S$, La persona no está registrada
$D_1: C = S$, La persona es genuina.

Una hipótesis que pruebe la formulación inherente contiene dos tipos de errores:

Tipo I: Tasa de falsa Aceptación o False Accept Rate (FAR) por sus siglas en inglés.
Tipo II: Tasa de falso Rechazo o False Reject Rate (FRR) por sus siglas en inglés.

4.1.1. Tasa de Falsa Aceptación FAR

También llamado FMR (False Match Rate), Se presenta cuando se realiza la comparación entre huellas digitales y a pesar de no corresponder a la misma imagen el sistema las califica como similares. Este error se calcula como se ve en la ecuación (20), donde N_{FA} es el número de falsas aceptaciones y T_I es el número total de pruebas realizadas. (Adrada, 2010):

$$FAR = \frac{N_{FA}}{T_I} \tag{20}$$

$$FAR = \frac{0}{120} = 0$$

Este resultado determina que el porcentaje de falsa aceptación es nulo ya que, de ciento veinte (120) intentos de identificación, cero dieron falso positivo.

4.1.2. Tasa de Falso Rechazo FRR

También llamado FNMR (False Non-Match Rate), se presenta cuando un usuario SI autorizado es rechazado por el sistema como un usuario NO autorizado. Se calcula con la ecuación (21), donde N_{FR} es el número de falsos rechazos y T_I es el número total de intentos de identificación (Adrada, 2010):

$$FRR = \frac{N_{FR}}{T_I} \tag{21}$$

$$FRR = \frac{2}{120} = 0{,}017 = 1{,}7\%$$

Este resultado determina que el porcentaje de falsa aceptación es bajo, ya que, de ciento veinte 120 intentos de identificación, dos 2 dieron falso negativo.

Las gráficas de FAR y FRR se observan en la Figura 1-4. Los valores de FAR y FRR disminuyen y aumentan (respectivamente) a medida que el umbral cambia. El umbral está determinado por el número de minucias emparejadas como se observa en la ecuación (22). Con el umbral se puede ajustar la sensibilidad del sistema para que compare y reconozca una huella (Adrada, 2010).

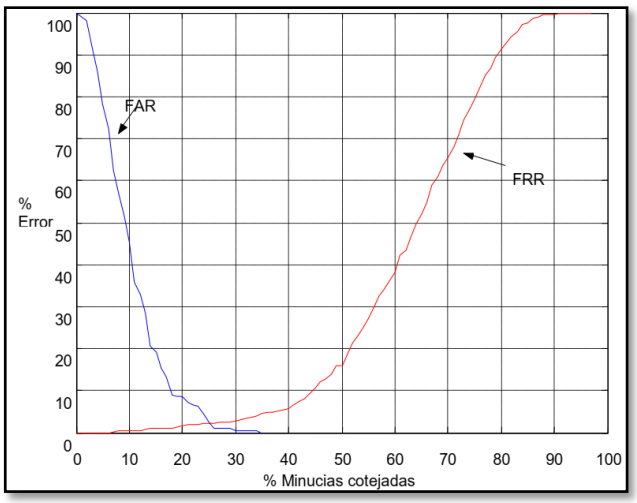


Figura 1-4. (FAR) y (FRR) en funciones del umbral de aceptación (u).
Fuente: (Adrada, 2010).

$$u = \frac{Número\ de\ minucias\ emparejadas}{Número\ de\ minucias\ totales} \tag{22}$$

$$u = \frac{20}{50} = 0{,}4 = 40\%$$

El valor establecido del umbral es de 0.4 lo que equivale a decir, que con el cuarenta por ciento de coincidencia entre plantillas se considera que corresponde a la misma huella digital. Mientras más bajo es el umbral se puede dar un mayor número de falsas aceptaciones y mientras más alto es el umbral puede existir un mayor número de falso rechazos.

El punto de intersección de la FRR y FAR está denotado por u*, en este punto se encuentran el mínimo de la falsa aceptación y rechazó. Este valor se lo conoce como tasa de error de intersección (cross-over error rate), se puede utilizar como medida para identificar el grado de seguridad de un sistema biométrico. Cuando se utiliza el programa en un entorno de máxima seguridad, es necesario que el FAR sea el más pequeño posible.

4.1.3. Tasa de Resultado (SR)

La tasa de resultado o Success Rate, es la combinación FAR y FRR, se utiliza como indicador de la resolución total del sistema (Burga, 2007). La tasa de resultado se calcula con la fórmula de la ecuación (*23*).

$$SR = 1 - (FAR + FRR) \tag{23}$$

$$SR = 1 - (0 + 0{,}017) = 0{,}983$$

La tasa de resultado es del noventa y ocho punto tres por ciento (98,3%), lo que indica que el sistema es confiable, sin embargo, factores ajenos al programa como: dedo lastimado, impurezas en la pantalla del sensor, reducen su eficiencia.

4.1.4. Tasa de Error (EER)

También llamado Equal Error Rate, es un parámetro o umbral que permite igualar los dos factores FAR y FRR debido a que responde a parámetros inversamente proporcionales, por lo tanto, varían de acuerdo a condiciones prefijadas por el programa del dispositivo biométrico. Posee una escala normalizada que va de 0 a 1, el mismo que determinará el poder de identificación del sistema, el error se calcula con la fórmula de la ecuación (24).

$$ERR = \frac{(FAR + FRR) \times 100}{Total\ de\ comparaciones} \tag{24}$$

$$ERR = \frac{(0 + 0{,}017) \times 100}{120} = 0{,}014 = 1{,}4\%$$

La tasa de error del sistema es muy baja, con uno punto cuatro por ciento (1,4%), por tal motivo se puede concluir que el sistema es bastante confiable para llevar a cabo el proceso de reconocimiento de personas utilizando su huella digital. Finalmente, a través del error se calcula la eficiencia del sistema con ayuda de la ecuación (*25*), que no es nada más que el complemento del error.

$$EFICIENCIA = 1 - \%\ ERR \tag{25}$$

$$EFICIENCIA = 1 - 0{,}014 = 0{,}986 = 98{,}6\%$$

4.2. Resultados obtenidos de la encuesta

Para realizar la valoración de los datos obtenidos en la investigación, se aplicó la estadística descriptiva con cuadros y gráficos que se respalda con su respectivo análisis e interpretación, destacando tendencias o relaciones fundamentales de acuerdo a los objetivos. Tabulando las respuestas de los 100 participantes de la encuesta se puede obtener los siguientes resultados.

UNIVERSIDAD POLITÉCNICA DE CHIMBORAZO

ENCUESTA PREVIA AL DESARROLLO DEL "SISTEMA BIOMÉTRICO DE TRANSMISIÓN INALÁMBRICA PARA EL RECONOCIMIENTO DE PERSONAS A TRAVÉS DE SU HISTORIAL MÉDICO"

Encuesta dirigida al personal encargado de las unidades móviles del Ministerio de salud

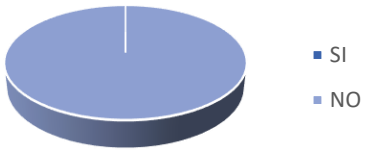
Pregunta 1.- ¿Existe algún sistema biométrico de identificación de personas?

( ) Si ( ) No

Tabla 1-4: Análisis e interpretación, pregunta 1.

Análisis **Interpretación**

Opciones	Frecuencia	%
SI	0	0%
NO	100	100%
TOTAL	100	100%



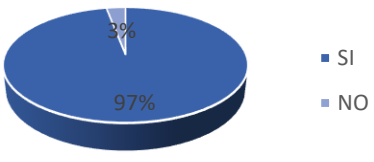
El cien por ciento (100%) de personas respondieron que no existe dispositivos para ayudar a identificar a las personas al momento de producirse un accidente.

Pregunta 2.- ¿Al identificar a una persona y conocer sus antecedentes médicos, la atención es más eficiente? ( ) Si ( ) No

Tabla 2-4: Análisis e interpretación, pregunta 2

Análisis **Interpretación**

Opciones	Frecuencia	%
SI	97	97%
NO	3	3%
TOTAL	100	100%



El noventa y siete por ciento (97%) de encuestados aseguran que la atención sería más eficiente al identificar a la persona y conocer su historial médico.

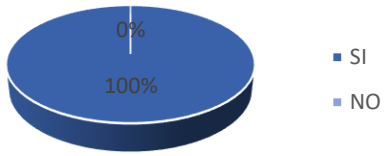
Pregunta 3.- ¿Existen consecuencias negativas en caso de que el sistema biométrico falle?

( ) Si ( ) No

Tabla 3-4: Análisis e interpretación, pregunta 3

Análisis	Interpretación

Opciones	Frecuencia	%
SI	100	100%
NO	0	0%
TOTAL	100	100%



El cien por ciento (100%) de encuestados aseguran que existirían consecuencias desastrosas en caso de que el sistema falle; por lo tanto, el sistema biométrico basado en el Rostro no es fiable ya que existen personas parecidas o cambia con el tiempo.

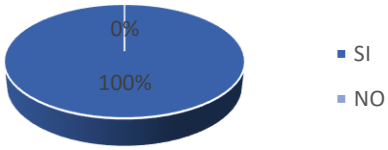
Pregunta 4.- ¿La implementación del sistema biométrico ayudará a tomar mejores decisiones?

( ) Si ( ) No

Tabla 4-4: Análisis e interpretación, pregunta 4

Análisis	Interpretación

Opciones	Frecuencia	%
SI	100	100%
NO	0	0%
TOTAL	100	100%



El cien por ciento (100%) de personas aseguran que el sistema les ayudaría a tomar mejores decisiones, ya sea para tratar medicamente a un paciente o saber a dónde trasladarlo ya sea al Seguro o al IESS.

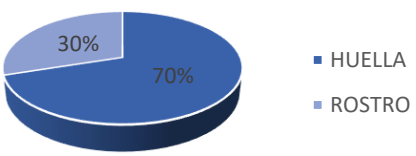
Pregunta 5.- ¿Qué sistema biométrico es más común para identificar a una persona?

