



**INSTITUTO TECNOLÓGICO SUPERIOR DE
HUAUCHINANGO**

**MAESTRÍA EN
TECNOLOGÍAS DE LA
INFORMACIÓN**

**“Prototipo de dispositivo electrónico para envío de
ubicación, temperatura y ritmo cardiaco de una
persona a un servidor de internet para el monitoreo
desde una app móvil”**

TESIS

QUE PARA OBTENER EL GRADO DE

**MAESTRO EN TECNOLOGÍAS DE LA
INFORMACIÓN**

P R E S E N T A

R I C A R D O L U N A S A N T O S

M 1 9 3 1 5 0 7

Director de Tesis:
Mtro. Aldo Hernández Luna

Coasesor de Tesis:
Dr. Jacinto Torres Jiménez

HUAUCHINANGO, PUE. FEBRERO DE 2022

MAESTRÍA EN TECNOLOGÍAS DE LA INFORMACIÓN

ACTA DE REVISIÓN DE TESIS

En la ciudad de Huauchinango, Puebla, siendo las **10:00** horas del día **28-02-2022** se reunieron los miembros del Comité Sinodal designado por el H. Consejo de Posgrado de la Maestría en Tecnologías de la Información del Instituto Tecnológico Superior de Huauchinango, para examinar y evaluar la tesis de grado titulada:

“PROTOTIPO DE DISPOSITIVO ELECTRÓNICO PARA ENVIÓ DE UBICACIÓN, TEMPERATURA Y RITMO CARDIACO DE UNA PERSONA A UN SERVIDOR DE INTERNET PARA EL MONITOREO DESDE UNA APP MÓVIL”

Presentada por el (la) alumno (a):

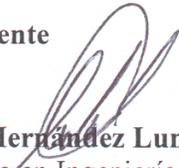
RICARDO LUNA SANTOS, con número de control M1931507

Aspirante al grado de:

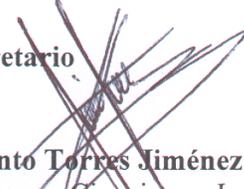
MAESTRO(A) EN TECNOLOGÍAS DE LA INFORMACIÓN

Después de intercambiar opiniones los miembros de la comisión manifestaron la aprobación e impresión final de la tesis, en virtud que satisface los requisitos señalados por las disposiciones reglamentarias vigentes.

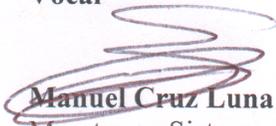
Presidente


Aldo Hernández Luna
Maestro en Ingeniería
Cédula profesional: 11606723

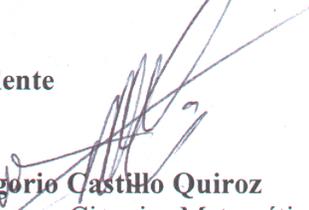
Secretario


Jacinto Torres Jiménez
Doctor en Ciencias en Ingeniería Eléctrica
Cédula profesional: 8112868

Vocal


Manuel Cruz Luna
Maestro en Sistemas Computacionales
Cédula profesional: 8257588

Suplente


Gregorio Castillo Quiroz
Doctor en Ciencias Matemáticas
Cédula profesional: 09174990



Sello de la institución
INSTITUTO TECNOLÓGICO
SUPERIOR DE HUAUCHINANGO
SUBDIRECCIÓN DE POSGRADO E INVESTIGACIÓN

SECCIÓN: SUBDIRECCIÓN DE POSGRADO
E INVESTIGACIÓN
OFICIO: SPeI/035/2022
ASUNTO: AUTORIZACIÓN DE
IMPRESIÓN DEFINITIVA DE
TESIS

Huauchinango, Pue., 28 de febrero de 2022

ING. RICARDO LUNA SANTOS
ESTUDIANTE DE LA MAESTRÍA EN TECNOLOGÍAS
DE LA INFORMACIÓN
PRESENTE:

Por este conducto tengo el agrado de comunicarle que el jurado designado para que Usted obtenga el grado de **MAESTRO EN TECNOLOGÍAS DE LA INFORMACIÓN**, ha informado a esta Subdirección de Posgrado e Investigación, su aceptación para la impresión Definitiva de su trabajo de tesis, en el cual lleva por título:

**“PROTOTIPO DE DISPOSITIVO ELECTRÓNICO PARA ENVIÓ DE UBICACIÓN,
TEMPERATURA Y RITMO CARDIACO DE UNA PERSONA A UN SERVIDOR
DE INTERNET PARA EL MONITOREO DESDE UNA APP MÓVIL”**

Por lo anterior se autoriza la impresión de su trabajo, esperando que el logro del mismo sea acorde con sus aspiraciones profesionales.

ATENTAMENTE



C. FERNANDO GONZÁLEZ CRUZ
SUBDIRECTOR DE POSGRADO E INVESTIGACIÓN



**INSTITUTO TECNOLÓGICO
SUPERIOR DE HUAUCHINANGO**
SUBDIRECCION DE POSGRADO E INVESTIGACION

DEDICATORIA Y LEMA

El presente trabajo está dedicado a mi familia, esposa e hijos que me apoyaron en el desarrollo de esta idea y me animaron en este camino, me compartieron su paciencia y palabras de ánimo, para mejorar cada día.

El trabajo en equipo comienza construyendo la confianza. Y la única manera de hacerlo es superando nuestra necesidad de invulnerabilidad.

Patrick Lencioni

AGRADECIMIENTOS

Gracias principalmente a Dios por prestarme vida, darme el conocimiento para compartir y desarrollar este proyecto que permitirá mejorar la calidad de vida de los seres humanos.

Gracias a mi familia, esposa e hijos por estar presentes a lado de mi carrera profesional y apoyarme en todos los aspectos.

Gracias a mis profesores que sirvieron de guías en este proceso de desarrollo personal, por sus enseñanzas.

Gracias a mis padres que me enseñaron el valor de la perseverancia, buscar llegar más lejos y esforzarme día a día.

RESUMEN

En la actualidad existen seres humanos que por sus problemas de salud necesitan cuidados y monitorización constante, sin embargo, debido a la inseguridad no solo necesitamos conocer el estado de salud de nuestros seres queridos si no también su ubicación. El presente proyecto pretende contribuir al cuidado de las personas mediante un dispositivo electrónico que llamaremos UbiC's el cual se portará en alguna parte del cuerpo para el control y monitoreo constante del pulso cardiaco, temperatura y ubicación, esta información será enviada y almacenada en un servidor de base de datos alojado en un hosting utilizando la red GPRS (Servicio General de Paquetes vía Radio) y estará vinculada a un dispositivo móvil por medio de una App instalada en el celular del cuidador, la App permitirá acceder a la información enviada de los últimos 7 días en forma de reporte, además de monitorear en tiempo real el estado de salud, también funciona como dispositivo de seguridad, pues UbiC's contará con un botón de ayuda que se activa manualmente, en caso de que el usuario se encuentre en una situación de peligro UbiC's manda la ubicación en tiempo real del portador, así como la última lectura de sus signos vitales, de este modo, la persona que se encuentra a su cargo podrá saber si la salud del usuario se encuentra estable o si él o ella corre algún peligro.

Palabras clave: *monitorización, ubicación, dispositivo electrónico, base de datos, hosting, app Móvil*

ABSTRACT

Currently there are human beings who need constant care and monitoring due to their health problems, however, due to insecurity we not only need to know the health status of our loved ones but also their location. This project aims to contribute to the care of people through an electronic device that we will call Ubic's which will be carried in some part of the body for the constant control and monitoring of the heart pulse, temperature and location, this information will be sent and stored on a server of database hosted on a hosting using the GPRS (General Packet Radio Service) network and will be linked to a mobile device through an App installed on the caregiver's cell phone, the App will allow access to the information sent in the last 7 days in the form of a report, in addition to monitor health status in real time, it also works as a security device, since Ubic's will have a help button that is activated manually, in case the user is in a dangerous situation Ubic's sends the real-time location of the carrier, as well as the last reading of his vital signs, in this way, the person in charge will be able to know if the user's health is stable or if he or she is in danger.

Keywords: *monitoring, location, electronic device, database, hosting, Mobile app*

ÍNDICE

Dedicatoria y lema	II
Agradecimientos	III
Resumen.....	IV
Abstract	V
Índice	VI
Lista de figuras.....	VIII
Lista de tablas	X
Lista de acrónimos y abreviaturas	XI
1.1. Introducción	2
1.2. Estado del Arte.....	3
1.3. Objetivo General.....	5
1.4. Objetivos Específicos	5
1.5. Justificación	6
1.6. Motivación Para Elaborar la Investigación.....	7
1.7. Organización de la Tesis.....	8
1.8. Publicaciones que Sustenten el Trabajo de Tesis	8
2.1. Marco Conceptual.....	10
2.1.1. Signos vitales	10
2.1.2. Parámetros fisiológicos	10
2.1.3. Monitores de signos vitales.....	11
2.1.4. Proyectos de investigación.....	11
2.1.5. Ventajas del proyecto	13
3.1. Metodología	16
3.1.1. Análisis.....	17
3.1.2. Diseño	17
3.1.3. Desarrollo.....	18
3.1.4. Pruebas	18
3.1.5. Puesta a punto del prototipo.....	18
3.2. Métodos y Procedimientos.....	19
3.2.1. El análisis y la planeación	19

3.2.1.1. Cronograma de actividades	19
3.2.1.2. Listado de herramientas, materiales y software	20
3.2.1.3. Listado de costos	23
3.2.1.4. Análisis de funcionamiento del proyecto	23
3.2.2. Diseño del proyecto.....	25
3.2.2.1. Diseño de casos de uso	25
3.2.2.2. Diagramas de flujo	26
3.2.2.3. Diseño de la base de datos.....	27
3.2.2.4. Modelo relacional.....	28
3.2.2.5. Diccionario de datos.....	29
3.2.3. Diseño eléctrico y electrónico del dispositivo	30
3.2.4. Modelado 3D.....	35
3.2.5. Diseño de interfaces del software web para la administración de usuarios y dispositivos.....	38
3.2.6. Diseño de interfaces de la App Móvil.....	41
3.3. Desarrollo del Proyecto	42
3.3.1. Programación del microcontrolador y sus componentes	42
3.3.2. Programación del sistema de administración web	49
3.3.6. Programación de la App Móvil.....	53
4.1. Resultados y Discusión.....	57
4.1.1. Desarrollo del prototipo	57
4.1.2. App Móvil.....	59
4.1.3. Sistema de administración de usuarios, cuidadores y dispositivos.....	60
5.1. Conclusiones.....	63
5.2. Recomendaciones	64
5.3. Trabajos Futuros	65
A.1. Publicación de un artículo en revista arbitrada e indexada.....	68
Referencias.....	69

LISTA DE FIGURAS

Figura 3.1 Etepas del desarrollo del proyecto utilizando la metodología XP (Programación Extrema).....	16
Figura 3.2 Etapa de análisis de la información.....	19
Figura 3.3 Etapas de Diseño, Desarrollo y pruebas.....	20
Figura 3.4 Ciclo de envío, consulta, vinculación y proceso de los datos en el prototipo	25
Figura 3.5 Casos de uso del sistema web, app móvil y dispositivo.....	26
Figura 3.6 Proceso de almacenamiento, vinculación y consulta de datos	27
Figura 3.7 Modelo Entidad Relación de la base de datos.....	28
Figura 3.8 Modelo relacional de la base de datos.....	28
Figura 3.9 Placa Arduino Nano	31
Figura 3.10 Modulo GPS6MV2.....	31
Figura 3.11 Sensor de temperatura infrarrojo MLX90614.....	31
Figura 3.12 Pulsioxímetro MAX30102	32
Figura 3.13 Modulo IoT GA6- B GSM/GPRS	32
Figura 3.14 Diagrama electrónico para envío de ubicación, temperatura, y ritmo cardiaco a través de una red GPRS	33
Figura 3.15 Diagrama eléctrico del prototipo.....	35
Figura 3.16 Base del brazalete que contiene los espacios para introducir los componentes	36
Figura 3.17 Taba del brazalete.....	36
Figura 3.18 Base de baterías.....	37
Figura 3.19 Tapa del banco de baterías	37
Figura 3.20 Brazalete portado en el brazo	38
Figura 3.21 Interfaz de acceso al sistema solo para administradores	39
Figura 3.22 Interfaz del menú principal registro de usuarios	39
Figura 3.23 Interfaz de menú principal registro de dispositivos.....	40
Figura 3.24 Interfaz de vinculación de usuarios con dispositivos, un usuario puede vincularse a varios dispositivos, pero un dispositivo solo se vincula a un usuario	40
Figura 3.25 Interfaz de acceso e interfaz de dispositivos vinculados al dispositivo móvil ..	41
Figura 3.26 Interfaz de monitoreo del dispositivo e interfaz de historial semanal de las lecturas del UbiC's	42

Figura 3.27 Importar librerías y declarar variables del sistema.....	43
Figura 3.28 Programación de la función setup y la función loop	44
Figura 3.29 Código de la función ubicación.....	44
Figura 3.30 Código de la función temperatura	45
Figura 3.31 Codificación del sensor de ritmo cardiaco	46
Figura 3.32 Código para calcular el ritmo cardiaco.....	47
Figura 3.33 Código para el envío del mensaje de texto de forma automática o manual	48
Figura 3.34 Función de acceso al sistema.....	49
Figura 3.35 Insertar información en la base de datos	50
Figura 3.36 Código que muestra los resultados de búsqueda	51
Figura 3.37 Paginador y columnas de eliminar y modificar.....	52
Figura 3.38 Programando pantalla de acceso	53
Figura 3.39 Código de validación de claves de acceso.....	54
Figura 3.40 Código para cargar los dispositivos vinculados a la cuenta de asistente cuidador	55
Figura 4.1 Lecturas del dispositivo.....	57
Figura 4.2 Lecturas en rango normal	58
Figura 4.3 Lecturas fuera de rango	58
Figura 4.4 Acceso y lista de dispositivos vinculados al cuidador	59
Figura 4.6 Notificaciones en mensaje de texto	60
Figura 4.8 Monitoreo desde la App móvil.....	61