( ) Huella Digital ( ) Rostro

Tabla 5-4: Análisis e interpretación, pregunta 5

Análisis	Interpretación

Opciones	Frecuencia	%
HUELLA	70	70%
ROSTRO	30	30%
TOTAL	100	100%



El setenta por ciento (70%) de personas aseguran que la huella digital es más común y más eficiente que los sistemas basados en el rostro ya que existe personas similares o que con el pasar de los años cambian su fisionomía.

4.3. Comprobación de la Hipótesis

Para la verificación de la hipótesis se utiliza el estadístico de contraste, que partiendo de la información recolectada de la población que maneja las unidades móviles del ministerio de salud, permite obtener una conclusión en base a la estimación para aceptar o rechazar la hipótesis planteada. Para lo cual se plantea una hipótesis negativa o nula (H0) y una hipótesis positiva o alternativa (H1).

H0: El sistema biométrico permite identificar y reconocer personas a través de su huella digital o número de cédula y no hace más eficiente la atención médica.

H1: El sistema biométrico permite identificar y reconocer personas a través de su huella digital o número de cédula y hace más eficiente la atención médica.

4.3.1. Nivel de Significación

El nivel de confianza escogido para la investigación es del 5% o $\propto = 0{,}05$ (95%).

4.3.2. Elección de la prueba estadística

Con la información obtenida a través de las encuestas realizadas al personal encargado de las unidades móviles, se efectúa la verificación de la hipótesis planteada, utilizando la prueba del Chi-Cuadrado, con la ecuación (*26*):

$$x^2 = \sum \frac{(O_i - E_i)^2}{E_i} \tag{26}$$

Dónde:

- $x^2 =$ Chi Cuadrado
- $o_i =$ Datos observados (Encuestas)
- $E_i =$Datos esperados (Observación)

4.3.3. Zona de aceptación y rechazo (Grados de libertad)

Los grados de libertad se calcula con la ecuación (*27*).

$$gl = (f - 1) \times (c - 1) = (4 - 1) \times (2 - 1) = 3 \tag{27}$$

Dónde:

- gl= Grados de libertad
- f = filas
- c = columnas

4.3.4. Frecuencias Observadas

Estos datos se obtuvieron de las encuestas realizadas a las personas que se les realizó las pruebas de reconocimiento de las huellas digitales.

Tabla 6-4: Frecuencias Observadas, resultados de la encuesta

Parámetros	Opciones		**Total**
	SI	NO	
Pregunta 2	97	3	100
Pregunta 3	100	0	100
Pregunta 4	100	0	100
Reconocimiento	118	2	120
Total	415	5	420

Fuente: Tabulación de las encuestas realizadas
Realizado por: Salazar Byron, 2019

4.3.5. Frecuencias Esperadas

Para calcular la frecuencia esperada se utiliza la ecuación (*28*):

$$E_i = \frac{(total\ de\ fila \times total\ de\ columna)}{valor\ total} \tag{28}$$

Tabla 7-4: Frecuencias Esperadas

Parámetros	Opciones		**Total**
	SI	NO	
Pregunta 2	98,8	1,19	100
Pregunta 3	98,8	1,19	100
Pregunta 4	98,8	1,19	100
Reconocimiento	98,8	1,19	120
Total	415	5	420

Fuente: Datos calculados a partir de las encuestas
Realizado por: Salazar Byron, 2019

4.3.6. Chi Cuadrado

Tabla 8-4: Cálculo de Chi Cuadrado

$x^2 = \sum \frac{(O_i - E_i)^2}{E_i}$	O_i	E_i	$O_i - E_i$	$(O_i - E_i)^2$	$\frac{(O_i - E_i)^2}{E_i}$
Pregunta 2 (SI)	97	98,8	-1,80	3,24	0,03
Pregunta 2 (NO)	3	1,19	1,81	3,28	2,75
Pregunta 3 (SI)	100	98,8	1,20	1,44	0,01
Pregunta 3 (NO)	0	1,19	-1,19	1,42	1,19
Pregunta 4 (SI)	100	98,8	1,20	1,44	0,01
Pregunta 4 (NO)	0	1,19	-1,19	1,42	1,19
Reconoció (SI)	118	98,8	19,20	368,64	3,73
Reconoció (NO)	2	1,19	0,81	0,66	0,55
				$\boldsymbol{x^2 =}$	**9,48**

Realizado por: Salazar Byron, 2019

Utilizando el grado de libertad (gl=3) y el margen de error (0,05) se pudo obtener el Chi cuadrado mediante la tabla de distribución, de esta manera se puede realizar la comprobación de la hipótesis mediante los siguientes parámetros:

1. $X^2_{Calc} < X^2_{Tabla} \rightarrow H0$
2. $X^2_{Calc} > X^2_{Tabla} \rightarrow H1$

Tabla 9-4: Distribución Chi Cuadrado

n/p	0,001	0,0025	0,005	0,01	0,025	0,05	0,1	0,15	0,2
1	10,8274	9,1404	7,8794	6,6349	5,0239	3,8415	2,7055	2,0722	1,6424
2	13,8150	11,9827	10,5965	9,2104	7,3778	5,9915	4,6052	3,7942	3,2189
3	16,2660	14,3202	12,8381	11,3449	9,3484	7,8147	6,2514	5,3170	4,6416

Fuente: http://labrad.fisica.edu.uy/docs/tabla_chi_cuadrado.pdf
Realizado por: Salazar Byron, 2019

- $X^2_{Calc} = 9{,}48$
- $X^2_{Tabla} = 7{,}81$

$$X^2_{Calc} = 9{,}48 > X^2_{Tabla} = 7{,}81 \rightarrow H1$$

Por lo tanto, se acepta la hipótesis alternativa (H1). "El sistema biométrico permite identificar y reconocer personas a través de su huella digital o número de cédula y hace más eficiente la atención médica". La Figura 2-4, muestra la curva de Chi Cuadrado que demuestra la hipótesis.

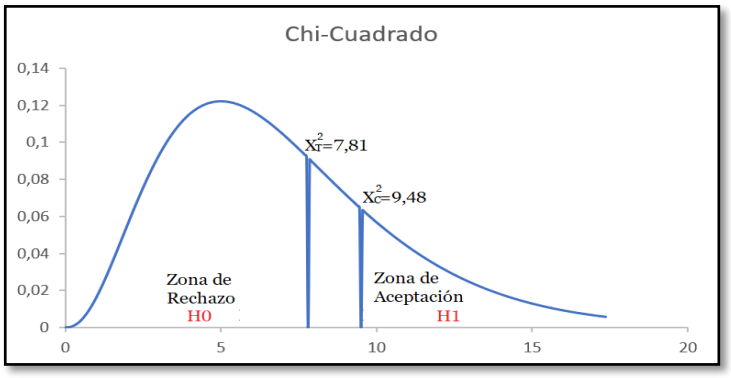


Figura 2-4. Curva de Chi Cuadrado para comprobación de hipótesis.
Realizado por: Salazar Byron, 2019

CAPÍTULO V

5. PROPUESTA

En base al problema de identificación de personas accidentadas en el sistema de salud, se ha propuesto implementar un prototipo de identificación de personas mediante su huella digital o número de cédula, esto permite tomar mejores decisiones para tratar a un herido en base a su historial médico y evitar que se produzcan accidentes al suministrar algún tratamiento que podría poner en peligro su integridad física.

5.1. Hipótesis

El sistema biométrico de transmisión inalámbrica permite la identificación y reconocimiento de personas a través de su huella digital o número de cédula y hace más eficiente la atención médica.

5.2. Sistema Biométrico

La Figura 1-5, muestra un esquema del funcionamiento básico del sistema de identificación biométrica, utilizando la huella dactilar. Para esto se necesita un lector biométrico, que adquiere o captura la imagen de la huella del individuo y la convierte en formato digital. A continuación, el módulo de inscripción aplica los algoritmos necesarios para la extracción de sus características y las almacena en una base de datos.

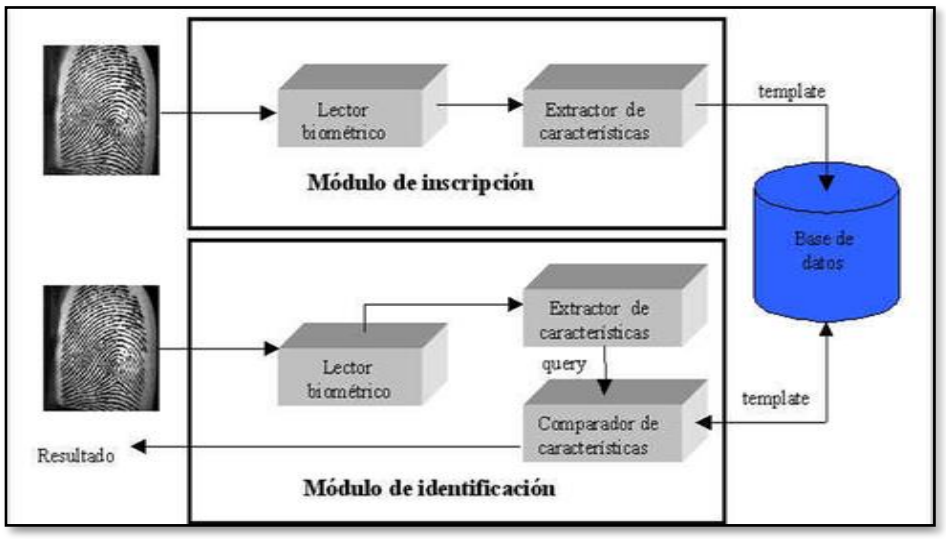


Figura 1-5. Diagrama en Bloques del Sistema Biométrico.
Fuente: (González, 2013)

El módulo de identificación, luego de extraer las características, las compara con las que se encuentran en la base de datos para buscar e identificar el usuario al que le corresponde la huella digital (González, 2013).