LISTA DE TABLAS

Tabla 3.1. Relación de materiales, herramientas y software	22
Tabla 3.2. Relación de gastos de inversión del proyecto	23
Tabla 3.3 cliente.....	29
Tabla 3.4 Índices del cliente	29
Tabla 3.5 cliente-dispositivo.....	29
Tabla 3.6 Índices de tabla cliente dispositivo	29
Tabla 3.7 dispositivo.....	30
Tabla 3.8 Índices de tabla dispositivo.....	30
Tabla 3.9 registro	30
Tabla 3.10 Índices tabla registros	30

LISTA DE ACRÓNIMOS Y ABREVIATURAS

- °C.- Abreviatura de grados centígrados.
- API KEY. - Abreviatura de application programming interfaces que se traduce al español como interfaz de programación de aplicación.
- APP. - Programa o conjunto de programas informáticos que realizan un trabajo en específico.
- AT. - Son instrucciones codificadas que conforman un lenguaje de comunicación.
- BPM. - Hace referencia a la frecuencia cardiaca y se mide el número de latidos por minuto.
- CRUD. - Es un acrónimo para la manipulación de la información almacenada que tiene como función crear, leer, guardar, actualizar y eliminar.
- CURP. - Es la clave única de registro de población en México.
- ECG. - Abreviatura utilizada para referirse a un electrocardiograma.
- GANTT. - Herramienta grafica cuyo objetivo es exponer el tiempo y dedicación previsto para diferentes tareas o actividades.
- GND. - Es la definición clásica de tierra en ingles ground en la electrónica.
- GPRS. - Hace referencia a una señal de radio que proporciona servicios de comunicación, servicio general de paquetes vía radio.
- GPS. - Sistema de navegación terrestre que significa sistema de posicionamiento global.
- GSM. - Sistema de radiotelefonía celular digital.
- I2C. - Protocolo de comunicación serial, define la trama de datos y las conexiones físicas entre varios dispositivos utilizando 2 bits.
- IDE. - Se define como un entorno de desarrollo integrado que proporciona herramientas para la compilación y programación.
- IMEI. - Es un identificador internacional de equipo móvil el cual cuenta con 15 dígitos únicos y no repetitivos.
- IP. - Significa protocolo de internet y se compone por un conjunto de reglas que permiten la comunicación.
- IR. - Es la radiación infrarroja que emite un dispositivo al encender.

- JSON. - Es un formato de texto sencillo para el intercambio de datos basado en JavaScript.
- MYSQL. - Es un sistema de gestión de base de datos relacionales basado en el lenguaje de consulta SQL.
- OMS. - Abreviatura de Organización Mundial de Salud.
- PANI. - Hace referencia a la presión arterial y es un indicador esencial del estado fisiológico de un ser humano.
- PC. - Computadora personal.
- PCB. - Son las siglas para referirse a una placa de circuito impreso.
- PDF. - Es un formato de documento portátil.
- RF. - Radiofrecuencia.
- RX. - Recepción de señales eléctricas.
- SCL. - Es una línea de pulsos de reloj que permite sincronizar un sistema.
- SDA. - Es una línea por la cual se mueven los datos en forma de pulsos eléctricos entre los dispositivos.
- SIM. - Módulo de identificación abonado es una tarjeta inteligente desmontable utilizada en teléfonos móviles.
- SPO2. - Es la cantidad de oxígeno en la sangre.
- TCP. - Protocolo de transmisión de control que son reglas para permitir conexión entre dos o más dispositivos.
- TX. - Envío de señales eléctricas.
- UBIC´S. - Dispositivo de ubicación y signos vitales.
- VCC. - Hace referencia a un colector de voltaje de corriente eléctrica positiva.
- IoT. - Internet e las cosas.

CAPITULO1

INTRODUCCIÓN

1.1. Introducción

En la actualidad existen diferentes padecimientos que provocan un desgaste en la salud, las enfermedades crónico degenerativas provocar muchas fallas en nuestro organismo, existen muchas personas que tienen que realizar visitas constantes al médico para conocer signos que denotan fallas en su organismo, un ejemplo sería el ritmo cardiaco y temperatura. Todas estas alertas que envía nuestro cuerpo son síntomas que no se deben tomar a la ligera, si a este proceso le sumamos la falta de atención por parte de las personas que se encargan del cuidado de un familiar, amigo o conocido podemos identificar que el riesgo es mayor.

Como parte de las necesidades y la importancia de llevar un monitoreo constante y un historial se propone crear un prototipo electrónico que permita obtener los valores de temperatura y ritmo cardiaco cada determinado tiempo, esta información se almacena en un servidor de internet, además para dar mayor seguridad a la implementación de dicho dispositivo agregaremos los datos de ubicación, de esta forma al monitorear al paciente no solo obtendremos lecturas importantes si no que podríamos saber cuál es su ubicación.

El presente documento muestra la forma detallada de como interactúa el prototipo electrónico de monitoreo para obtener los datos, el sistema web que permitirá vincular estos dispositivos con los celulares de las personas que se encargan de su cuidado y un App instalada en los celulares con sistema operativo Android que funcionará como un monitor gráfico que muestra información en tiempo real del portador.

El dispositivo de monitoreo de temperatura, ritmo cardiaco y ubicación no solo está diseñado para pacientes con enfermedades crónico degenerativas, también se puede utilizar para garantizar la seguridad de cualquier persona ya que incorpora un botón de ayuda el cual permite enviar una alerta al celular de las personas vinculadas en donde podrán conocer la ubicación del portador en forma de mensaje de texto.

1.2. Estado del Arte

El cuidado de la salud es fundamental para que el organismo se muestre bien y con buena vitalidad. Cada persona tiene su propio metabolismo y lleva un estilo particular de vida, pero en general todos deben tomar en cuenta una serie de conceptos si se requiere estar saludable. Algo que preocupa en México, es que las personas le dan poca importancia al cuidado de su salud.

De acuerdo a la OMS (2008) las enfermedades cardíacas, los infartos, el cáncer, las enfermedades respiratorias y la diabetes, son las principales causas de mortalidad en el mundo, siendo responsables del 63% de las muertes, en 2008, 36 millones de personas murieron de una enfermedad crónica, de las cuales la mitad era de sexo femenino y el 29% era de menos de 60 años de edad, otro factor importante en el país es la inseguridad en la que se vive, pues constituye otro gran problema en México. Según la Universidad Nacional Autónoma de México indica que 5 personas son secuestradas diariamente y la cifra probablemente vaya en aumento en los próximos años, por esta razón se han creado dispositivos que permiten controlar algunos de estos problemas.

En el mercado existen dispositivos conocidos como monitores de signos vitales estos dispositivos se utilizan para medir los parámetros fisiológicos básicos y poder notificar a los médicos cualquier cambio en las condiciones del paciente. Dependiendo de su configuración, estas unidades pueden medir y mostrar en pantalla las cifras de PANI (monitores de presión arterial independientes), SpO₂ (cantidad de oxígeno transportado por las células rojas de la sangre) y temperatura [1].

Los monitores de signos vitales se pueden clasificar en dos tipos: 1) Intrahospitalario: se utilizan para trasladar al paciente de un área a otra dentro del mismo hospital o centro de salud. 2) Interhospitalario: Estos se requieren cuando el paciente será trasladado de un hospital a otro una desventaja de estos dispositivos es que deben contar con una conexión de 12V para que puedan ser instalados durante el traslado en ambulancia.

Otro producto existente en el mercado son los monitores de signos vitales portátiles que son equipos biomédicos que permiten detectar, procesar y desplegar de manera continua toda la información de los parámetros fisiológicos de un paciente. Con este tipo de dispositivos se pueden descubrir situaciones adversas o fuera del rango deseado, sirviendo como guías visibles y audibles para todos los médicos que lo utilizan. Cuentan con un tamaño compacto

que los hace aún más sencillo de utilizar y transportar y, gracias a la tecnología algunos monitores de signos vitales portátiles pueden transmitir todos los datos obtenidos a una PC por medio de un puerto de conexiones o bien, sin cables vía Bluetooth, para así no sólo monitorizar, sino analizar, almacenar y comparar toda la información médica de cada paciente [1].

Los dispositivos que están disponibles en el mercado únicamente se limitan a generar estadísticas dentro de los hospitales o para beneficio del médico, no tienen un respaldo de información en la nube y no generan un historial para el familiar del paciente o la persona que se encuentra a cargo de la salud del mismo, por esta razón no se puede tener la información disponible en cualquier momento, además que son difícil de configurar e interpretar sin mencionar su alto costo.

Los productos existentes en el mercado son variados, existen Reloj Smart Watch o pulseras inteligentes que permiten medir la presión arterial, ritmo cardiaco y hasta generar estadísticas de arritmias, pero no es un dispositivo diseñado para el control y la interpretación de las mediciones de los signos vitales, está limitado y no ofrece la capacidad de generar historiales de otros signos vitales ni almacenar la información en la nube. Las pulseras inteligentes y los relojes pueden sincronizarse con el dispositivo móvil pero no está diseñado para enviar tu ubicación y generar estadísticas en tiempo real de la lectura de tus signos vitales, sin contar que no tienen un botón de ayuda que permita enviar una alerta al celular de otra persona [12]. Una persona enferma, adulto mayor o infante necesita ser monitoreado constantemente para conocer su ubicación y tener cuidado con cualquier síntoma anormal que presente de esta forma tratar de recibir atención inmediata. El presente proyecto permitirá al portador del dispositivo electrónico, presionar un botón de ayuda el cual enviara por la red móvil una notificación que se visualizará en el teléfono celular de la persona que se encargue de su cuidado. El brazalete utilizará la red móvil para enviar información como: ritmo cardiaco, temperatura y ubicación, a la persona encargada del cuidado del paciente el cual estará informado y en constante monitoreo. El encargado del cuidado del paciente tendrá la posibilidad de consultar por medio de una app móvil los datos en forma de reporte impreso de los últimos 7 días, podrá interpretar los datos de forma gráfica. La app móvil permitirá al usuario consultar su ubicación y los signos vitales en el momento que el usuario lo requiera.

El dispositivo que se desarrolla como proyecto de investigación está pensado para el cuidado de la salud y para combatir la inseguridad, el público a quien está dirigido son personas enfermas, pero en realidad cualquiera que cuide su salud o se preocupe por la seguridad de sus seres queridos, puede utilizarlo.

1.3. Objetivo General

Diseñar y manufacturar un prototipo de dispositivo electrónico que permita leer datos (ritmo cardiaco, temperatura, ubicación) y enviarlos a un servidor de base de datos, la información almacenada estará ligada a una app móvil que mostrará la información y recibirá notificaciones sobre la variación de los signos vitales o la solicitud de ayuda generada por el dispositivo.

1.4. Objetivos Específicos

- Crear un prototipo electrónico llamado Ubic's que permite recolectar y enviar datos de ritmo cardiaco, temperatura y ubicación por la red a un servidor de base de datos.
- Crear una base de datos en un servidor de hosting que permita almacenar la información de los signos vitales recolectados por el dispositivo y ligar la información a un usuario que se encargará del monitoreo.
- Crear un sistema web que permita el registro de los usuarios cuidadores y los dispositivos Ubic's para crear una cuenta de administración y vincular los datos.
- Crear una app móvil que permitirá a los usuarios y portadores interpretar la información generada por el dispositivo electrónico, mostrar gráfica de las últimas 5 lecturas, generar alertas por variación en los signos vitales, generar alertas de ayuda solicitadas por el portador de Ubic's, verificar la ubicación en tiempo real del portador del dispositivo electrónico en cualquier momento.

1.5. Justificación

La presente investigación se enfocará en desarrollar un dispositivo electrónico que permite tomar la temperatura corporal, calcular el número de latidos por minuto de una persona y obtener la ubicación. Este dispositivo enviará la información por medio de la red GPRS a un servidor de internet que almacenará de forma constante los valores en una base de datos.

La base de datos creada y almacenada en el servidor permitirá organizar la información de los pacientes o portadores del dispositivo, sus signos vitales generados en tiempo real y las personas encargadas de su cuidado, el administrador utilizará un sistema web que permitirá tomar los datos del dispositivo y asignar a un cuidador agregando permisos de acceso por medio de un sistema web.

Una vez registrada esta información el cuidador podrá descargar una app móvil e ingresar con un usuario y contraseña, la app mostrará una lista de dispositivos vinculados a su cuenta las cuales pueden ser monitoreadas por el cuidador indicando su temperatura actual, su número de latidos y generar reportes semanales de los datos registrados por medio del dispositivo electrónico, además podrá conocer la ubicación de la persona que porta el dispositivo para saber dónde se encuentra.

La tecnología en la actualidad tiene que servir como un apoyo. Al realizar e implementar este proyecto, ayudaremos a personas que no pueden estar las 24 horas monitoreando a su familiar, además el dispositivo está planeado para pacientes con enfermedades crónico degenerativas que necesitan un monitoreo constante sin embargo este no se limita a ser usado solo por ellos ya que el módulo de ubicación incorporado ayuda a que cualquier personas que necesite de nuestro monitoreo específicamente niños y adolescentes que requieren de un seguimiento puntual y atención por parte de sus padres o familiares.

1.6. Motivación Para Elaborar la Investigación

En México existe un índice elevado de personas que presentan una enfermedad crónica degenerativa, cada día tienen que tener cuidados personales de su salud para lograr llevar una mejor calidad de vida, muchas ocasiones en las clínicas y hospitales llegan personas de la tercera edad a solicitar el servicio de monitoreo de glucosa, temperatura, presión arterial con la finalidad de llevar un control adecuado, sin embargo estas personas no tienen la fuerza suficiente para asistir diariamente a su clínica, por el clima, distancia o dada la situación en la que se encuentran pueden perderse en la ciudad e incluso perder la memoria.