5.3. Arquitectura

La Figura 2-5, muestra la Arquitectura del sistema biométrico, donde se encuentran conectados de forma secuencial los módulos electrónicos para escanear la huella digital o ingresar el número de cédula, enviar de forma inalámbrica la información, procesarla y compararla con los datos almacenados para identificar personas. Finalmente, el sistema muestra la información consultada, en una pantalla nextión de 4,3 pulgadas.

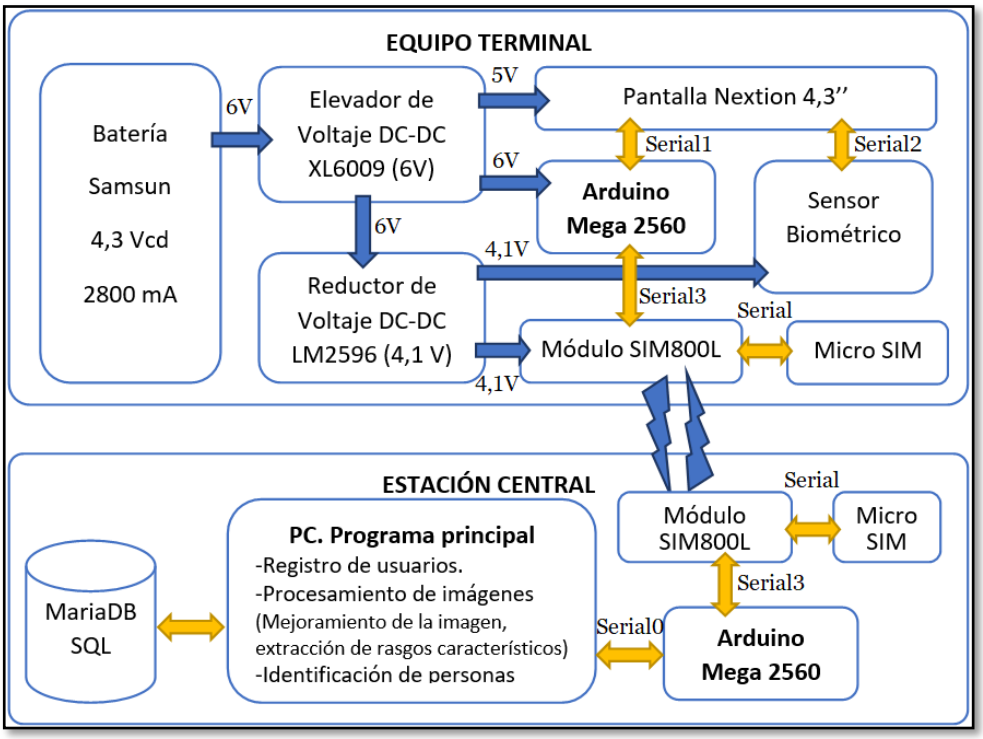


Figura 2-5. Arquitectura del Sistema Biométrico de Acceso Inalámbrico.
Realizado por: Salazar Byron, 2019

5.4. Objetivos

5.4.1. Objetivo General

Implementar un sistema biométrico de transmisión inalámbrica para la identificación y reconocimiento de personas en el Hospital Regional Docente Ambato

5.4.2. Objetivos Específicos.

- Implementar un sistema biométrico para la identificación y reconocimiento de personas.
- Implementar una base de datos para almacenar la información médica de las personas
- Probar el sistema con el personal médico para analizar la eficiencia del Sistema Biométrico.

5.5. Métodos de Investigación

Se utilizó el método experimental y de campo porque se implementó el sistema y se llevó a cabo pruebas de funcionamiento con el personal médico encargado de las unidades móviles del Hospital Regional Docente Ambato, Centro de Salud Número uno y dos, centro de salud de Totoras y Centro de Salud de Izamba., para verificar el cumplimiento de los objetivos establecidos en la propuesta. Una vez realizadas las pruebas y mediante el análisis de los resultados obtenidos a través de las encuestas se establece los resultados que se observa en la Tabla 1-5.

Tabla 1-5: Medida de la eficiencia y funcionamiento del Sistema Biométrico

DESCRIPCIÓN	RESULTADO
TOTAL, DE HUELLAS	120
TASA DE FALSA ACEPTACIÓN	0 %
TASA DE FALSO RECHAZOS	1,7 %
TASA DE RESULTADO	98,3 %
TASA DE ERROR	1,7 %
EFICIENCIA	98,6 %
UMBRAL DE COINCIDENCIA ENTRE PLANTILLAS	40 %
TIEMPO DE CONSULTA DE USUARIO POR CÉDULA	8 segundos
TIEMPO DE CONSULTA DE USUARIO POR HUELLA	15 segundos
AUTONOMÍA DE LA BATERÍA DEL EQUIPO TERMINAL	5 horas

Realizado por: Salazar Byron, 2019
Nota. Medida de la eficiencia y funcionamiento del Sistema Biométrico

PRESUPUESTO

Detalle del producto	Cantidad	Costo Unitario	Costo Total
Bienes			
Elevador de voltaje XL6009	1	3$	3$
Elevador de voltaje MT3608	1	3$	3$
Reductor de voltaje LM2596	2	3$	6$
Módulo inalámbrico SIM800L	2	12$	24$
Arduino Mega 2560	2	10$	20$
Pantalla Nextion 4,3 pulgadas	1	65$	65$
Impresiones 3D	9 piezas		20$
Sensor biométrico	1	40$	40$
Computador	1	1200$	1200$
Recursos			
Materiales Bibliográficos			200$
Humanos			500$
Extras			
Marcador indeleble	1	3$	3$
Baquelita	2	2$	4$
Cloruro férrico	2	0,5$	1$
Internet			50$
Impresiones			100$
Viáticos			100$
Fotocopias			50$
Total			**2389**

Fuente: Mercado Libre Ecuador,2019
Realizado por: Salazar Byron, 2019

CONCLUSIONES

- Al utilizar módulos electrónicos más eficientes para implementar el sistema biométrico y colocarlos en estado de reposo cuando el equipo terminal no está en uso, se disminuye el consumo de corriente y aumenta la autonomía del sistema.

- La imagen borrosa de una huella digital, reduce la eficiencia del sistema biométrico, por lo tanto, limpiar el escáner y colocar correctamente el dedo permite obtener una imagen de alta calidad, esto aumenta la probabilidad de identificar alguna persona siempre y cuando haya sido registrada.

- Bajas Tasas de Falsa Aceptación y altas Tasas de Falso Rechazo, dan como resultado sistemas biométricos más robustos, por lo que, pueden ser utilizados para aplicaciones médicas o en situaciones que se necesite mayor seguridad.

- Mientras más imágenes se utilice para generar una plantilla de rasgos característicos, más eficiente se vuelve el sistema biométrico, lo que ocasiona un incremento en el tiempo de procesamiento de imágenes para identificar personas.

- La tasa de error del sistema es muy baja con un (1,7%), por tal motivo se puede concluir que el sistema es bastante confiable para llevar a cabo el proceso de reconocimiento de personas utilizando huellas digitales para en el área de la salud.

RECOMENDACIONES

- Una imagen clara aumenta la probabilidad de identificación de las personas mediante procesamiento digital de imágenes, por lo tanto, se recomienda limpiar la pantalla del sensor biométrico y que los dedos estén limpios para obtener una imagen de calidad.

- El bajo consumo de corriente aumenta la autonomía del sistema, por lo tanto, es recomendable suspender la pantalla Nextion y mantener el módulo inalámbrico en modo reposo cuando el equipo terminal no este operando.

- La información se encuentra almacenada en un servidor virtual, así que se recomienda levantar los servicios de Apache y MariaDB para tener acceso a la base de datos caso contrario no se puede llevar a cabo el reconocimiento de personas.

- Es recomendable disminuir el espacio de análisis de la huella digital, y enfocarse más en el centro ya que es ahí donde se encuentra el mayor número de rasgos característicos.

- En ocasiones las unidades móviles se desplazan largas distancias, por lo que se recomienda utilizar la red de telefonía móvil para enviar la información vía inalámbrica en vez de los módulos de radio frecuencia ya que su distancia es limitada.

BIBLIOGRAFÍA

Admired. (s.f.). *M2MSupport.net.* Obtenido de https://m2msupport.net/m2msupport/atcsq-signal-quality/

Adrada, J. (2010). *comparación de huellas dactilares utilizando c++*. Obtenido de https://www.academia.edu/8384429/comparaci%C3%B3n_de_huellas_dactilares_utilizando_c_

Alcívar y Yapur. (sf). Desarrollo de un sistema alternativo para la implementacion de Telemedicina mediante internet inalambrico.

AppServ. (2018). *AppServ : Apache + PHP + MYSQL*. Obtenido de http://www.dipler.org/2008/10/que-es-appserv/

banggood.com. (s.f.). Obtenido de https://es.banggood.com/2_8-Inch-Nextion-HMI-Intelligent-Smart-USART-UART-Serial-Touch-TFT-LCD-Screen-Module-p-1129487.html?cur_warehouse=CN

Barbazán, P., & Candela , C. (30 de 11 de 2017). *SlideShare*. Obtenido de https://es.slideshare.net/ChzAlbert/huella-digital-83027614

Bernardo, H. (2018). *INFOOTEC.NET*. Obtenido de https://www.infootec.net/arduino-mega-2560/

Blasco. (05 de 09 de 2016). *BBC News*. Obtenido de https://www.bbc.com/mundo/noticias-37247130#:~:text=GPRS%20significa%20General%20Packet%20Radio,conectados%22%2C%20entre%20otras%20cosas.

Botella, I. (2017). *DocPlayer.* Obtenido de https://docplayer.es/51044533-Capitulo-2-el-sistema-gsm.html

Burga, A. (2007). *Diseño e Implementación de un Sistema de Verificación Mediante Huella Dactilar*. Quito.

Carnicero. (2011). Desarrollo de la eSalud en Europa. .