En algún momento nos hemos preocupado por el cuidado de un familiar con una enfermedad crónica degenerativa o simplemente porque necesitamos que nuestros padres, hijos, hermanos, abuelos y tíos se encuentren seguros. El trabajo diario y las actividades cotidianas no nos permiten pasar mucho tiempo con la familia ni dedicarle el cuidado o la atención que ellos necesitan. Por poner un ejemplo muchas personas trabajan y no son capaces de poder llevar a sus hijos al colegio y se vive con la incertidumbre diaria de saber si llegaron con bien a la escuela o si durante sus clases no se sentían mal.

La motivación de realizar el presente proyecto no solo es para desarrollarme profesionalmente, también como padre de familia me preocupa si mis hijos ya salieron del colegio, si se trasladaron de forma adecuada a la casa, si no presenta alguna irregularidad en su salud y como persona que tiene una enfermedad crónica degenerativa me hubiese gustado que existiera un dispositivo que me permitiera obtener estos datos y llevar un mejor control de los parámetros fisiológicos de mi cuerpo.

Pensando en los factores que se presentan en el país, situaciones complicadas de salud, falta de atención médica, apoyo a personas de la tercera edad y la inseguridad que se vive se propone la idea de proyecto que funcionará como un asistente de monitoreo que me avisa cuando un familiar está en una situación delicada o que al presionar un botón de ayuda ubicado en el dispositivo tendría una alerta en mi celular para saber su temperatura, ritmo cardiaco, su ubicación y brindar el apoyo necesario en el mejor tiempo posible.

1.7. Organización de la Tesis

El proyecto de investigación presentado en este documento consta de 4 capítulos los cuales describen de forma muy puntual cada una de las etapas del proyecto:

- Capítulo 1: Introducción daremos a conocer todos los detalles del proyecto, estado del arte, justificación, hipótesis general, objetivos del proyecto y motivación para elaborar la investigación.
- Capítulo 2: Marco conceptual se realiza un análisis y recopilación teórico práctica del proyecto abordando los temas que se relacionan con los signos vitales y la importancia de estos en el cuidado de la salud, parámetros fisiológicos y rangos normales, monitores de los signos vitales utilizados en hospitales o en la vida cotidiana, proyectos de investigación dedicados a dar una solución similar, ventajas de nuestro proyecto y propuestas de mejora frente a los ya existentes.
- Capítulo 3: Metodología en este capítulo se lleva a cabo una serie de métodos y técnicas que se utilizan en el desarrollo del proyecto iniciando desde el análisis, diseño del proyecto (diseño del prototipo electrónico, diseño del sistema de administración web, diseño de la app móvil para el monitoreo), desarrollo de cada uno de los módulos incluida la programación e implementación y puesta a punto del proyecto.
- Capítulo 4: Resultados y conclusiones en esta parte daremos a conocer el proyecto final, los resultados de la ejecución del mismo, las áreas de mejora, las oportunidades del proyecto y los trabajos futuros.

1.8. Publicaciones que Sustenten el Trabajo de Tesis

Como parte del desarrollo del proyecto se presenta un artículo en la revista de investigación en tecnologías de la información (RITI) Vol. 9, No 18, Enero Julio, titulada Monitor de temperatura, ritmo cardiaco y ubicación enviando mensajes de texto a un teléfono celular presentado por: Ricardo Luna Santos, Aldo Hernández Luna, Jacinto Torres Jiménez, Manuel Cruz Luna. ver Anexo 1

CAPITULO 2

MARCO TEÓRICO

2.1. Marco Conceptual

2.1.1. Signos vitales

El control de signos vitales es una práctica que permite el control en los cuidados del ser humano y aporta información objetiva sobre la estabilidad cardiorrespiratoria, hemodinámica y térmica del paciente. Los signos vitales son hallazgos percibidos por un explorador capacitado obteniendo datos objetivos. Se modifican según la edad, el sexo, el peso, y la presencia de enfermedad.

Los signos vitales son la determinación de la frecuencia cardiaca, la frecuencia respiratoria y la temperatura corporal. Aunque no se considera estrictamente un signo vital, también se incluye la presión arterial. Las anomalías en los signos vitales tienen una gran importancia para el diagnóstico de ciertas enfermedades, y su modificación se utiliza para evaluar la evolución del paciente [1].

Los signos vitales son indicadores que reflejan el estado fisiológico de los órganos vitales (cerebro, corazón, pulmones). Expresan de manera inmediata los cambios funcionales que suceden en el organismo, cambios que de otra manera no podrían ser cualificados ni cuantificados [3].

2.1.2. Parámetros fisiológicos

Dentro de los parámetros fisiológicos que necesitamos monitorear tenemos a la temperatura corporal que es el resultado del equilibrio entre la producción y la pérdida de calor del cuerpo. Se puede medir la temperatura corporal central: rectal, esofágica y en la arteria pulmonar, la temperatura corporal periférica, axilar o inguinal, y la temperatura de la piel, mediante un sensor fijado a la piel. El control consiste en la medición de la temperatura para evaluar la función termorreguladora la cual tiene que estar entre los 36,5°C y los 37°C según la Organización Mundial de Salud cualquier valor fuera de este rango se considera fiebre.

Otro parámetro importante es la frecuencia cardiaca que es el número de contracciones por unidad de tiempo, en otras palabras, es el conteo de contracciones cardíacas durante un minuto. Los rangos normales varían según edad, sexo y estado físico por ejemplo en un adulto hombre es de 70 latidos por minuto, en una mujer adulta es de 80 latidos por minuto en caso de obtener lecturas inferiores a 60 latidos por minuto se considera bradicardia y es una

alteración, en caso de que las pulsaciones sean superiores a 100 latidos por minuto se considera taquicardia y se necesita atención en ambas situaciones.

2.1.3. Monitores de signos vitales

Con la finalidad de medir los signos vitales de los pacientes se utilizan dispositivos existentes en el mercado la única desventaja es que se limitan a controlar cierta información, pero aún no existe un dispositivo que permita controlar la mayoría de signos vitales o necesarios del cuerpo, los dispositivos no fueron creados con esa funcionalidad ni mucho menos para el control y monitoreo remoto de los signos vitales.

Algunas de las herramientas utilizadas para el monitoreo de los signos vitales que existen son los sensores o bio-sensores (denominados así por estar directamente relacionados con parámetros biológicos o químicos del ser humano) que durante los últimos años se han intentado varias metodologías, una de ellas es implantarlos. El implante cae dentro de la categoría “wearable” porque la interacción con el usuario es constante a pesar del factor invasivo, sin embargo, estas cirugías son muy pequeñas y rápidas, las cuales puede evitar incomodidades posteriores. Algunas aplicaciones para esta técnica pueden ser: medir los niveles de glucosa de la sangre, evitando el pinchazo diario para los pacientes con diabetes, medir temperatura corporal interior, monitorear órganos específicos como corazón o hígado, etc. [10].

La obtención de la señal por estos sensores debe ser transformada de una señal análoga a una digital, la exactitud de la transformación depende de la cantidad de bit que posea el microcontrolador, para luego procesar la información y finalmente se envía hacia un servidor central o a un dispositivo electrónico de forma inalámbrica. Los bio-sensores en general nos proveen de información importante del estado de salud de los pacientes, dependiendo de esta información el personal clínico determinará cual es el diagnóstico y cuáles son los pasos a seguir para tratar al paciente y lograr su recuperación.

A continuación, daremos a conocer algunos proyectos que se desarrollaron con la finalidad de atender esta necesidad y describiremos cómo funcionan.

2.1.4. Proyectos de investigación

La universidad técnica de Ambato facultad de ingeniería en sistemas electrónica e industrial carrera de ingeniería en electrónica y comunicaciones presenta el tema: “sistema ambulatorio

para el control de signos vitales y prevención de la diabetes mellitus” como trabajo de graduación modalidad: proyecto de investigación, presentado previo la obtención del título de ingeniero en electrónica y comunicaciones sub línea de investigación: comunicaciones inalámbricas autor: Diana Mary Anchaluisa Abril tutor: ing. Marco Antonio Jurado Lozada, mg. Ambato - ecuador octubre – 2018 con el objetivo de implementar un prototipo de un sistema ambulatorio para el control de signos vitales y prevención de la diabetes mellitus [6]. La universidad de concepción facultad de ingeniería departamento de ingeniería eléctrica con ayuda del profesor patrocinante: DSC. Pablo Squeveque n. se desarrolló un informe de memoria de título para ingeniero civil biomédico el proyecto sistema de monitoreo continuo de signos vitales con sensores no invasivos y transmisión inalámbrica de datos concepción, marzo de 2016 por Christopher Andrés Gutiérrez. El proyecto consiste en Desarrollar un sistema de monitoreo continuo de variables fisiológicas clínicas con sensores no invasivos y transmisión inalámbrica de datos [11].

Otra técnica que se utiliza en el tema de monitorización es el “wearable”, Desarrollar sistema de monitoreo “wearable” para medición de las principales variables fisiológicas (electrocardiograma, frecuencia respiratoria y temperatura corporal) y variables ambientales como temperatura y humedad relativa, es una forma cómoda y fácil de utilizar. Implementar sistema de monitoreo con sensores no invasivos y transmisión inalámbrica de datos, realizar interfaz gráfica para visualización de los datos adquiridos por el equipo y almacenamiento de estos para su análisis.

La Revista Ingeniería Biomédica publica un Sistema de transmisión inalámbrica de señales ECG y de temperatura para ambientes hospitalarios (SINHO) Dora María Ballesteros Ψ , Harvey Enrique Melo, Ascanio José Maya Quintero Grupo de Investigación TIGUM, Universidad Militar Nueva Granada, Colombia [5].

La investigación se basa en presentar el diseño, desarrollo y pruebas de un dispositivo multicanal de adquisición y transmisión de señales electrocardiográficas y de temperatura, utilizando tecnologías de transmisión inalámbrica (Bluetooth, ZigBee, Radiofrecuencia RF), con una interfaz de usuario en una central de monitoreo. Se analizó el desempeño del sistema de acuerdo a su disponibilidad (97%), tiempo de conexión (6 segundos) y precisión en las mediciones (98%).

Sistema Telemédico de electrocardiografía basado en internet embebido presentado por Julio Cruz, Guillermo Villegas, Jorge Uyá, Alfredo Hernández, Gianfranco Passariello, Fernando Mora del laboratorio de sistemas biomédicas, grupo de bioingeniería y biofísica aplicada universidad simón bolívar, caracas, Venezuela presentan un proyecto en el que se describe un sistema telemédico de electrocardiografía vía internet totalmente comunicante. el sistema está compuesto de un módulo de adquisición de datos médicos, un módulo de internet embebido que permite la transmisión directamente a una intranet o internet de los datos médicos adquiridos y estaciones local y central basadas en la arquitectura pc donde corren clientes para monitorizar las señales. El uso del protocolo TCP/IP y la arquitectura cliente/servidor en esta aplicación telemédica, permite explotar al máximo la flexibilidad y alcance de internet [8].

Actualmente las tecnologías de la información y las comunicaciones han permitido el diseño de complejos sistemas de transmisión, recepción y análisis de señales de origen biológico. Este tipo de sistemas se caracterizan por monitorizar las señales biológicas y establecer un control continuo de los diferentes parámetros fisiológicos en los pacientes, además de contribuir a la prevención de enfermedades [7]. Los sistemas de biotelemedicina debido a su complejidad y costo tienen la desventaja de ser poco asequibles por poblaciones en lugares apartados a las entidades de salud, necesidad que ha conllevado a los investigadores del grupo de investigación TIGUM a desarrollar una arquitectura de biotelemedicina operativa y funcional a corto alcance, basada en plataforma Arduino® Mega ADK., útil para el diagnóstico y prevención de enfermedades mediante el registro y trazabilidad de la información de tres signos vitales. La arquitectura utiliza las tecnologías Bluetooth y ZigBee para facilitar la transmisión y la visualización de las señales en diferentes equipos electrónicos.

2.1.5. Ventajas del proyecto

Como podemos verificar en los proyectos anteriores, todos estamos interesados en controlar la salud de las personas con diferentes aportes, pero en realidad la base de los proyectos es analizar los parámetros que permiten el control o diagnóstico de enfermedades en el organismo, las aplicaciones que existen en el mercado están limitadas y ninguna ofrece la capacidad de geolocalización del paciente ni el monitoreo en tiempo real por medio de un dispositivo móvil, las ventajas que ofrece el presente proyecto de investigación en

comparación con las soluciones anteriores son visibles y se planea obtener resultados que garanticen el control del paciente y el cuidado del individuo logrando preservar su seguridad.

CAPITULO 3

MARCO METODOLÓGICO

3.1. Metodología

En el presente proyecto se describen una serie de métodos y técnicas utilizadas para el desarrollo de la investigación, esto permitirá alcanzar los resultados e implementar los procedimientos adecuados para el cumplimiento de los objetivos del proyecto.

Dentro de las metodologías consultadas para el desarrollo de este proyecto se optó por utilizar “Extreme Programming” o “Programación Extrema” como metodología de desarrollo ya que nos permite establecer tiempo de entrega y agilizar el proceso de desarrollo gracias al número de iteraciones cortas del proyecto tal como se muestra en la Figura 3.1



Figura 3.1 Etapas del desarrollo del proyecto utilizando la metodología XP (Programación Extrema)

Las ventajas que ofrece como metodología de desarrollo son el desarrollo de trabajo ágil que tiene como finalidad la entrega de valor en períodos cortos de tiempo y para ello se basa en tres pilares: la transparencia, inspección y adaptación.

- La transparencia permite dar a conocer toda la información correspondiente al proyecto a las partes interesadas o que integran el grupo de desarrollo.

- La inspección garantiza que las revisiones sean periódicas con la finalidad de detectar y corregir fallos.
- La adaptación tiene que ver con la característica del equipo de trabajo para ajustar tiempos y lograr dichos objetivos.

3.1.1. Análisis

La identificación de una necesidad, enunciada en términos concretos, es el punto de partida para la puesta en marcha de un proyecto y la evaluación de las posibles soluciones que dará la viabilidad del mismo [11]. Para el análisis se requiere de tres etapas fundamentales:

- Conceptualización
- Planeación
- Requerimientos del proyecto

3.1.2. Diseño

El diseño comprende cuatro tipos de actividades: diseño de datos, arquitectónico, procedimental y diseño de interfaces, desde el punto de vista del proyecto evoluciona desde un diseño preliminar al diseño detallado [11].

- Diseño de casos de uso
- Diseño de Diagramas de Flujo
- Diseño eléctrico y electrónico del dispositivo
- Modelado 3D
- Diseño de la Base de Datos
- Diseño de interfaces del software

3.1.3. Desarrollo

Crear los datos de prueba, interpretar código fuente, implementar el código fuente, generar la documentación, planificar y realizar la integración de los módulos [11].

- Creación del circuito e instalación de los sensores
- Programación del microcontrolador y sus componentes
- Programación del sistema de control
- Desarrollo de la app Móvil

3.1.4. Pruebas

Las tareas que abarca son las siguientes: pruebas de verificación, revisiones y auditoría e incluye las tareas de validación y pruebas de validación que se realizan durante el ciclo de vida del software para asegurar la satisfacción con los requisitos [11].

- Verificación del envío de los datos del dispositivo a la red móvil
- Verificación de la ubicación desde la interfaz del administrador
- Verificación de los reportes del sistema
- Verificación de la App Móvil conectada al dispositivo y a la red
- Verificación de las alertas
- Verificación del botón de ayuda y el envío de los datos

3.1.5. Puesta a punto del prototipo

Implementación del sistema y todos sus componentes en un entorno real de funcionamiento y con sujetos de pruebas para la obtención de resultados satisfactorios.

- Configuración y puesta a punto del servidor o hosting y base de datos
- Instalación del sistema web en el servidor
- Instalación de la App Móvil en un dispositivo con sistema operativo Android

3.2. Métodos y Procedimientos

3.2.1. El análisis y la planeación

La planificación del proyecto es una de las etapas más importantes para la ejecución de las actividades, basados en nuestra metodología los resultados deben considerarse de forma rápida y bien estructuras, como se muestra en la Figura 3.2 y Figura 3.3 se consideren los tiempos estimado y el control del avance del proyecto, por este motivo presentaremos el cronograma con la lista de actividades programadas y entregadas del prototipo.

3.2.1.1. Cronograma de actividades

Id		Modo de tarea	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras
1			Analisis	67 días	lun 01/06/20	mar 01/09/20	
2			Recopilar información del proyecto con sus	10 días	lun 01/06/20	vie 12/06/20	
3			Establecer Objetivos	5 días	lun 15/06/20	vie 19/06/20	2
4			Crear diagrama de GANT	10 días	lun 22/06/20	vie 03/07/20	3
5			Determinación de los requerimientos de la información	10 días	lun 06/07/20	vie 17/07/20	4
6			Análisis de las necesidades del software	5 días	lun 20/07/20	vie 24/07/20	5
7			Determinar los requerimientos del prototipo	5 días	lun 27/07/20	vie 31/07/20	6
8			Definir el presupuesto	10 días	lun 03/08/20	vie 14/08/20	7
9			Entrega, revision y corercciones por el Asesor	12 días	lun 17/08/20	mar 01/09/20	8
10			Diseño	45 días	mar 01/09/20	sáb 31/10/20	
11			Diseño de casos de us	15 días	mar 01/09/20	lun 21/09/20	
12			Diseño de Diagramas de Flujo	10 días	mar 22/09/20	lun 05/10/20	11

Figura 3.2 Etapa de análisis de la información

Id		Modo de tarea	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras
13			Diseño de la Base de Datos	9 días	mar 06/10/20	vie 16/10/20	12
14			Diseño eléctrico y electrónico del dispositivo	10 días	lun 19/10/20	vie 30/10/20	13
15			Diseño de interfaces del software	6 días	lun 02/11/20	lun 09/11/20	14
16			Desarrollo	87 días	dom 01/11/20	dom 28/02/21	
17			Creación del circuito e instalación de los sensores	15 días	dom 01/11/20	jue 19/11/20	
18			Programación del microcontrolador y sus componentes	25 días	vie 20/11/20	jue 24/12/20	17
19			Programación del sistema de control y la App Movil	46 días	vie 25/12/20	vie 26/02/21	18
20			Pruebas	66 días	lun 01/03/21	lun 31/05/21	
21			Envío de los datos del dispositivo a la red móvil	10 días	lun 01/03/21	vie 12/03/21	
22			Conexión de la App Móvil al dispositivo y a la red	15 días	lun 15/03/21	vie 02/04/21	21
23			Ubicación desde la interfaz del administrador	10 días	lun 05/04/21	vie 16/04/21	22
24			Reportes del sistema	6 días	lun 19/04/21	lun 26/04/21	23
25			Notificaciones y Alert	10 días	mar 27/04/21	lun 10/05/21	24
26			botón de ayuda	10 días	mar 11/05/21	lun 24/05/21	25
27			Puesta a Punto del Prototipo	11 días	mar 25/05/21	mar 08/06/21	26
28			Artículo Científico	31 días	mié 09/06/21	mié 21/07/21	
29			Tesis	320 días	lun 01/06/20	vie 20/08/21	

Figura 3.3 Etapas de Diseño, Desarrollo y pruebas

3.2.1.2. Listado de herramientas, materiales y software

En la tabla 1 se muestran las herramientas que se utilizaran para el desarrollo del prototipo, materiales que se consideran para la implementación, equipos y software necesario para la aplicación véase Tabla 3.1

Como primer procedimiento realizaremos la presentación y especificación de las herramientas de software y hardware que permitirán realizar el desarrollo del proyecto.

Material	Descripción	Cantidad
Arduino Nano	Es una placa de creación de electrónica de código abierto, la cual está basada en hardware y software libre, flexible y fácil de utilizar para los creadores y desarrolladores	1
Sensor MAX30102	Módulo integrado detector que mide la frecuencia del pulso.	1
Sensor MLX90614	Es un termómetro infrarrojo para mediciones de temperatura sin contacto	1
Módulo GPS6MV2	Modulo electrónico que utiliza el sistema de posicionamiento global (GPS) para determinar la localización exacta de un vehículo, persona, u otro activo	1
Batería Recargable	Batería de Litio Tipo 7.4 Volts A 1500 mAh	1
Protoboard	Es un tablero con orificios que se encuentran conectados eléctricamente entre sí de manera interna, habitualmente siguiendo patrones de líneas, en el cual se pueden insertar componentes electrónicos y cables para el armado y prototipado de circuitos electrónicos y sistemas similares	1
Cable Jumpers	cables que pueden tener terminales macho o hembra y que permiten interconectar circuitos	20
Chip de telefonía móvil SIM	Esta tarjeta permite almacena de manera segura tu número de teléfono, así como las claves de acceso de un usuario concreto en una operadora de telefonía y servicios móviles de la red.	1
Material	Descripción	Cantidad
Equipo de computo	Dispositivo electrónico que permite procesar información y será de gran ayuda para la configuración, programación, desarrollo y pruebas del prototipo	1

IDE de Arduino	Es un programa o aplicación de software para escribir y cargar programas en placas compatibles con Arduino	1
Net Beans IDE	Es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java, que se puede utilizar para desarrollar software multiplataforma y escribir programas de cómputo a la medida	1
Android Studio IDE	Es el entorno de desarrollo integrado (IDE) oficial para el desarrollo de apps para Android	1
Paquetería Office	Es una suite ofimática que abarca el mercado completo en Internet e interrelaciona aplicaciones de escritorio, servidores y servicios para los sistemas operativos Microsoft Windows, Mac OS X, iOS y Android.	1
Simulador tinkercad	Es un programa gratuito de modelado 3D en línea que se ejecuta en un navegador web, conocido por su simplicidad y facilidad de uso.	1
Windows 10 profesional	Sistema operativo que permitirá ejecutar múltiples aplicaciones y software para el desarrollo del prototipo	1
Acceso a Internet	Servicio que ofrecen los operadores para navegar y acceder a los recursos de la red mundial	1
Web Hosting	Servicio de alojamiento de sitios web dinámicos y sistemas informáticos para su acceso por internet	1
Cargador de Baterías IMAX B6 AC/DC	Permite cargar baterías, LiPo , LiFe, NiCd y NiMH. Gracias a su conexión AC 100 ~ 240 V y 12V DC se puede cargar tanto en casa como en el campo de vuelo conectándolo a la batería del coche.	1

Tabla 3.1. Relación de materiales, herramientas y software

3.2.1.3. Listado de costos

En la Tabla 3.2 se muestra la relación de costos del material utilizado para el desarrollo de prototipo.

MATERIAL	COSTO
Arduino Nano	\$97.00
Sensor MAX30102	\$93.00
Sensor MLX90614	\$222.00
Módulo GPS6MV2	\$323.00
Batería Recargable	\$450.00
Protoboard	\$90.00
Cable Jumpers	\$50.00
Chip de telefonía móvil SIM	\$100.00
Acceso a Internet	\$4800.00
PCB	\$300.00
Web Hosting	\$1500.00
Total	\$8025.00

Tabla 3.2. Relación de gastos de inversión del proyecto

3.2.1.4. Análisis de funcionamiento del proyecto

La comunicación de los sensores con el microcontrolador o la placa Arduino se llevará por medio de circuitos eléctricos, cables o jumpers, posteriormente se genera una PCB para la conexión de los sensores en la placa.

La comunicación para el envío de la información del dispositivo electrónico a un servidor de base de datos será por medio de GPRS, es una tecnología probada y funcional que permite conectividad total, Acceder a Internet, Conectarte con cámaras digitales (capturar y enviar imágenes), Conectarte con reproductores de MP3, Acceder a contenidos informativos y servicios en línea vía WAP, en cualquier momento y lugar.

La tecnología GPRS permite:

- **Concurrencia:** GPRS permite un uso concurrente entre la voz y datos. Así, en el momento que se recibe o se realiza una llamada, la comunicación de datos se interrumpe, luego se vuelve a reanudar en forma automática cuando la llamada de voz se termina.
- **Conexión Permanente:** Con GPRS, desde que se encienden los dispositivos móviles están preparados para realizar transmisión de datos. No se requiere conectar y desconectar una llamada, lo que hace esta tecnología más cómoda y efectiva.
- **Velocidad de Transmisión:** Esta tecnología permite conseguir en condiciones óptimas rendimientos de hasta 5 veces la velocidad máxima de GSM. Así es posible desarrollar múltiples servicios con un mayor y mejor contenido de imágenes, sonido y video. Además, posibilita en el futuro la creación de otras aplicaciones en base a esta plataforma.

Además, GPRS permite:

- Acceso a correo electrónico por Internet.
- Servicios de mensajería y comunicación por grupos.
- Servicios de información.
- Juegos.
- Comercio electrónico.
- Servicios financieros.
- Transferencia de imagen, audio y video.
- Navegación por Internet.
- Y mucho más.

El servidor de base de datos estará implementado en un web hosting y la base de datos se genera en MySQL en esta se almacena información enviada desde el dispositivo electrónico con la ayuda de servicios web, los cuales permitirá comunicar la app, y el dispositivo electrónico para el almacenamiento y consulta de la información.

Finalmente se desarrollará un software basado en la nube el cual funcionará como administrador de la base de datos y permitirá asignar administradores a la información recolectada por el dispositivo electrónico formando así un clico de administración, almacenamiento y consulta de la información tal como se presenta en la Figura 3.4

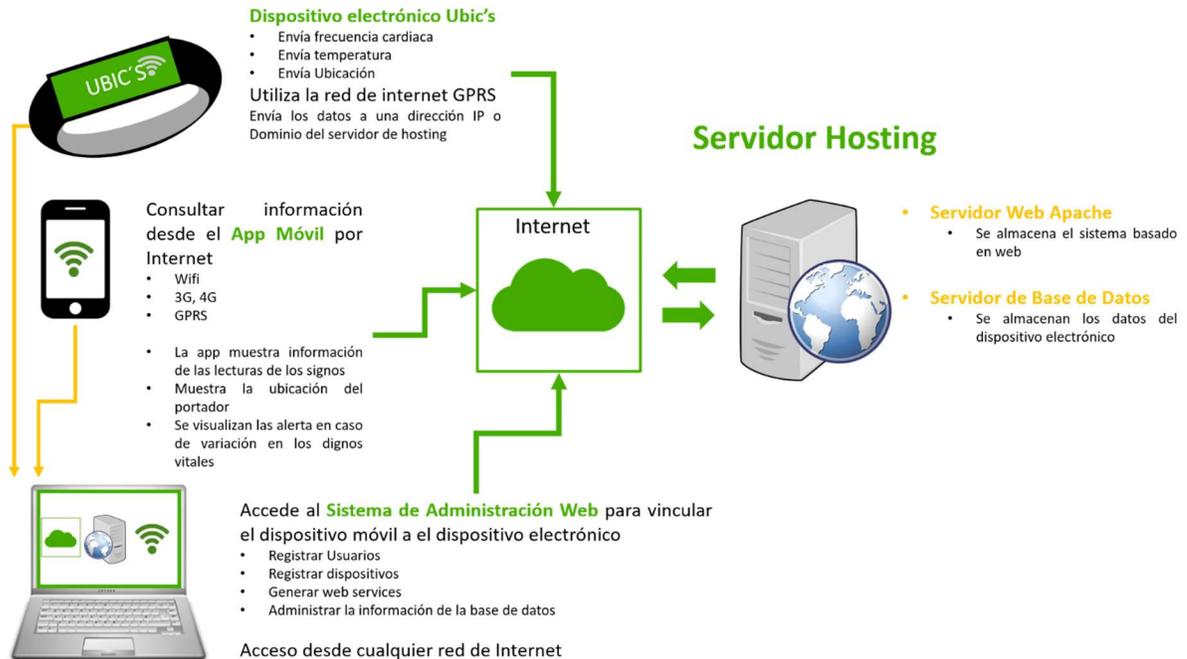


Figura 3.4 Ciclo de envío, consulta, vinculación y proceso de los datos en el prototipo

3.2.2. Diseño del proyecto

3.2.2.1. Diseño de casos de uso

Los diagramas de caso uso nos permiten identificar los componentes principales que forman el sistema y nos ayuda a recolectar los requerimientos del sistema en pocas palabras cuales son las funciones que realiza nuestro sistema.