Carpeta de Salud. (2018). *La 'Carpeta de Salud' Online permite acceder a todo el historial médico de forma sencilla.*

Conel, S. (27 de 06 de 2012). *AT commands.* Obtenido de https://www.induo.com/wp-content/uploads/2014/03/at-kommandon-conel-router.pdf

Crespo. (2018). *Aprendiendo Arduino.* Obtenido de https://aprendiendoarduino.wordpress.com/2016/09/25/que-es-arduino/

Crespo, E. (21 de 01 de 2017). *Aprendiendo Arduino*. Obtenido de https://aprendiendoarduino.wordpress.com/tag/software/

Dailly, N., & Collet, P. (11 de 2018). *Red GSM*. Obtenido de https://www.researchgate.net/figure/Architecture-du-reseau-GPRS-E-GPRS_fig2_328783680

Descubrearduino. (2014). Obtenido de https://descubrearduino.com/contacta-con-nosotros/

Econectia. (05 de 12 de 2018). *Econectia*. Obtenido de https://www.econectia.com/blog/tipos-redes-inalambricas

Edward, H. (1900). *Puntos característicos de las huellas digitales*. Obtenido de https://www.estudiocriminal.eu/blog/puntos-caracteristicos-de-las-huellas-digitales/

Escalona, M., & Paradels, B. (2002). *Delivery of NonStandardized Assistance Data in E-Otd/Gnss Hybrid Location Systems.* Barcelona.

González, J. (04 de 02 de 2013). SISTEMA DE IDENTIFICACIÓN BIOMETRICA. Leganés.

Gualberto, A., Sánchez , G., Toscano, K., Nakano, M., & Pérez, H. (2008). *Reconocimiento de Huellas Dactilares Usando.* Antioquia.

Hesse, H. F. (2010). Social participation in health 2.0. Computer.

Lara, E. (17 de 11 de 2015). *Hetpro*. Obtenido de Pantalla Nextion NX3224T028: https://hetpro-store.com/TUTORIALES/pantalla-nextion-arduino/

Llamas, L. (23 de 10 de 2014). *Ingeniería, informática y diseño.* Obtenido de https://www.luisllamas.es/entradas-analogicas-en-arduino/

López, J. (s.f.). *SISTEMAS DE IDENTIFICACIÓN PERSONAL Y BIOMETRÍA*. Obtenido de https://upcommons.upc.edu/bitstream/handle/2099.1/8082/proyecto%20final%20de%20carrera.pdf?sequence=1

Maugard, J. (17 de 12 de 2015). *Kill My Bill*. Obtenido de https://www.killmybill.es/tarjeta-sim/

NationalInstrument. (2014). *Entorno NI LabVIEW*. Obtenido de https://www.ni.com/academic/students/learnlabview/esa/environment.htm

Noticias, G. (2013). *El Hospital Provincial implanta la historia clínica electrónica móvil para agilizar la atención a los pacientes.* Obtenido de https://www1.dipcas.es/es/el-hospital-provincial-implanta-la-historia-clinica-electronica-movil-para-agilizar-la-atencion-a-los-pacientes/

Oscar B. (20 de 02 de 2009). *XATAKA MÓVIL*. Obtenido de XATAKA MÓVIL: https://www.xatakamovil.com/desarrollo/pantallas-tactiles-capacitivas-vs-resistivas

Perdices. (2016). *Estado del arte de la eHealth / eSalud / Telemedicina.* Obtenido de https://ehealthwars.wordpress.com/2011/07/19/estado-del-arte-de-la-ehealth-esalud-telemedicina/

Plan Nacional de Desarrollo. (2017-2021).

Rouse. (2005-2018). *TechTarget.* Obtenido de https://searchdatacenter.techtarget.com/es/definicion/MySQL

Shea, J. (2004). Handbook of Fingerprint Recognition. *Electrical Insulation Magazine, IEEE*.

SolidBi. (s.f.). *SolidWork.* Obtenido de https://solid-bi.es/solidworks/

Torres. (2018). *Hetpro*. Obtenido de https://hetpro-store.com/TUTORIALES/lector-de-huella-digital/

Villacrés. (2009). *Estándar IEEE 802.15.4.* Obtenido de Analisis del desempeño de una red WPAN

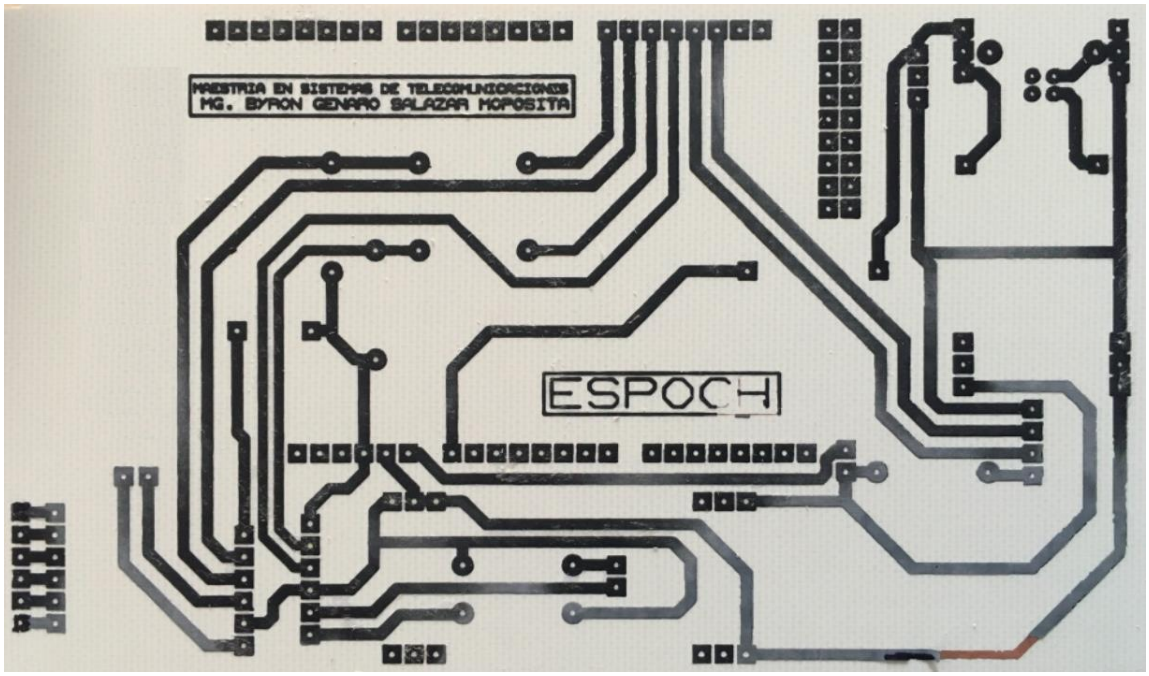
Xbee. (s.f.). *Digi.* Obtenido de https://xbee.cl/xbee-pro-s3b-xsc-rpsma/