Al crear un caso de uso se determinan que los actores que intervienen son tres, el administrador, el dispositivo electrónico y la app móvil. Los casos de uso que corresponden al administrador del sistema permiten al administrador la posibilidad de registrar, buscar, modificar y eliminar usuarios y dispositivos. Una vez registrados los usuarios se podrá vincular a los usuarios que son los que utilizaran la App Móvil para monitorear los dispositivos electrónicos que enviaran los signos vitales.

El dispositivo electrónico se encargará de enviar la información a la base de datos la cual se almacenará y estará lista para ser consultada por los usuarios de la App Móvil misma que estará vinculada al dispositivo, por lo tanto, el caso de uso que corresponde a este actor es almacenar información.

El último actor que tenemos es la App Móvil la cual estará ligada a un usuario previamente registrado por el administrador y vinculado a un dispositivo para poder consultar información que ya existe en la base de datos véase la Figura 3.5

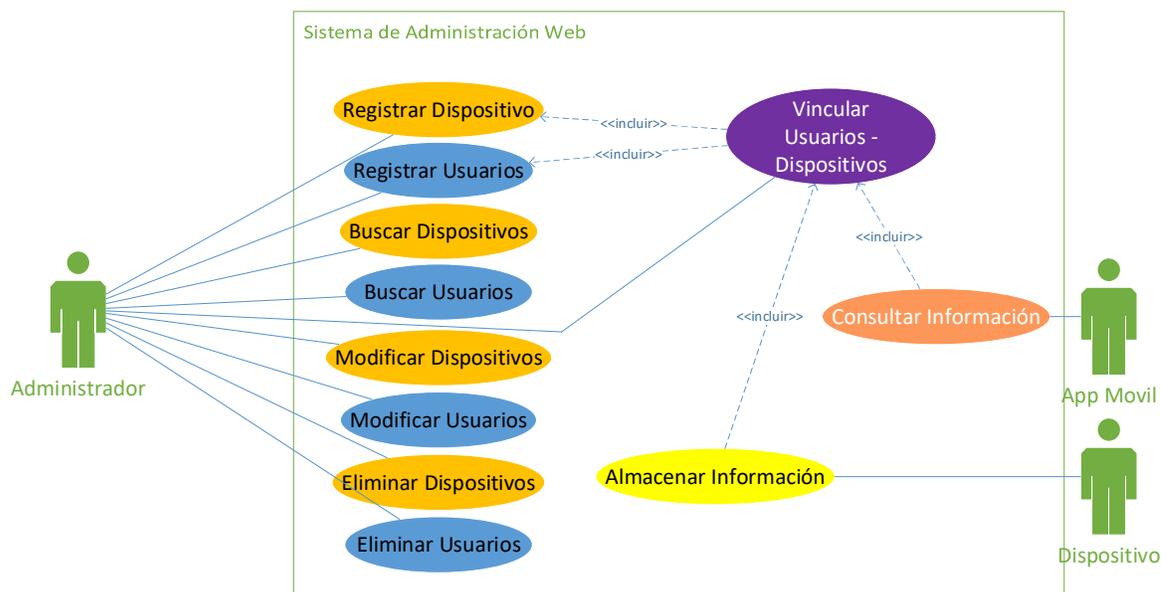


Figura 3.5 Casos de uso del sistema web, app móvil y dispositivo

3.2.2.2. Diagramas de flujo

Esta herramienta se utiliza para representar los procesos del sistema informático, es de gran importancia ya que permite documentar, planificar, mejorar y comunicar el proceso de forma clara, para su codificación y análisis de uso en la Figura 6 el diagrama de flujo representa las acciones que ejecutara el administrador como son registro de dispositivos, registros de usuarios, actualización, modificación y eliminación de registros o en su defecto bloqueo, bajas y vinculación de usuarios a los dispositivos electrónicos, una vez realizada esta acción los dispositivos electrónicos tendrán la posibilidad de enviar lecturas de temperatura, ritmo cardiaco y ubicación, mientras que los usuarios vinculados al dispositivo podrían monitorear por medio de una App los datos registrados por el dispositivo en la base de datos siempre y cuando está vinculado tal como se muestra en la Figura 3.6

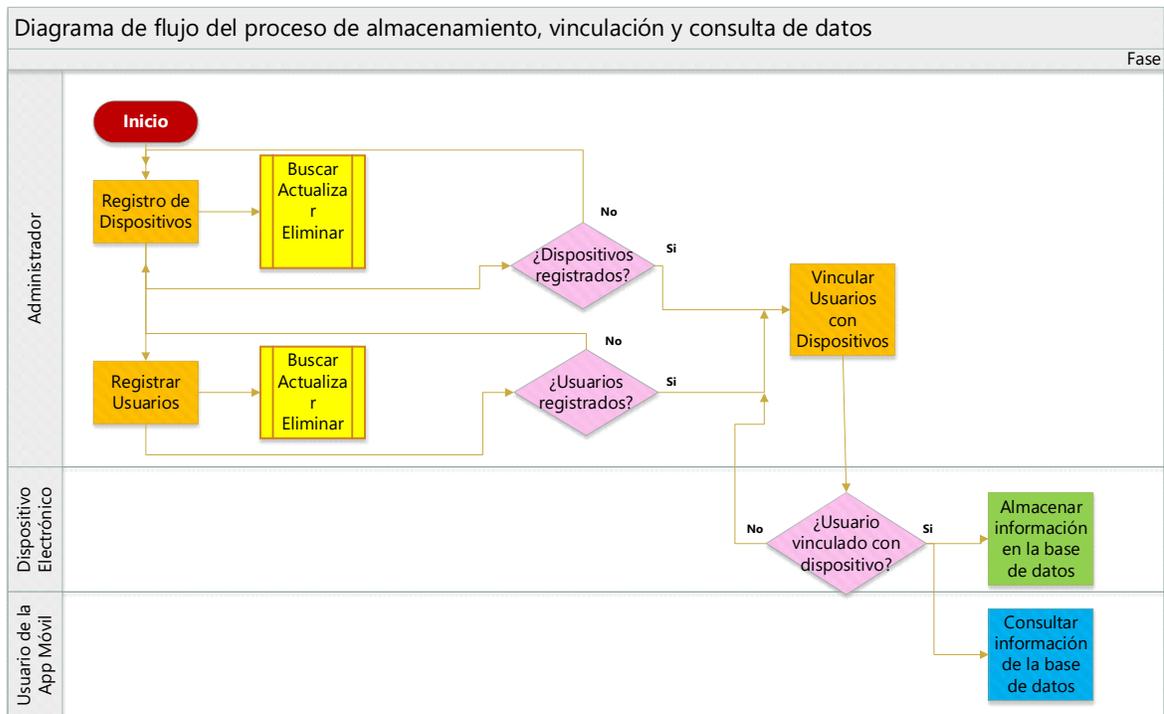


Figura 3.6 Proceso de almacenamiento, vinculación y consulta de datos

3.2.2.3. Diseño de la base de datos

En el mundo del desarrollo de software informático es de suma importancia considerar el almacenamiento de la información por esta razón se diseña la base de datos, como primer paso se genera el modelo entidad relación: El modelo entidad relación permite representar de manera simplificada los componentes que participan en el flujo de la información.

En el modelo entidad relación presentado para el almacenamiento de los datos colocamos 3 entidades las cuales representan un objeto o concepto del mundo real con existencia independiente y con ciertas características a las cuales denominaremos atributos, entre las relaciones existentes en el diagrama se puede visualizar que un asistente está vinculado directamente con un dispositivo y su cardinalidad indica que un asistente puede estar vinculados a muchos dispositivos y un dispositivo puede estar asociado a varios asistentes, el dispositivo vinculado genera múltiples registros, un registro está vinculado a un solo dispositivo, por lo tanto la información será mostrada o consultada solo por el asistente vinculado al dispositivo véase en la Figura 3.7

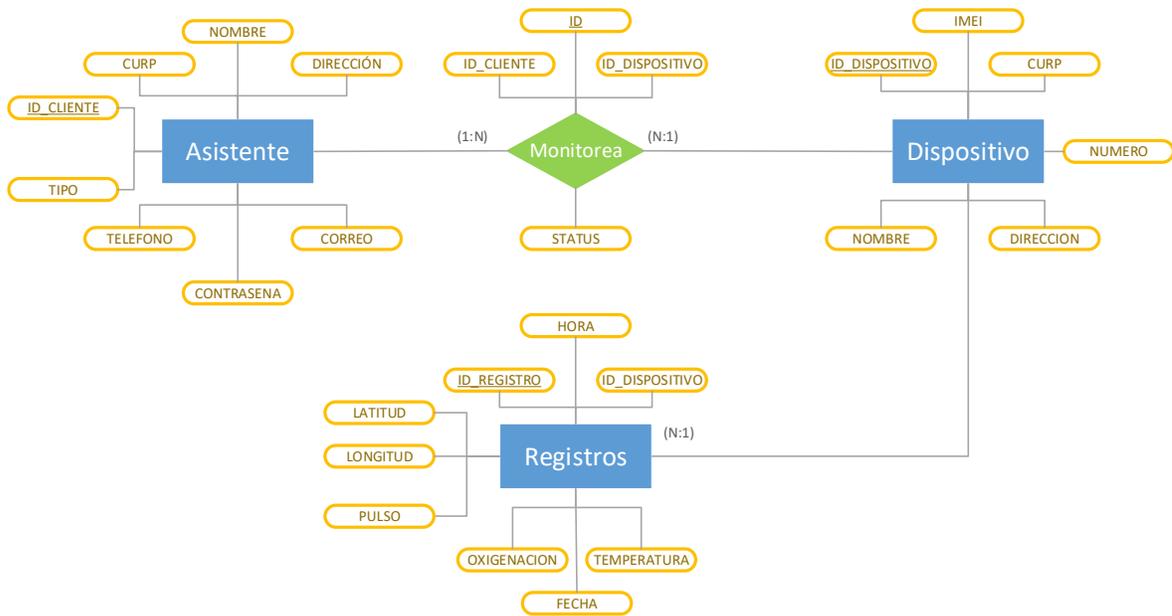


Figura 3.7 Modelo Entidad Relación de la base de datos

3.2.2.4. Modelo relacional

Este modelo permite lograr almacenar los datos de forma efectiva y organizar la información en tablas utilizando filas o duplas y columnas o campos de esta forma garantizaremos la correcta administración de la base de datos, las relaciones y estructura de almacenamiento respeta la normalización tal como se muestra e la Figura 3.8



Figura 3.8 Modelo relacional de la base de datos

3.2.2.5. Diccionario de datos

A continuación, daremos a conocer un listado organizado de todos los datos que pertenecen al sistema, definiremos las características de los campos de forma correcta y rigurosa para que el usuario como el analista tenga un entendimiento común de todas las entradas, salidas, componentes de los almacenes y cálculos intermedios en la aplicación.

Columna	Tipo	Nulo	Predeterminado	Comentarios
id_cliente (<i>Primaria</i>)	int(11)	No		
curp	varchar(25)	No		
nombre	varchar(50)	No		
dirección	varchar(150)	No		
teléfono	varchar(15)	No		
correo	varchar(50)	No		
contraseña	varchar(50)	No		
tipo	varchar(15)	No		

Tabla 3.3 cliente

Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
PRIMARY	BTREE	Sí	No	id_dispositivo	4	A	No	

Tabla 3.4 Índices del cliente

Columna	Tipo	Nulo	Predeterminado	Comentarios
id (<i>Primaria</i>)	int(11)	No		
id_cliente	int(11)	No		
id_dispositivo	int(11)	No		
status	varchar(10)	No		

Tabla 3.5 cliente-dispositivo

Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
PRIMARY	BTREE	Sí	No	Id	2	A	No	
id_cliente	BTREE	No	No	id_cliente	1	A	No	
id_dispositivo	BTREE	No	No	id_dispositivo	2	A	No	

Tabla 3.6 Índices de tabla cliente dispositivo

Columna	Tipo	Nulo	Predeterminado	Comentarios
id_dispositivo (<i>Primaria</i>)	int(11)	No		
imei	varchar(25)	No		
numero	varchar(15)	No		
curp	varchar(25)	No		
nombre	varchar(100)	No		
dirección	varchar(150)	No		

Tabla 3.7 dispositivo

Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
PRIMARY	BTREE	Sí	No	id_dispositivo	4	A	No	

Tabla 3.8 Índices de tabla dispositivo

Columna	Tipo	Nulo	Predeterminado	Comentarios
id_registro (<i>Primaria</i>)	int(11)	No		
id_dispositivo	int(11)	No		
latitud	varchar(15)	Sí	NULL	
longitud	varchar(15)	Sí	NULL	
pulso	varchar(10)	Sí	NULL	
oxigenación	varchar(10)	Sí	NULL	
temperatura	varchar(10)	Sí	NULL	
fecha	date	No		
hora	time	No		

Tabla 3.9 registro

Nombre de la clave	Tipo	Único	Empaquetado	Columna	Cardinalidad	Cotejamiento	Nulo	Comentario
PRIMARY	BTREE	Sí	No	id_registro	16	A	No	
id_dispositivo	BTREE	No	No	id_dispositivo	2	A	No	

Tabla 3.10 Índices tabla registros

3.2.3. Diseño eléctrico y electrónico del dispositivo

El esquema eléctrico es de gran ayuda para identificar averías de las instalaciones eléctricas que se encuentran instaladas en los dispositivos de montaje, sin embargo, la realización de estos diagramas ofrece la ventaja de representar circuitos o sistemas eléctricos y de esta forma montarlos de tal forma que sea fácil identificar los componentes por cualquier persona y de esta forma realizar la instalación o el montaje de los dispositivos.

Para el desarrollo del proyecto fue necesario la utilización de diferentes tipos de sensores, dispositivos electrónicos los cuales en conjunto con una codificación adecuada conforman un sistema de comunicación embebido que permite la interpretación de señales eléctricas a lenguaje humano por medio del envío y recepción de datos.

Para el desarrollo del prototipo como primera etapa se utiliza Arduino que es una plataforma de desarrollo basada en una placa electrónica de hardware libre, el modelo

utilizado en el presente prototipo es Arduino nano que contiene un microcontrolador ATmega328P de bajo costo y versatilidad [3] véase la Figura 3.9

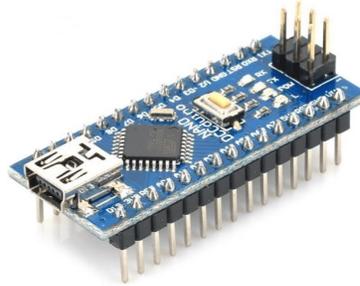


Figura 3.9 Placa Arduino Nano

Con la finalidad de controlar la ubicación del dispositivo se utiliza el módulo GPS6MV2 este dispositivo electrónico utiliza el sistema de posicionamiento global (GPS) para determinar la localización exacta de un vehículo, persona, u otro activo este módulo está basado en el receptor de la marca Ublox modelo NEO 6M, puede funcionar con un voltaje de alimentación en el rango de 3.0 a 5.0 volts véase la Figura 3.10

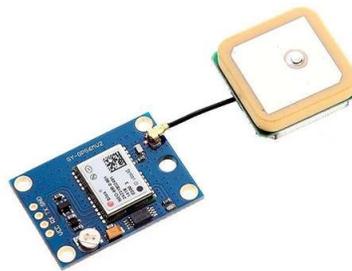


Figura 3.10 Modulo GPS6MV2

La temperatura corporal es resultado del equilibrio entre la producción y la pérdida de calor del cuerpo, para mantener el control de este parámetro y adquirir una lectura utilizaremos la medición de temperatura de la piel, mediante el sensor MLX90614 el cual tiene la función de un termómetro infrarrojo y su principal característica es la medición de temperatura sin contacto como se muestra en la Figura 3.11



Figura 3.11 Sensor de temperatura infrarrojo MLX90614

Otro parámetro que tomaremos en cuenta es la frecuencia cardíaca a la cual se identifica como el número de contracciones cardíaca por unidad de tiempo, consiste en el conteo de la cantidad de contracciones cardíacas durante un minuto. El Módulo integrado MAX30102 es un detector que mide la frecuencia del pulso haciendo uso de dos Leds: un led rojo (660nm) y un led infrarrojo (920nm), un fotodetector, óptica especializada, filtro de luz ambiental entre 50 y 60Hz, y un conversor ADC delta sigma de 16 bits y de hasta 1000 muestras por segundo en las cuales se mide la cantidad de luz reflejada véase la Figura 3.12



Figura 3.12 Pulsioxímetro MAX30102

La comunicación para el envío de la información del dispositivo electrónico a un dispositivo móvil se realizará por medio de GPRS, es una tecnología probada y funcional que permite conectividad total, Acceder a Internet, Conectarte con cámaras digitales (capturar y enviar imágenes), Conectarte con reproductores de MP3, Acceder a contenidos informativos y servicios en línea vía WAP, en cualquier momento y lugar. El módulo Iot G A6 Gsm Gprs será utilizado para este prototipo y será el encargado de realizar el envío de los datos al dispositivo móvil tal como se muestra en Figura 3.13

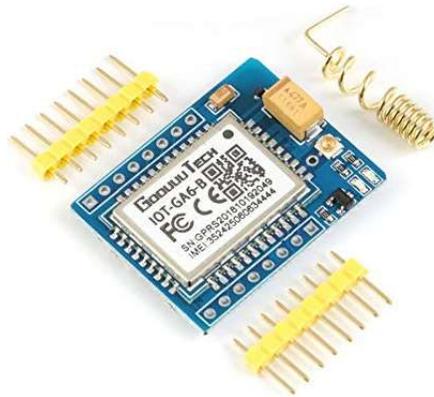


Figura 3.13 Modulo IoT GA6- B GSM/GPRS

Con la ayuda de una placa de pruebas protoboard se desarrolla la topología electrónica de los sensores y módulos que conforman el prototipo, donde la placa de Arduino nano es la encargada de realizar la sincronización a través de código de cada una de las funciones que realizarán los módulos tal como se presenta en la Figura 4.14

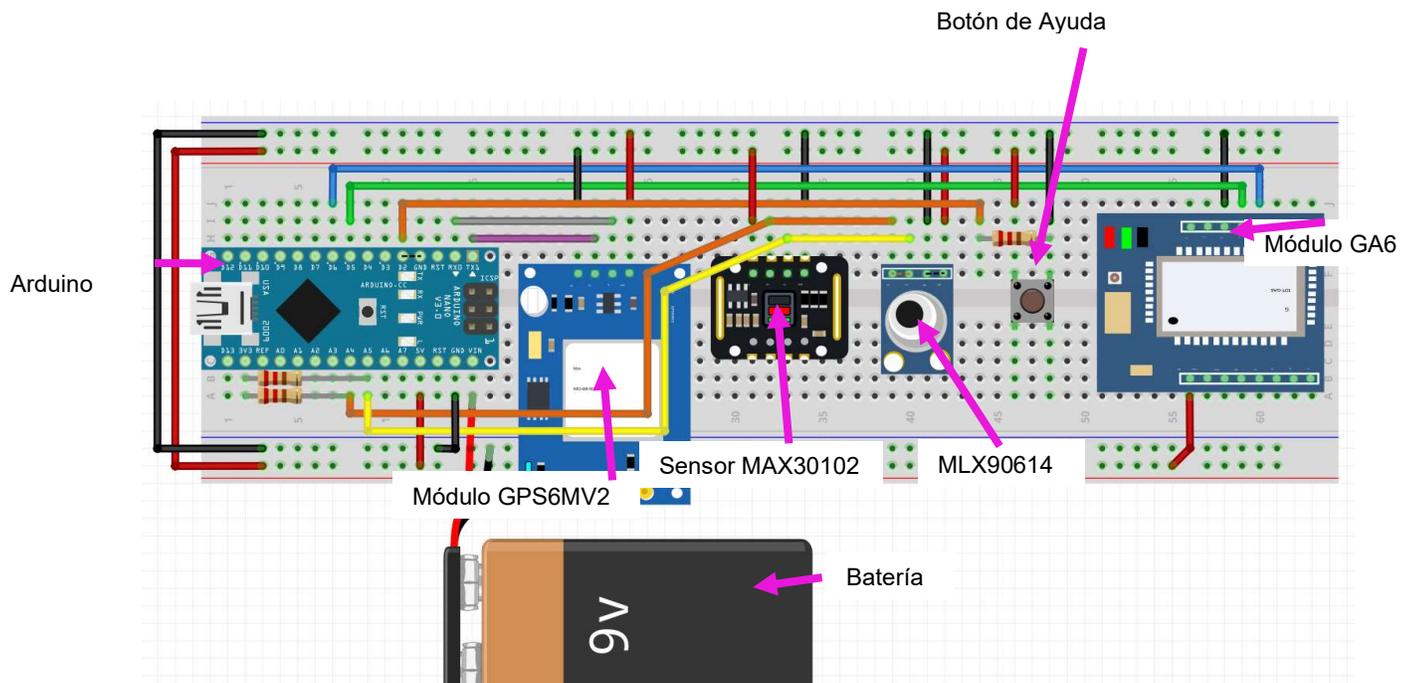


Figura 3.14 Diagrama electrónico para envío de ubicación, temperatura, y ritmo cardíaco a través de una red GPRS

Dentro del diseño principal utilizaremos los puertos analógicos, digitales, envío TX y recepción RX de la placa Arduino, como primera parte los puertos TX y RX de la placa van conectados directamente a los puertos TX y RX del Módulo GPS6MV2 tomando en cuenta una conexión cruzada donde el TX de la Placa Arduino se conecta con RX del Módulo y el puerto RX de Arduino se conecta al TX de módulo GPS el módulo cuenta con otros dos pines GND que va al negativo de la batería y VCC que tiene que ir conectado a 5 volts.

El segundo sensor que conectaremos el de pulsioxímetro MAX30102 este módulo cuenta con múltiples pines de los cuales solo utilizaremos el VCC que va conectado directamente a 5 volts, el pin GND que va comentado al negativo de la batería, posteriormente tenemos el puerto SCL que va conectado al pin Analógico 4 de la placa Arduino y el SDA conectado al puerto Analógico 5 de la placa Arduino. Para este sensor

se realiza una conexión de tipo I2C que es un protocolo síncrono. I2C usa solo 2 cables, esto puede ocasionar problemas con la comunicación a través del mismo medio con otros sensores por esta razón se utiliza resistencias de pull-up esto permitirá poner el bus a nivel alto, y los dispositivos envían niveles bajos. Si quieren enviar un nivel alto simplemente lo comunican al bus y será reconocido ya que cada sensor tiene una dirección única que permitirá trabajar las señales de múltiples sensores por medio de dos cables.

El tercer sensor que también utiliza una comunicación I2C es el sensor infrarrojo de temperatura MLX90614 se conecta de la siguiente manera pin VCC a 5 volts, pin GND al negativo de la batería, pin SCL al pin Analógico 4 de Arduino y SDA al pin analógico 5 de la placa Arduino como puedes ver este sensor comparte los puertos de comunicación con otro, esto es posible gracias al protocolo I2C.

Como cuarto paso conectaremos el módulo que permitirá el envío de las lecturas de los sensores procesadas por el microcontrolador en forma de mensaje de texto a un dispositivo móvil, estamos hablando del módulo GA6 -B GSM/GPRS este hace uso de un chip para el acceso a la red móvil y está conectado a la placa Arduino por medio de una conexión serial que utiliza el puerto URX y UTX del módulo conectado a los puertos digitales 5 y 6 de la placa Arduino, un puerto GND conectado al negativo de la batería y un puerto VCC que se conecta directamente a 5 volts.

Por último, pero no menos importante conexión del botón de ayuda que va de un extremo al GND de la batería seguido de una resistencia la cual conecta al otro extremo al botón y directamente al pin digital 2 de la placa Arduino. La conexión de estos módulos y sensores crean un dispositivo electrónico capaz de realizar lecturas de ubicación, temperatura, frecuencia cardiaca, procesar estas señales con el microcontrolador de la placa Arduino y enviarlas por la red móvil en forma de mensaje de texto a un teléfono celular o smartphone de forma automática cuando se detectan variaciones fuera de lo común en las lecturas de signos vitales o en su defecto cuando el portador del dispositivo oprime el botón de ayuda tal como se indica en la Figura 3.15

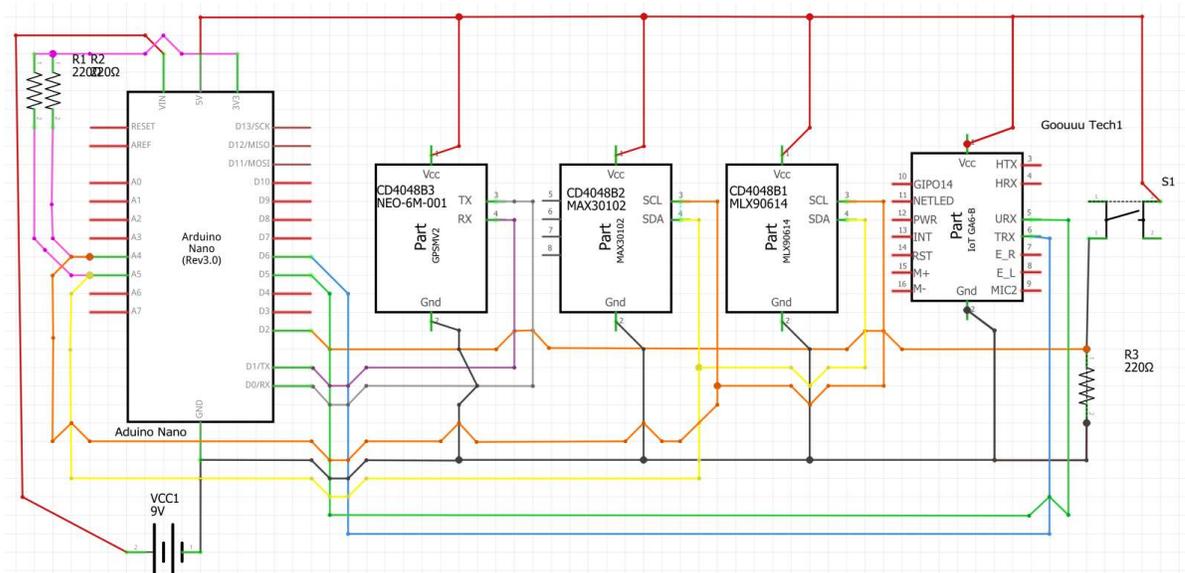


Figura 3.15 Diagrama eléctrico del prototipo

3.2.4. Modelado 3D

El dispositivo electrónico para envío de ubicación, temperatura y ritmo cardiaco a un servidor de internet está formado por una estructura ver Figura 3.16 la cual permite ordenar y agrupar los sensores de temperatura (6), GPS (7), ritmo cardiaco (4), módulo GPRS (8) y microcontrolador (3) que conforman dicho invento además se crearon conductos (2) que transporta el cableado ubicados en el mismo componente y un espacio para incorporar un botón (5) que se encargará de enviar una alerta por mensaje de texto.