zigzagmlt. (2018). *Inkscape*. Obtenido de https://inkscape.org/es/acerca-de/

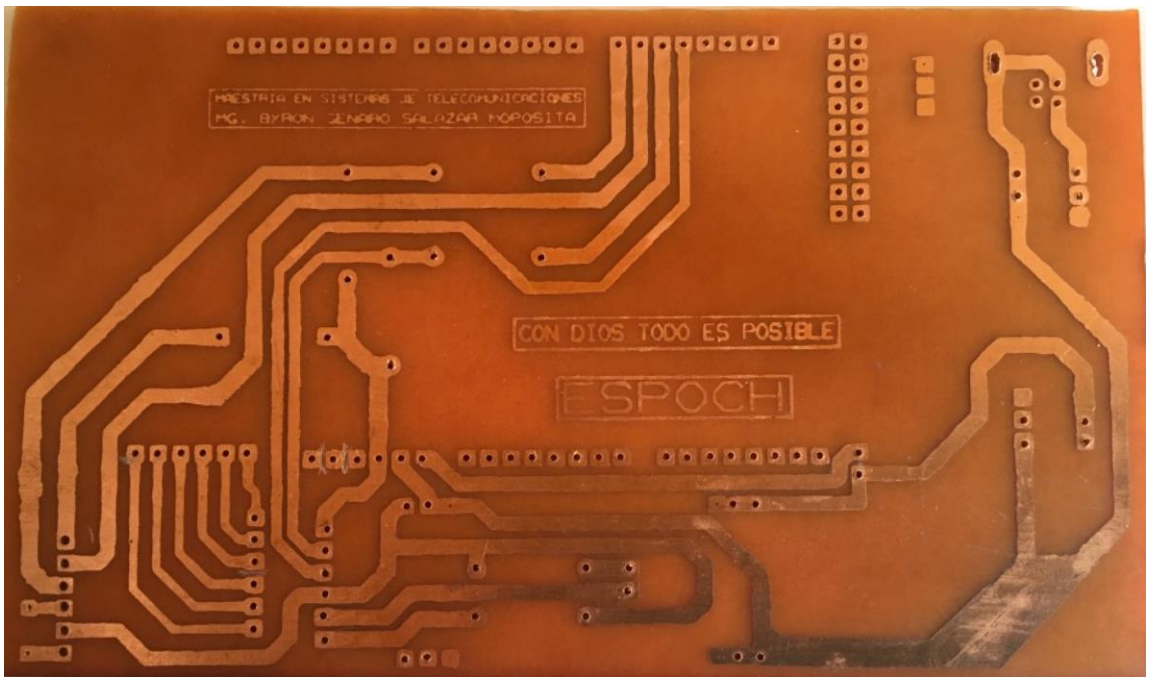
ANEXOS

ANEXO A: BAQUELITA EQUIPO TERMINAL Y ESTACIÓN CENTRAL

Pistas del Equipo Terminal

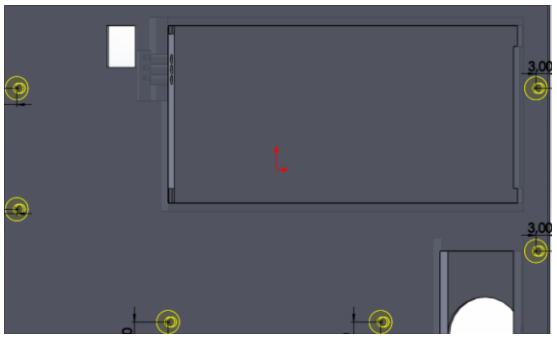


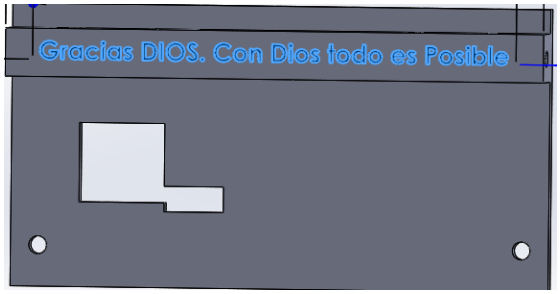
Pistas de la Estación Central

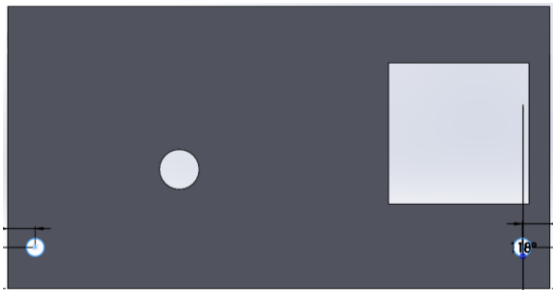


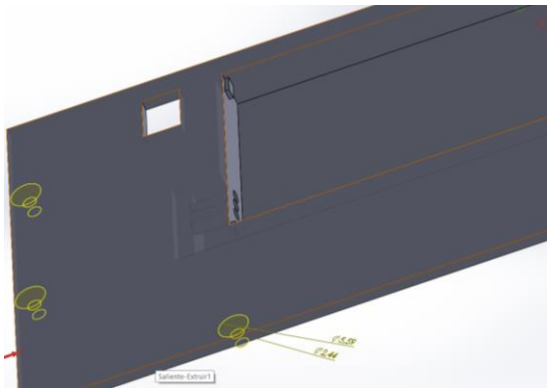
ANEXO B: PARTES DEL CHASIS EN 3D

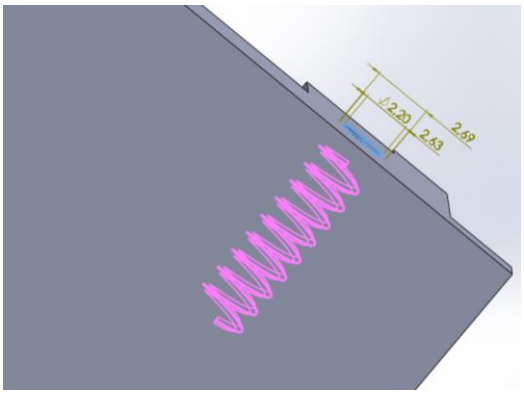
Chasis del Equipo Terminal

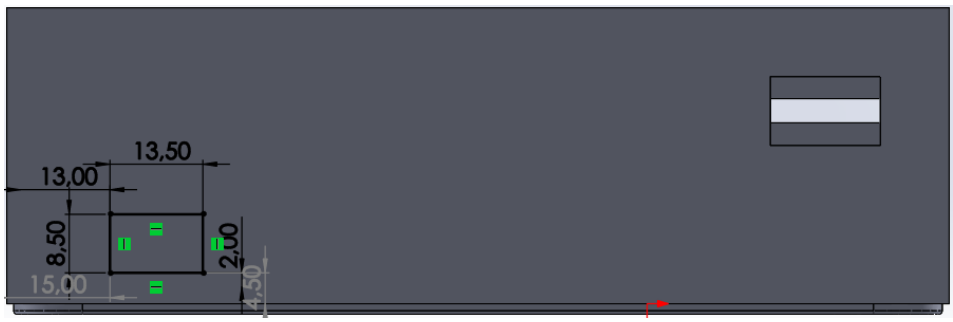




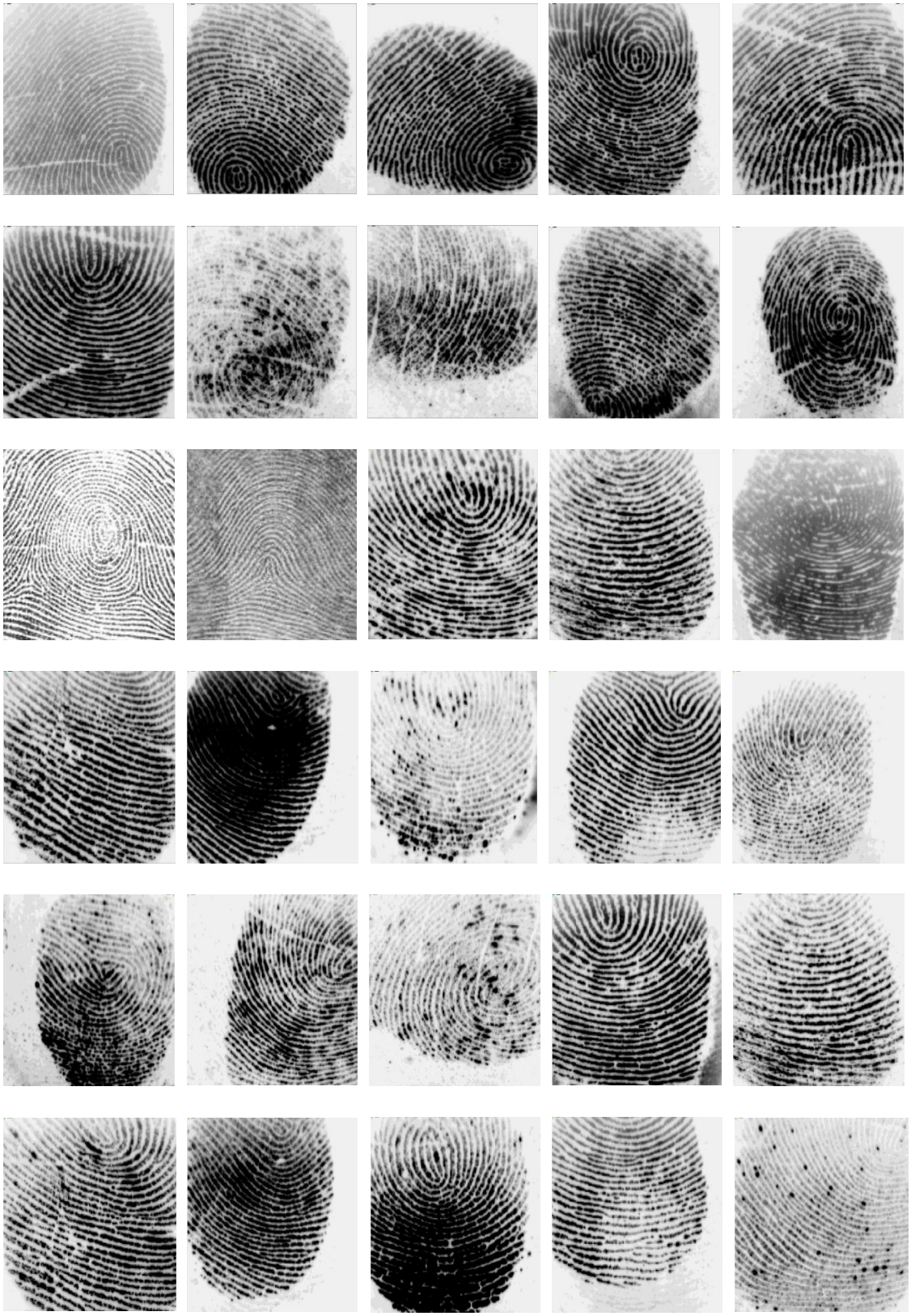




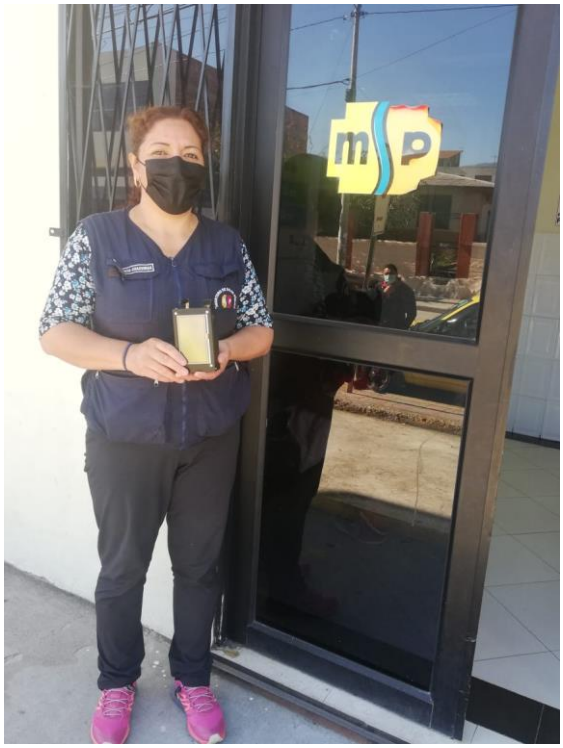




ANEXO C: BASE DE DATOS DE HUELLAS DIGITALES



ANEXO D: FOTOS DE PRUEBAS REALIZADAS




TEI-1


MINISTERIO
DE SALUD PUBLICA
AMBULANCIA
SVA
911

ANEXO E: PROGRAMAS DEL ARDUINO

PROGRAMA ARDUINO (EQUIPO TERMINAL)

```
/***libreria para uso de Pantalla Nextion***/
#include "Nextion.h"
/**códigos de control del sensor biomético**/
#define FINGERPRINT_STARTCODE 0xEF01
#define FINGERPRINT_COMMANDPACKET 0x1
#define FINGERPRINT_TIMEOUT 0xFF
#define FINGERPRINT_HISPEEDSEARCH 0x1B
#define FINGERPRINT_BADPACKET 0xFE
#define FINGERPRINT_ACKPACKET 0x7
#define FINGERPRINT_OBTENERIMAGEN 0x0A
#define FINGERPRINT_IMAGENBUFFER 0x0B
#define FINGERPRINT_REGMODEL 0x05
#define FINGERPRINT_UPLOADPC 0x08
#define FINGERPRINT_DOWNLOADBUFFER 0x09
#define FINGERPRINT_MATCH 0x03

/***********COMANDO CAPTURAR IMAGEN********/
#define SDACTILAR_CAPTURARIMG 0x01
/*genera archivo de caracteres de la imagen*/
#define SDACTILAR_GENERARIMG2TZ 0x02

/**NOTIFICACION ACUSE DE RECIBO DEL SENSOR**/
#define SDACTILAR_OK 0x00
#define SDACTILAR_PAQUETE_IN_ERROR 0x01
#define SDACTILAR_D_NO_DETECTADO 0x02
#define SDACTILAR_IMG_ERROR 0x03

/**********************CONSTANTES**********************/
#define DEFAULTTIMEOUT 5000  // milliseconds
uint32_t theAddress = 0xFFFFFFFF;
uint16_t Id_Dactilar = 0xFFFF;
uint16_t Seguridad = 0xFFFF;

/**********ESTADO DE RECEPCION DE LA SEÑAL**********/
#define icon1 5 //excelente señal
#define icon2 6 //muy buena
#define icon3 7 //de acuerdo
#define icon4 8 //
#define icon5 9 //
#define icon6 10 //Sin señal

/*Instanciación de objetos de la pantalla Nextion*/
NexText ObjPantallaPrincipal = NexText(0, 4, "tpp");
NexText ObjPantallaCedula = NexText(1, 15, "tpc");
NexText ObjNombre = NexText(2,6,"tn");
NexText ObjCedula = NexText(2,8,"tc");
NexText ObjTipodeSangre = NexText(2,10,"tts");
NexText ObjFechadeNacimiento = NexText(2,12,"tfn");
NexText ObjNombredeReferencia = NexText(2,14,"tnr");
NexText ObjTelefonodeReferencia = NexText(2,16,"ttr");

NexText ObjDireccion = NexText(3,6,"td");
```

```
NexText ObjAlergias = NexText(3,8,"ta");
NexText ObjEnfermedades = NexText(3,10,"te");
NexText ObjMedicamentos = NexText(3,12,"tm");
NexPicture ObjSenial = NexPicture(0, 5, "ps1");
NexNumber ObjBateria = NexNumber(0, 7, "n0");

/**********************VARIABLES**********************/
uint8_t datos[20];
uint8_t datosMinucias[20];
/*cuenta el numero de datos de Nextion*/
uint16_t contador=0;
uint16_t minucias[534];
int numerodeMinucias=0;

byte banderadeImpresiondePantallaNextion=0;
int auxiliar=0;
int ContadordeMonitoreo=0;
char opcion='0';

/*variables para gsm*/
char comandos;
char dato;

/*variables para medir voltaje de la bateria*/
/*pin para el potenciometro*/
int sensorPin = A0;
int aux = 0;      // select the pin for the LED
/*Almacena el valor del estado de la batería*/
int sensorValue = 0;
int outputValue = 0;

/*Variables para escribir la información en pantalla*/
String Nombre="";
String Cedula="";
String TipodeSangre="";
String FechadeNacimiento="";
String NombredeReferencia="";
String TelefonodeReferencia="";

String Direccion="";
String Alergias="";
String Enfermedades="";
String Medicamentos="";
String ContenidodelMensaje="";
String EstadodeSim="";
/***Banderas de control de procesos***/
bool stringCompletoNextion=false;
bool BanderaModuloSim=false;

bool tramaNextion=false;
bool impresiondeTramaCompleta=false;
bool impresionSoloMinucias=false;
bool tramadeMinuciasCompleta=false;

boolean comandoAT=false;
```

```
boolean habilitar=false;
boolean enviarMensaje=false;

void setup() {
  /*Configuracion de velocidad de los puertos Serie*/
  Serial3.begin(57600);
  Serial2.begin(57600);
  Serial1.begin(9600);
  Serial.begin(57600);
  /**Inicio del estado de la señal del módulo Sim**/
  ObjSenial.setPic(icon6);

  /***aqui debo indicar en la pantalla nextion que si hay conexion***/
  ObjPantallaPrincipal.setText("Comprobando el enlace");
  delay(500);
  Serial3.write("AT\r");
  delay(500);
  /*Configuración de formato en modo TEXTO*/
  Serial3.write("AT+CMGF=1\r");
  delay(500);
  /*Envia los mensajes SMS recién llegados directamente al puerto serie*/
  Serial3.print("AT+CNMI=2,2,0,0,0\r");
  delay(500);
  Serial3.write("AT\r");
  delay(500);
  /*Coloca al módulo en modo reposo para ahorrar energía*/
  Serial3.print("AT+CSCLK=2\r");
  delay(500);
  /*Escribe en el control de notificaciones que el sistema esta en línea*/
  ObjPantallaPrincipal.setText("Sistema en linea");
  delay(1000);
    /*Declaracion de parametros para medir el voltaje de la bateria*/
  sensorValue = analogRead(sensorPin);
  aux=outputValue = map(sensorValue, 725, 876, 0, 100);
  ObjBateria.setValue(0);
}

/***BUCLE PRINCIPAL***/
void loop() {

  if(tramaNextion){
    contador=0;
    /*vacia la variable*/
    ContenidodelMensaje="";
    /*verificacion de tipo de trama de control*/
    ordenPantallaNextion();
    /*envio de mensaje a estación base*/
    enviarMensajeSms();
    ContenidodelMensaje="";
    tramaNextion=false;
  }

  /*impresion de mensajes en la  pantalla nextion*/
  else if(stringCompletoNextion){
    impresionPantallaNextion();
```

```
    stringCompletoNextion=false;
  }

  /*Monitoreo del estado del módulo SIM800L*/
  else if(BanderaModuloSim){
    StatusModuloSim();
    BanderaModuloSim=false;
  }

  /*Bandera para el monitoreo del voltaje y señal inalámbrica*/
  else if(ContadordeMonitoreo==2000){
    MedidordeVoltaje();
    ComprobandoSenialdeModuloSim();
    ContadordeMonitoreo=0;
  }

  /*Monitoreo del estado de la Batería*/
  else if(ContadordeMonitoreo==1000){
    MedidordeVoltaje();
  }
  ContadordeMonitoreo++;
  delay(1);
}

/***Funcion para Medidor de voltaje de la Batería***/
void MedidordeVoltaje(){
  sensorValue = analogRead(sensorPin);
  outputValue = map(sensorValue, 725, 876, 0, 100);

  while(aux!=outputValue && (outputValue%20==0)){
    ObjBateria.setValue(outputValue);
    aux=outputValue;
  }
}

/***Comandos AT para control de modulo SIM800L***/
void ComprobarFuncionamientodelModuloSim(void){
  Serial3.write("AT\r");
}
void ComprobandoSenialdeModuloSim(void){
  Serial3.write("AT+CSQ\r");
}

/***Status del Modulo Sim800L***/
void StatusModuloSim(void){
  char opcionRssi='0';
  String rssi="", ber="";
  byte TipodeConsulta=0;
  if(EstadodeSim.substring(0, 2)=="OK"){
    if(TipodeConsulta==1){
      ObjPantallaPrincipal.setText("Sistema en línea");
      TipodeConsulta=0;
    }
  }