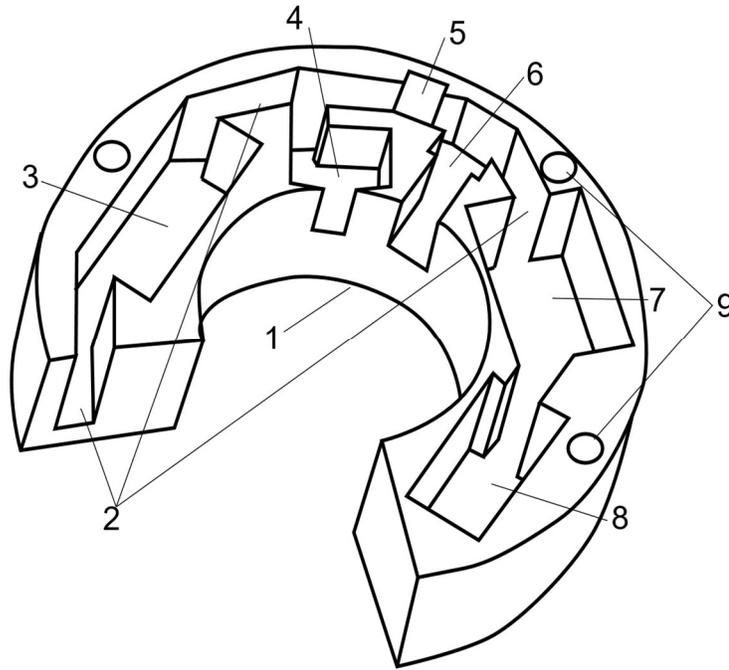


Figura 3.16 Base del brazalete que contiene los espacios para introducir los componentes

Una vez organizados los módulos, sensores y microcontrolador se procede a sellar el componente ensamblando la tapa ver Figura 3.17 que permitirá la protección de los componentes eléctricos y electrónicos esta cuenta con unas piezas en forma de postes (11) que se ensamblan directamente en las guías de la base del brazalete (9).

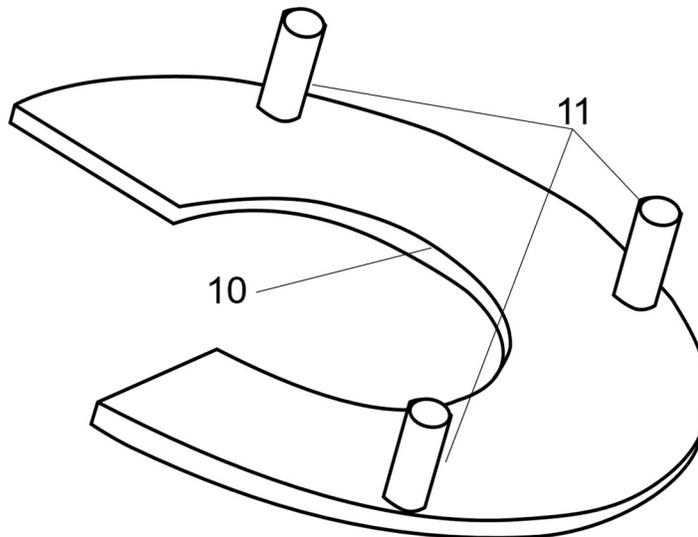


Figura 3.17 Tapa del brazalete

Para garantizar el funcionamiento del dispositivo es necesario conectar las baterías que alimentaran de corriente eléctrica directa, esto se hace por medio de una pieza diseñada para incorporar sus baterías ver Figura 3.18 con un canal (2) que permite conectar con la base de dispositivos, la entrada de almacenamiento de baterías (13) está diseñada para insertar dos baterías de 3.7 volts.

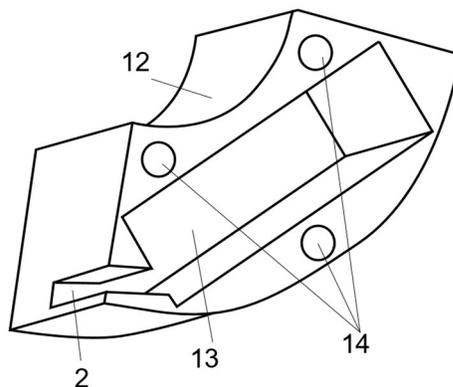


Figura 3.18 Base de baterías

La tapa del banco de baterías ver Figura 3.19 es parte del ensamble con el brazaletе garantizando la unión de ambas utilizando guías (14) y postes (15) para el ajuste al brazaletе se utilizan unas guías (17) con tornillos (18) y se coloca alrededor de la muñeca del portador (19).

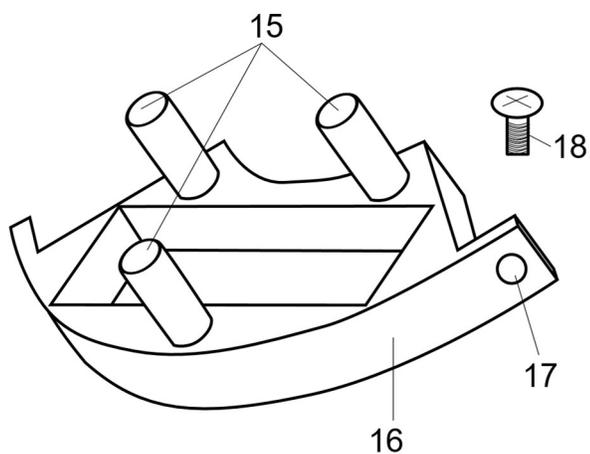


Figura 3.19 Tapa del banco de baterías

En la Figura 3.20 se puede visualizar claramente que el brazalete no tiene un diseño incorporado en la carcasa solo un botón (20) que permite enviar una señal de ayuda con la ubicación, ritmo cardíaco y temperatura del portador al dispositivo móvil vinculado.

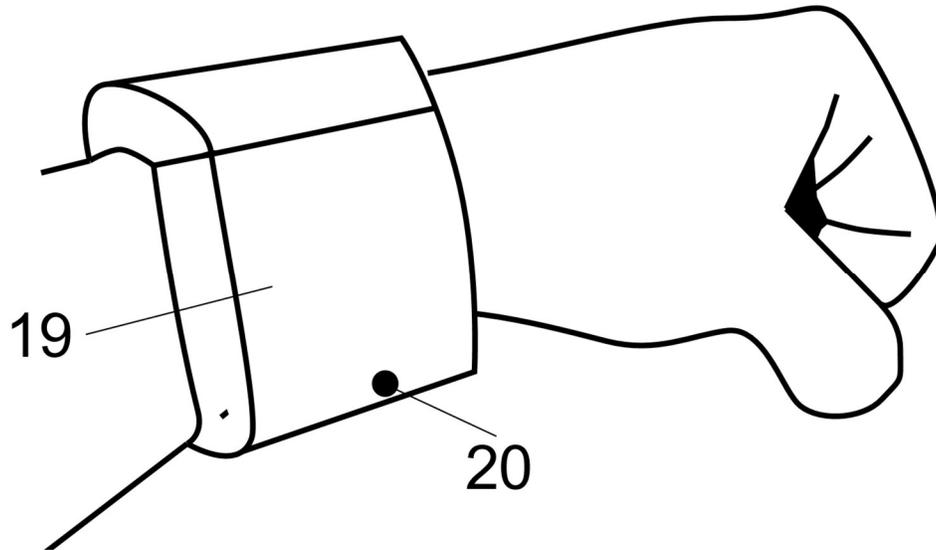


Figura 3.20 Brazalete portado en el brazo

3.2.5. Diseño de interfaces del software web para la administración de usuarios y dispositivos.

El software web de administración permitirá registrar a los usuarios que tienen la necesidad de monitorear a un familiar, este registro permitirá vincular a uno o varios dispositivos electrónicos que de igual forma se registran en esta plataforma de tal manera que solo los usuarios registrados y vinculados podrán acceder a las características de la app móvil. Una vez que formen parte de este sistema podrán gozar de los beneficios de la App Móvil. A continuación, presentamos y detallamos las interfaces del sistema web. La interfaz de Acceso permite a los Administradores ingresar a las características del sistema, gestionar usuarios, dispositivos y vincular estos registros tal como se muestra en la Figura 3.21



Figura 3.21 Interfaz de acceso al sistema solo para administradores

EL registro de los usuarios es muy importante ya que permitirá acceder a la app móvil para gestionar dispositivos electrónicos, se valida su información tomando en cuenta datos personales como CURP, Nombre, Dirección, Teléfono al cual llegaran las notificaciones, Correo Electrónico y Contraseña que servirá para acceder desde su dispositivo móvil a la app que se encargará de la gestión de los datos de los dispositivos véase en la Figura 3.22



Figura 3.22 Interfaz del menú principal registro de usuarios

Posterior al registro de usuarios, se registrarán los dispositivos estos tienen ciertas características que permitirán la gestión de los datos a los usuarios, para vincular ambos registros se tiene que recolectar información importante del dispositivo como IMEI, Número de teléfono desde donde navegara en internet y datos personales de la persona a la cual se va a monitorear tal como se indica en la Figura 3.23



Figura 3.23 Interfaz de menú principal registro de dispositivos

Una vez que los usuarios y los dispositivos se encuentran registrados en la base de datos el administrador del sistema podrá buscar ambos registros en el formulario de vinculación. Con la ayuda sensitiva del teclado al escribir las primeras letras del CURP dará una lista de usuarios registrados para seleccionar el que corresponde al cliente y en la parte de dispositivo se genera la misma función, pero con el IMEI de un dispositivo que no se encuentre vinculado, si ambos registros son validados y vinculados esta acción garantiza que el usuario con CURP X podría monitorear el dispositivo con IMEI X relacionando la información por medio de una app móvil véase Figura 3.24



Figura 3.24 Interfaz de vinculación de usuarios con dispositivos, un usuario puede vincularse a varios dispositivos, pero un dispositivo solo se vincula a un usuario

3.2.6. Diseño de interfaces de la App Móvil.

Las aplicaciones móviles permiten acceder a datos en la nube de forma rápida y accesible, la implementación de una app para el monitoreo del dispositivo permite brindar información clara y oportuna, haciendo uso de las notificaciones móviles que ofrecen como garantía una comunicación informativa en caso de la variación de resultados o parámetros fuera de rango de los pacientes.

El diseño de las interfaces de la App Móvil es de gran ayuda para entender la interpretación de los datos, esta app solo permite realizar consultas de los datos que almacenan los dispositivos vinculados al usuario.

La aplicación móvil contiene una interfaz de acceso en la cual se captura un usuario y contraseña previamente registrado por el administrador. Ver Figura 25 Una vez que accede a la App Móvil se mostrará la lista de dispositivos que están vinculados al usuario para monitoreo. Ver Figura 3.25 Al seleccionar un dispositivo de la lista se abrirá una ventana donde se muestra el último registro de los signos vitales en la base de datos, en esta ventana tenemos dos opciones una para generar el historial semanal y otra para el monitoreo en tiempo real. Ver Figura 26 Las lecturas del monitor es cada 30 minutos en caso de una variación de los signos vitales fuera del rango normal la app recibirá una notificación y se mostrarán las lecturas en tiempo real.

El usuario tendrá la posibilidad de generar un archivo y exportarlo para respaldar la información del monitoreo ver Figura 3.26.

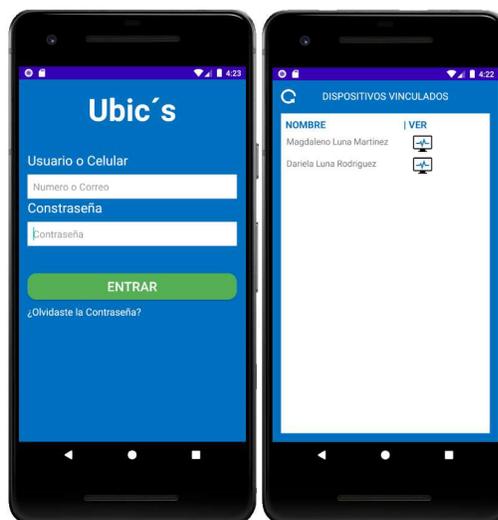


Figura 3.25 Interfaz de acceso e interfaz de dispositivos vinculados al dispositivo móvil

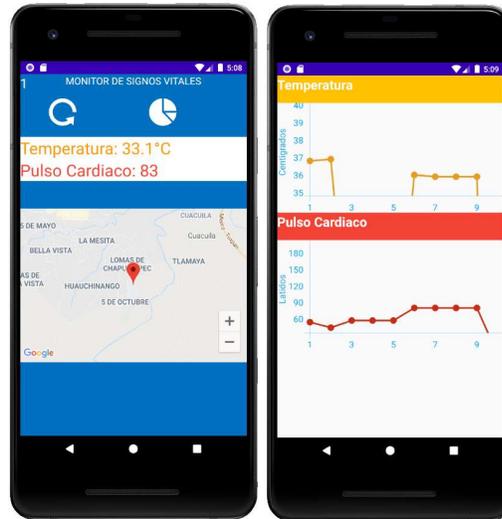


Figura 3.26 Interfaz de monitoreo del dispositivo e interfaz de historial semanal de las lecturas del UbiC's

3.3. Desarrollo del Proyecto

3.3.1. Programación del microcontrolador y sus componentes

Para la Lic. María Tenutto en la obra Escuela para Educadores año 2004, ‘programar es un proceso de toma de decisiones. Es establecer un conjunto de actividades en un contexto y tiempo determinado para enseñar los contenidos seleccionados en función de los objetivos establecidos. Es un proceso continuo, dinámico, no acabado ni rígido’ Pág. 705

En el mundo actual la automatización de procesos demanda de la programación de funciones, estas líneas de código permiten la comunicación entre el microcontrolador, módulos o sensores, con la finalidad de interpretar las lecturas analógicas o digitales, tomar decisiones, procesar la información, mostrarla o enviarla en un formato entendible para el ser humano, realizar conversiones, operaciones, análisis y tratamiento de los datos con la finalidad de establecer un intercambio de información, esto es posible gracias a la codificación.