  else if(EstadodeSim.startsWith("+CPIN:") || EstadodeSim.startsWith("SMS Ready")){
```

```
   delay(1000);
   Serial3.write("AT\r");
   delay(1000);
   Serial3.write("AT+CMGF=1\r");
   delay(1000);
   Serial3.print("AT+CNMI=2,2,0,0,0\r");
   TipodeConsulta=1;
   delay(500);
  }

  else if(EstadodeSim.startsWith("+CSQ:")){
   TipodeConsulta=1;
   for(byte h=5; h<EstadodeSim.length(); h++){
    if((EstadodeSim.charAt(h)==' ')||(EstadodeSim.charAt(h)==',')){
     opcionRssi=EstadodeSim.charAt(h);
     continue;
    }
    switch(opcionRssi){
     case ' ':
      rssi+=EstadodeSim.charAt(h);
     break;
     case ',':
      ber+=EstadodeSim.charAt(h);
     break;
     defaul:
      Serial.print("Error Encontrado");
     break;
    }
   }

   /*Imagen para mostrar el estado de la señal inalámbrica*/
   if(rssi.toInt()==0){
    ObjSenial.setPic(icon6);
   }else if((rssi.toInt()>=1) && (rssi.toInt()<30)){
    // ObjSenial.setPic(icon6);
    ObjSenial.setPic(icon1);
   }else if((rssi.toInt()>=30) && (rssi.toInt()<67)){
    ObjSenial.setPic(icon2);
   }else if((rssi.toInt()>=67) && (rssi.toInt()<70)){
    ObjSenial.setPic(icon3);
   }else if((rssi.toInt()>=70) && (rssi.toInt()<80)){
    ObjSenial.setPic(icon4);
   }else if((rssi.toInt()>=80) && (rssi.toInt()<90)){
    ObjSenial.setPic(icon5);
   }else if(rssi.toInt()>=90){
    ObjSenial.setPic(icon6);
   }
  }
  EstadodeSim="";
}

/***ANÁLISIS DE TRAMA DE LA PANTALLA NEXTION***/
/***TIPO DE INSTRUCCIÓN NEXTION***/
void ordenPantallaNextion(void){
  String ContenidoMsj;
```

```
  /*PROCESAMIENTO DE HUELLA DÍGITAL*/
  if(datos[2]==0x01){
    optenerPlantilla();
    return;
  }

  /*EMPAQUETADO DE TRAMA CON DATOS DE LA CÉDULA*/
  else if(datos[2]==0x02){
    for(int k=3; k<=12; k++){
      ContenidoMsj=(datos[k]-48);
      ContenidodelMensaje+=ContenidoMsj;
    }

    ContenidodelMensaje=String("EF0102")+ContenidodelMensaje+String("FFFF");
    return;
  }

  /*IMPRESION DE DATOS EN PANTALLA NEXTION*/
   else if(datos[2]==0x03){
    impresionPantallaNextion();
    return;
  }
}

/***ENVIO DE MENSAJE SMS***/
void enviarMensajeSms(void){
  Serial3.write("AT+CMGF=1\r");
  delay(100);
  Serial3.write("AT+CMGS=\"+593986502852\"\r");
  //Serial3.write("AT+CMGS=\"+593986536258\"\r");
  delay(100);
  /*Informacion del usuario a consultar*/
  Serial3.print(ContenidodelMensaje);
  delay(100);
  Serial3.write((char)26);
  delay(100);
  Serial.write("mensaje enviado ....:)");
}

/***PUERTOS DE COMUNICACION SERIE***/
/***PUERTO SERIAL 0 (ARDUINO-PC)***/
void serialEvent(){
  while(Serial.available()>0){
    Serial2.write(Serial.read());
  }
}

/***PUERTO SERIAL 1 (ARDUINO Y PANTALLA NEXTION)***/
void serialEvent1(){
  while(Serial1.available()){
    datos[contador]=Serial1.read();
    if((contador==0) && (datos[0]!=0xEF)){
      continue;
    }
```

```
   if((contador==15)&&(datos[contador]==0xFF)&&(datos[contador-
1]==0xFF)&&(datos[contador-2]==0xFF)){
     ObjPantallaCedula.setText("Consultando datos de Usuario");
     contador=0;
     tramaNextion=true;
     continue;//Coloco continue para que no realice más operaciones
   }
   Else if((contador<15)&&(datos[contador]==0xFF)&&(datos[contador-
1]==0xFF)&&(datos[contador-2]==0xFF)){
     Serial.write("¡La cedula debe contener 10 Digitos!");
     ObjPantallaCedula.setText("La Cedula debe contener 10 Digitos");
     contador=0;
     continue;//Coloco continue para que no realice más operaciones
   }
   contador++;
  }
}

/***PUERTO SERIAL 2 (ARDUINO Y SENSOR BIOMETRICO)***/
void serialEvent2(){

  while(Serial2.available()){
    Serial.write(Serial2.read());
  }
}

/***PUERTO SERIAL 3 (ARDUINO Y MÓDULO SIM800L)***/
void serialEvent3(){
  while(Serial3.available()){
    char inChar=(char)Serial3.read();
    if(inChar=='&'){
      auxiliar=1;
      banderadeImpresiondePantallaNextion=1;
      break;
    }
    if(inChar!='&' && banderadeImpresiondePantallaNextion==1){

      if(auxiliar==1){
        opcion=inChar;
        if(opcion=='a'){
          vaciarVariablesNextion();
        }
        auxiliar=0;
        continue;
      }
      if(inChar=='\r'){
        banderadeImpresiondePantallaNextion=0;
        stringCompletoNextion=true;
      }
      switch(opcion){
        case 'a':
          Nombre+=inChar;
        break;
        case 'b':
          Cedula+=inChar;
```

```
        break;
        case 'c':
          TipodeSangre+=inChar;
        break;
        case 'd':
          FechadeNacimiento+=inChar;
        break;
        case 'e':
          NombredeReferencia+=inChar;
        break;
        case 'f':
          TelefonodeReferencia+=inChar;
        break;
        case 'g':
          Direccion+=inChar;
        break;
        case 'h':
          Alergias+=inChar;
        break;
        case 'i':
          Enfermedades+=inChar;
        break;
        case 'j':
          Medicamentos+=inChar;
        break;
        default:
          //Nombre="error";
        break;
      }
    }else{
      if( (inChar=='\n') && ( EstadodeSim.startsWith("\r") ) ){
        EstadodeSim="";
        break;
      }
      else if(inChar=='\n'){
        BanderaModuloSim=true;
        break;
      }
      else{
        EstadodeSim+=inChar;
      }
    }
  }
}

/***IMPRESION EN PANTALLA NEXTION***/
void impresionPantallaNextion(void){
  /*indica como cambiar el puerto a utilizar*/
  /*https://www.youtube.com/watch?v=936R5zau0t4 */
  //ObjMedicamentos.setText("CON DIOS TODO SE PUEDE");
  ObjNombre.setText(Nombre.c_str());
  ObjCedula.setText(Cedula.c_str());
  ObjTipodeSangre.setText(TipodeSangre.c_str());
  ObjFechadeNacimiento.setText(FechadeNacimiento.c_str());
  ObjNombredeReferencia.setText(NombredeReferencia.c_str());
```

```
  ObjTelefonodeReferencia.setText(TelefonodeReferencia.c_str());
  ObjDireccion.setText(Direccion.c_str());

  ObjAlergias.setText(Alergias.c_str());
  ObjEnfermedades.setText(Enfermedades.c_str());
  ObjMedicamentos.setText(Medicamentos.c_str());
}

void vaciarVariablesNextion(void){

  Nombre="";
  Cedula="";
  TipodeSangre="";
  FechadeNacimiento="";
  NombredeReferencia="";
  TelefonodeReferencia="";

  Direccion="";
  Alergias="";
  Enfermedades="";
  Medicamentos="";
}

/***FUNCION DE PROCESAMIENTO DE IMAGEN***/
void optenerPlantilla(void)
{
  int p = -1;
  uint8_t numeroDeScaneos=0;
  /*SDACTILAR_OK=0x00*/
  while (p != SDACTILAR_OK) {
    delay(100);
    /*Contador: numero de veces que se escanea el dedo*/
    numeroDeScaneos++;
    if (numeroDeScaneos >= 5) return;
    p = CapturarImagen();
    switch (p) {
    case SDACTILAR_OK:
      //Serial.println("Imagen Tomada");
      break;
    case SDACTILAR_D_NO_DETECTADO:
      //Serial.println("No se encuentra al dedo");
      break;
    case SDACTILAR_PAQUETE_IN_ERROR:
      //Serial.println("Error al recibir el paquete");
      break;
    case SDACTILAR_IMG_ERROR:
      //Serial.println("Error al determinar la imagen");
      break;
    default:
      //Serial.print("Error Desconocido: 0x"); Serial.println(p, HEX);
      break;
    }
  }

  /***CONVIERTE EN ARCHIVO DE CARACTERES LA IMAGEN***/
```

```
  p = GenerarImg2Tz(0x01);
  //p = GenerarImg2Tz();
  if (p != SDACTILAR_OK)  return;

  p = BusquedaRapida();
  //if (p != SDACTILAR_OK)  return;
  switch(p){
    case 0x00:
      //tramadeMinuciasCompleta=true;
      //Serial.write("huella encontrada");
      ObjPantallaPrincipal.setText("Consultando Usuario");
    break;
    case 0x01:
      //Serial.write("Problema");
      //tramadeMinuciasCompleta=false;
      ObjPantallaPrincipal.setText("Vuelva a colocar el dedo");
    break;
    case 0x09:
      //Serial.write("No existe coincidencia");
      //tramadeMinuciasCompleta=false;
      ObjPantallaPrincipal.setText("Registrar Usuario");
    break;
  }
}

/***FUNCION DE CAPTURA DE IMAGEN***/
uint8_t CapturarImagen(void)
{
  uint8_t  packet[]  =  {SDACTILAR_CAPTURARIMG};/*SDACTILAR_CAPTURARIMG
0x01*/
  /*(0x01, 0x01+2, 0x01)*/
  writePacket(FINGERPRINT_COMMANDPACKET, sizeof(packet) + 2, packet);
  return getReply(1);
}

/***FUNCION  PARA  GENERAR  ARCHIVO  DE  CARACTERES  EN  BUFFER
INDICADO***/
uint8_t GenerarImg2Tz(uint8_t slot) {
  /*SDACTILAR_GENERARIMG2TZ 0x02, 0X01*/
  uint8_t packet[] = {SDACTILAR_GENERARIMG2TZ, slot};
  /*(0X01, 2+2, {0X02, 0X01 })*/
  writePacket(FINGERPRINT_COMMANDPACKET, sizeof(packet)+2, packet);
  return getReply(1);
}

/***BUSQUEDA RAPIDA***/
uint8_t BusquedaRapida(void) {
  Id_Dactilar = 0xFFFF;
  Seguridad = 0xFFFF;
  // high speed search of slot #1 starting at page 0x0000 and page #0x00A3
  //(1B=27,1,0,0,0,A3=163)
  uint8_t packet[] = {FINGERPRINT_HISPEEDSEARCH, 0x01, 0x00, 0x00, 0x00, 0xA3};
  //(1, 8, PAQUETE)
  writePacket(FINGERPRINT_COMMANDPACKET, sizeof(packet)+2, packet);
  uint8_t verificaciondeTrama = getReply3();
```

```
  Id_Dactilar = datosMinucias[10];
  Id_Dactilar <<= 8;
  Id_Dactilar |= datosMinucias[11];

  Seguridad = datosMinucias[12];
  Seguridad <<= 8;
  Seguridad |= datosMinucias[13];

  if((Id_Dactilar>0)&&(datosMinucias[9]==0x00)){

 /*EMPAQUETADO DE TRAMA CON MINUCIAS*/
ContenidodelMensaje=String("EF0101FF01")+Id_Dactilar+String("FF02")+Seguridad+String("
FFFF");
  }else{
    ContenidodelMensaje="";
  }
  return verificaciondeTrama;
}

/***ENVÍO DE TRAMA DE DATOS HACIA EL SENSOR BIOMÉTRICO***/
void writePacket(uint8_t packettype, uint16_t len, uint8_t *packet){
  /*capturar imagen:(0x01, 0x01+2, 0x01)*/
  /*generar un archivo de caracteres desde desde la imagen(0X01, 2+2, {0X02, 0X01 })*/
  //packettype=>FINGERPRINT_COMMANDPACKET 0x1
  //INICIO 2
  //DIRECCION 4
  //IDENTIFICADOR DEL PAQUETE 1
  //LARGO DEL PAQUETE 2
  //PAQUETE DE DATOS XXX
  //CHECK SUM 2
  //.....................................................................
  // Serial.print("Longitud: "); Serial.println(len);
  //.....................................................................
  //INICIO

  /*FINGERPRINT_STARTCODE 0xEF01*/
  Serial2.write((uint8_t)(FINGERPRINT_STARTCODE >> 8));
  Serial2.write((uint8_t)FINGERPRINT_STARTCODE);
  //DIRECCION
  /*theAddress = 0xFFFFFFFF;*/
  Serial2.write((uint8_t)(theAddress >> 24));
  Serial2.write((uint8_t)(theAddress >> 16));
  Serial2.write((uint8_t)(theAddress >> 8));
  Serial2.write((uint8_t)(theAddress));
  //IDENTIFICADOR DEL PAQUETE
  Serial2.write((uint8_t)packettype);
  //LARGO DEL PAQUETE
  Serial2.write((uint8_t)(len >> 8));
  Serial2.write((uint8_t)(len));

  uint16_t sum = (len>>8) + (len&0xFF) + packettype;
  //PAQUETE DE DATOS
  for (uint8_t i=0; i< len-2; i++)
  {
```

```
    Serial2.write((uint8_t)(packet[i]));
    sum += packet[i];
  }
  //CHECK SUM
  Serial2.write((uint8_t)(sum>>8));
  Serial2.write((uint8_t)sum);
}
/***VALIDACION DE TRAMAS DE CONFIRMACION DE DATOS***/
int getReply(uint8_t OpcionSelect)
{
  uint8_t reply[20], idx;
  uint16_t timer=0;
  idx = 0;

while (true) {
    while (!Serial2.available()) {
      delay(1);
      timer++;
      if (timer >= DEFAULTTIMEOUT) return FINGERPRINT_TIMEOUT;
      /*DEFAULTTIMEOUT 5000; FINGERPRINT_TIMEOUT 0xFF*/
    }
    // something to read!
    reply[idx] = Serial2.read();
    /*FINGERPRINT_STARTCODE 0xEF01*/
    if ((idx == 0) && (reply[0] != (FINGERPRINT_STARTCODE >> 8)))
      continue;

    // check packet!
    if(OpcionSelect == 1)
    {
        if(idx == 11)
        {
          uint16_t ck_in_sum =0 ;
          uint16_t ck_sum_fg =0 ;

          for (uint8_t i = 6; i < 10; i++){ck_in_sum += reply[i];}

          ck_sum_fg = reply[10];
          ck_sum_fg = ck_sum_fg<<8;
          ck_sum_fg = ck_sum_fg | reply[11];

          if(ck_in_sum == ck_sum_fg){return reply[9];}
          return -1;
        }
    }
    if(OpcionSelect == 2)
    {
        if(idx == 13)
        {
          uint16_t ck_in_sum =0 ;
          uint16_t ck_sum_fg =0 ;

          for (uint8_t i = 6; i < 12; i++){ck_in_sum += reply[i];}

          ck_sum_fg = reply[12];
```

```
          ck_sum_fg = ck_sum_fg<<8;
          ck_sum_fg = ck_sum_fg | reply[13];

          if(ck_in_sum == ck_sum_fg)
          {
            ck_sum_fg = reply[10];
            ck_sum_fg = ck_sum_fg<<8;
            ck_sum_fg = ck_sum_fg | reply[11];
            return (uint8_t)ck_sum_fg;
          }
          return -1;
        }
    }
    idx++;
  }
}

/***GETREPLY2***/
uint8_t getReply2(uint8_t packet[]) {
  uint8_t reply[20], idx;
  uint16_t timer=0;
  idx = 0;

while (true) {
    while (!Serial2.available()) {
      delay(1);
      timer++;
      if (timer >= DEFAULTTIMEOUT) return FINGERPRINT_TIMEOUT;
    }
    // something to read!
    reply[idx] = Serial2.read();

    if ((idx == 0) && (reply[0] != (FINGERPRINT_STARTCODE >> 8)))
      continue;
    idx++;

    // check packet!
    if (idx >= 9) {
      if ((reply[0] != (FINGERPRINT_STARTCODE >> 8)) ||
          (reply[1] != (FINGERPRINT_STARTCODE & 0xFF)))
          return FINGERPRINT_BADPACKET;
      uint8_t packettype = reply[6];
      //Serial.print("Packet type"); Serial.println(packettype);
      uint16_t len = reply[7];
      len <<= 8;
      len |= reply[8];
      len -= 2;
      //Serial.print("Packet len"); Serial.println(len);
      if (idx <= (len+10)) continue;
      packet[0] = packettype;
      for (uint8_t i=0; i<len; i++) {
        packet[1+i] = reply[9+i];
      }
      return len;
    }
```

```
  }
}

/***GETREPLY3***/
int getReply3()
{
  uint8_t idx;
  uint16_t timer=0;
  idx = 0;

while (true) {
    while (!Serial2.available()) {
      delay(1);
      timer++;
      if (timer >= DEFAULTTIMEOUT) return FINGERPRINT_TIMEOUT;

      /*DEFAULTTIMEOUT 5000; FINGERPRINT_TIMEOUT 0xFF*/
    }
    datosMinucias[idx] = Serial2.read();
    /*FINGERPRINT_STARTCODE 0xEF01*/
    if ((idx == 0) && (datosMinucias[0] != (FINGERPRINT_STARTCODE >> 8)))
      continue;

    // check packet!
    if(idx == 15){
      uint16_t ck_in_sum =0 ;
      uint16_t ck_sum_fg =0 ;

      for (uint8_t i = 6; i < 14; i++){ck_in_sum += datosMinucias[i];}

      ck_sum_fg = datosMinucias[14];
      ck_sum_fg = ck_sum_fg<<8;
      ck_sum_fg = ck_sum_fg | datosMinucias[15];

      if(ck_in_sum == ck_sum_fg){return datosMinucias[9];}
      return -1;
    }

    idx++;
  }
}
```

PROGRAMA ARDUINO (ESTACIÓN BASE)

```
boolean comandoAT=false;
boolean habilitar=false;
boolean enviarMensaje=false;
boolean mensajeCompleto=false;
boolean BanderadeManejodeSim=false;
String DatosdeUsuario="";
String BufferdeSim="";
int auxiliar=0;
int auxiliar2=0;
int contador=0;
void setup() {
```

```
  Serial.begin(57600);
  Serial3.begin(57600);
  delay(500);
  Serial3.write("AT\r");
  delay(500);
  Serial3.write("AT+CMGF=1\r");
  delay(500);
  Serial3.print("AT+CNMI=2,2,0,0,0\r");
  delay(500);
  Serial3.write("AT\r");
  delay(500);
  Serial3.print("AT+CSCLK=2\r");
  delay(500);
}
void loop() {
  if(comandoAT){
    Serial3.write("AT\r");
    comandoAT=false;
  }
  else if(habilitar){
    digitalWrite(2, HIGH);
    delay(1000);
    digitalWrite(2,LOW);
    delay(5000);

    Serial3.write("AT+CMGF=1\r");
    delay(100);
    Serial3.print("AT+CNMI=2,2,0,0,0\r");
    delay(100);
    habilitar=false;
  }
  else if(enviarMensaje){
    enviarMensajeSms();
    enviarMensaje=false;
  }
  else if(mensajeCompleto){
    envioDatosUsuario();
    mensajeCompleto=false;
  }
  else if(BanderadeManejodeSim){
    habilitarRecepciondeMensajesdelModuloSim();
    BanderadeManejodeSim=false;
  }
}


/********habilitar la recepcion de mensajes del modulo SIM*******/

void habilitarRecepciondeMensajesdelModuloSim(){
    Serial3.write("AT\r");
    delay(1000);
    Serial3.write("AT+CMGF=1\r");
    delay(1000);
    Serial3.print("AT+CNMI=2,2,0,0,0\r");
    delay(500);
}
```

```
/***************Puertos de Comunicacion Seriales******************/
void serialEvent(){
  while(Serial.available()>0){

    char inChar=(char)Serial.read();

    if(inChar=='&' || DatosdeUsuario=="&"){
      auxiliar=1;
    }
    if(auxiliar==1){
      if(inChar=='\n' || (contador>=120 && inChar=='&')){
        //auxiliar=0;
      if(inChar=='\n'){
        delay(6000);
       }
        mensajeCompleto=true;
        break;
      }
      DatosdeUsuario+=inChar;
      contador++;
    }
  }
}
void serialEvent3 (){
  while(Serial3.available()>0){
    Serial.write(Serial3.read());
  }
}
void enviarMensajeSms(void){
  Serial3.write("AT+CMGF=1\r");
  delay(100);
  //Serial3.write("AT+CMGS=\"+593986502852\"\r");
  Serial3.write("AT+CMGS=\"+593986536258\"\r");
  //Serial3.write("AT+CMGS=\"+593979159888\"\r");
  delay(100);
  Serial3.print(DatosdeUsuario);
  delay(100);
  Serial3.write((char)26);
  delay(100);
  Serial.write("mensaje enviado ....:)");
}
void envioDatosUsuario(void){
  Serial.print("bien");
  Serial.print(DatosdeUsuario);
  Serial.print("bien");
  enviarMensajeSms();
  if(contador>=120){
    DatosdeUsuario="&";
  }else{
    auxiliar=0;
    DatosdeUsuario="";
  }
  contador=0;
```

PROGRAMA ARDUINO (REGISTRAR USUARIOS)

```
void setup() {
  // put your setup code here, to run once:
  Serial.begin(57600);
  Serial2.begin(57600);
  pinMode(2, OUTPUT);
}
void loop() {
}
void serialEvent(){
  while(Serial.available()>0){
      Serial2.write(Serial.read());
  }
}
void serialEvent2 (){
   while(Serial2.available()>0){
    Serial.write(Serial2.read());
  }
}
```