Para el desarrollo de este proyecto utilizaremos cuatro funciones principales que corresponden a cada módulo, serán programadas en la placa de desarrollo Arduino y se describe su funcionamiento a continuación.

- La placa Arduino Nano es el motor principal del funcionamiento de nuestro dispositivo electrónico, es el encargado de realizar el procesamiento, inicializar las funciones,

realizar la toma de decisiones y comunicar a los módulos o sensores con el procesador central para preparar la información y enviarla a su destino.

Para iniciar la codificación en la placa se deben importar las librerías que permite utilizar y gestionar todas las características de los módulos como el de comunicación GSM/GPRS, gestión a la red GPS, monitoreo de señales del sensor de temperatura y pulso cardiaco tal como se muestra en la Figura 3.27

```
1 #include <SoftwareSerial.h>//incluimos la libreria SoftwareSerial
2 #include <TinyGPS.h>//incluimos la libreria TinyGPS para el modulo GPSPV2
3 #include <Wire.h> //Libreria que permite comunicarse con dispositivos por bus I2C
4 #include <MAX30105.h> //Libreria que permite obtener las lecturas de lu reflejada para calcular
5 #include <spo2_algorithm.h>
6 #include <heartRate.h>
7 #include <Adafruit_MLX90614.h>//Libreria que permite utilizar el sensor optico de temperatura
8 Adafruit_MLX90614 mlx = Adafruit_MLX90614();
9 TinyGPS gps;//Declaramos el objeto gps
10 SoftwareSerial serialmovil(6, 5);//Declaramos el pin 6 Rx y 5 Tx
11 MAX30105 particleSensor; |
12 int x=0;
13 float latitud, longitud, temp;
14 const byte RATE_SIZE = 4; //numero que se utiliza para obtener promedio de lectura
15 byte rates[RATE_SIZE]; //Matriz donde se almacenan las frecuencias cardiacas
16 byte rateSpot = 0;
17 long lastBeat = 0; //Hora de deteccion del ultimo latido
18 float beatsPerMinute;
19 int beatAvg, val;
20 const int boton=2;
21 int envio=0;
```

Figura 3.27 Importar librerías y declarar variables del sistema

- El setup es la primera función en ejecutarse dentro de un programa en Arduino. Es, básicamente, donde se «setean» las funciones que llevará a cabo el microcontrolador. Aquí es donde establecemos algunos criterios que requieren una ejecución única, en la función setup declaramos el pin 2 digital de Arduino en modo entrada ya que será el responsable de captar el evento del botón de ayuda de esta forma saber si esta pulsado o no, posteriormente establecemos la comunicación con la interfaz serial del Arduino, la conexión serial del módulo GA6 -B GSM/GPRS véase Figura 3.28
- La función void loop es una función que se repite constantemente en forma de un ciclo infinito, dentro de esta función se manda a traer y ejecutar la función ubicación la cual está ligada a otras funciones que permiten obtener lecturas de nuestro modulo y sensores.

```

22 void setup()
23 {
24   pinMode(boton, INPUT);
25   Wire.begin();
26   Serial.begin(9600); //Iniciamos el puerto serie
27   serialmovil.begin(9600); //Iniciamos el puerto serie del modulo GA6 B GSM/GPRS
28   while(!Serial);
29 }
30 void loop()
31 {
32   ubicacion();
33 }

```

Figura 3.28 Programación de la función setup y la función loop

- La función correspondiente al módulo GPSPV2 utiliza la librería TinyGPS.h que es fácil de usar para extraer la posición, fecha, hora, altitud, velocidad y rumbo de los dispositivos, se encuentra programada tomando una serie de pasos, en la primera parte nombramos a la función como ubicación (). Posteriormente verificamos que la conexión serial esta activa o lista para la comunicación, en caso de que si este disponible realizamos una lectura la cual almacenaremos en la variable c y con la ayuda de la librería para el módulo GPS obtenemos la lectura de latitud y longitud la almacenamos en las variables y damos un tiempo de 500 milisegundos, ejecutamos la función temperatura, posteriormente pasando otros 500 milisegundos, ejecutamos la función frecuencia e inicializamos la variable x en 0 esperamos 500 milisegundos y ejecutamos la función mensaje para el envío de los datos por la red GSM/GPRS tal como se indica en la Figura 3.29

```

98 void ubicacion(){
99   if (Serial.available())
100   {
101     int c = Serial.read();
102     if(gps.encode(c))
103     {
104       gps.f_get_position(&latitud, &longitud);
105       Serial.print("La/Lo: ");
106       Serial.print(latitud,5);
107       Serial.print(", ");
108       Serial.println(longitud,5);
109       delay(500);
110       temperatura();
111       delay(500);
112       frecuencia();
113       x=0;
114       delay(500);
115       mensaje();
116       delay(3000);
117     }

```

Figura 3.29 Código de la función ubicación

- Para codificar la estructura que calcula la temperatura con el módulo MLX90614 es necesario crear una función llamada temperatura la cual utiliza la librería Wire.h para establecer la comunicación I2C y Adafruit_MLX90614.h para obtener las lecturas se utiliza una variable de tipo flotante llamada temp y se inicializa en 0, iniciamos la conexión del dispositivo con mlx.begin() la cual se encarga de mostrar la comunicación de las lecturas eléctricas del dispositivo y con el comando readObjectTempC obtenemos las lecturas en grados centígrados, posteriormente este valor se asigna a la variable temp y está lista para ser mostrada al usuario véase Figura 3.30

```
91 void temperatura() {  
92     temp=0;  
93     mlx.begin();  
94     delay(500);  
95     temp=mlx.readObjectTempC();  
96     delay(500);  
97 }
```

Figura 3.30 Código de la función temperatura

- Para configurar el sensor de pulso cardiaco MAX30102 se utiliza la librería MAX3010x de SparkFun que permite realizar el cálculo de la luz reflejada y analizar los parámetros para calcular el ritmo cardiaco, este sensor igual utiliza la comunicación I2C para la lectura y recepción de señales eléctricas.

Iniciamos la codificación de este sensor verificando la conectividad del puerto I2C, si existe una lectura diferente a la señal del sensor se manda el mensaje que indica cuando no está fusionando el sensor de lo contrario si se detecta una señal se pide al usuario que coloque su dedo para iniciar la lectura, se agrega una condición para que el botón de ayuda sea capaz de enviar los datos de ubicación y temperatura aun cuando no funcione el sensor de ritmo cardiaco ver Figura 3.31.

```

34 void frecuencia(){
35
36     if (!particleSensor.begin(Wire, I2C_SPEED_FAST)) //Utilizar el puerto I2C, velocidad 400kHz
37     {
38         Serial.println("MAX3010 No funciona el Sensor ");
39         while (1);
40     }
41     Serial.println("Coloca tu dedo en el sensor.");
42     particleSensor.setup(); //Configurar el sensor con parametros predeterminados
43     particleSensor.setPulseAmplitudeRed(0x0A); //Encender el led rojo para identificar su funcion
44     particleSensor.setPulseAmplitudeGreen(0); //Apagar el led verde
45     while( x<1500 ){
46         x=x+1;
47         long irValue = particleSensor.getIR();
48
49         if (checkForBeat(irValue) == true)
50         {
51             //Cuando se detecta un latido
52             long delta = millis() - lastBeat;
53             lastBeat = millis();
54
55             beatsPerMinute = 60 / (delta / 1000.0);

```

Figura 3.31 Codificación del sensor de ritmo cardiaco

Continuando con la codificación del módulo de ritmo cardiaco, generamos un ciclo de 1500 repeticiones en las cuales tomaremos lecturas de la luz led IR, con la función obtenemos una lectura y la almacenamos en una variable de tipo long llamada irValue, verificamos si la lectura está arriba de 50000 de esta forma garantizamos que se está realizando la lectura de la luz reflejada de lo contrario enviara un mensaje recomendando colocar el dedo o la muñeca, una vez detectada la señal asignando un valor verdadero y posteriormente se calcula el tiempo en milisegundos de una lectura a otra, este parámetro se almacena en la variable delta, posteriormente se toman la variable delta y se divide entre 1000.0 que son las muestras que se realizan por segundo, el resultado de este valor se divide entre 60 que es el tiempo de lectura y da como resultado las pulsaciones por minuto.

Una vez calculado el parámetro BPM se realiza una comparación para saber si el valor es menor que 255 y mayor que 20 si se cumple la condición almacenar el valor en un arreglo y posteriormente calcula el promedio de la lectura, una vez calculados los valores se imprimen en terminal serial cuanto se ejecuta cada lectura se activa una función correspondiente al botón de ayuda el cual al ser pulsado detectara la pulsación, tomara la temperatura y enviara los parámetros pausando el proceso de la toma de ritmo cardiaco dando prioridad al envío por del mensaje tal como se muestra en la Figura 3.32

```

57     if (beatsPerMinute < 255 && beatsPerMinute > 20)
58     {
59         rates[ratesSpot++] = (byte)beatsPerMinute; //Almacenar esta lectura en un arreglo
60         ratesSpot %= RATE_SIZE; //Ajustar el valor de la variable
61
62         //Tomar el promedio de las lecturas
63         beatAvg = 0;
64         for (byte x = 0 ; x < RATE_SIZE ; x++)
65             beatAvg += rates[x];
66         beatAvg /= RATE_SIZE;
67     }
68 }
69
70 Serial.print("IR=");
71 Serial.print(irValue);
72 Serial.print(", BPM=");
73 Serial.print(beatsPerMinute);
74 Serial.print(", Avg BPM=");
75 Serial.print(beatAvg);
76 val=digitalRead(boton); // detectar si el usuario pulsa o no el botón de ayuda
77 if(val==HIGH){
78     envio=1;
79     temperatura();
80     mensaje();
81     delay(500);
82 }
83 else {
84     envio=0;
85 }
86 if (irValue < 50000) //si la lectura de ir es menor a 50000 significa que no se ha colocado
87     Serial.print(" No hay lectura?");
88     Serial.println();
89 }
90 }

```

Figura 3.32 Código para calcular el ritmo cardíaco

- Una vez que se han desarrollado las codificaciones de todos los módulos y sensores pasamos al envío de los mensajes de texto por medio del módulo GA6 -B GSM/GPRS, este envío depende directamente de la interfaz serial llamada serial móvil, ejecutamos los comandos AT+CMGF=1 para configurar un mensaje de texto, para saber el destino del mensaje de texto utilizamos el comando AT+CMGS=\”teléfono al que se envía la información\””, posteriormente comparamos la temperatura si un parámetro (temperatura, ritmo cardíaco) este fuera de los rangos normales el mensaje es enviado automáticamente de lo contrario podemos enviar los cuándo se oprima el botón de ayuda en el dispositivo electrónico ver Figura 3.33

```

120 void mensaje(){
121     serialmovil.println("AT+CMGF=1"); // Configurando modo TEXTO
122     serialmovil.println("AT+CMGS=\"+527641114713\"");//cambiar codigo de pais +52 colocar
123     if(temp>37.2 or beatAvg>20 and beatAvg<50 or beatAvg>120){
124         serialmovil.print("Temperatura o Frecuencia Fuera de Rango\n"); //Contenido del
125         serialmovil.println("Ubicacion: https://maps.google.com/?q=");
126         serialmovil.print(latitud,5);
127         serialmovil.print(", ");
128         serialmovil.print(longitud,5);
129         serialmovil.println("\nLatidos: ");
130         serialmovil.print(beatAvg);
131         serialmovil.println("\nTemperatura: ");
132         serialmovil.print(temp);
133         serialmovil.write((char)26);
134         delay(2000);
135         Serial.println("Signos Vitales Fuera de Rango");
136         Serial.println("\nUbicacion: ");
137         Serial.print(latitud,5);
138         Serial.print(", ");
139         Serial.print(longitud,5);
140         Serial.println("\nLatidos: ");
141         Serial.print(beatAvg);
142         Serial.println("\nTemperatura: ");
143         Serial.print(temp);
144         serialmovil.write((char)26);
145         delay(2000);
146     }else if(envio==1){
147         serialmovil.println("Temperatura y Frecuencia Cardiaca Normal\n"); //Contenido
148         serialmovil.println("Ubicacion: https://maps.google.com/?q=");
149         serialmovil.print(latitud,5);
150         serialmovil.print(", ");
151         serialmovil.print(longitud,5);
152         serialmovil.println("\nLatidos: ");
153         serialmovil.print(beatAvg);
154         serialmovil.println("\nTemperatura: ");
155         serialmovil.print(temp);
156         serialmovil.write((char)26);
157         delay(2000);
158         Serial.println("Signos Vitales Normales");
159         Serial.println("\nUbicacion: ");
160         Serial.print(latitud,5);
161         Serial.print(", ");
162         Serial.print(longitud,5);
163         Serial.println("\nLatidos: ");
164         Serial.print(beatAvg);
165         Serial.println("\nTemperatura: ");
166         Serial.println(temp);
167     }else{
168         Serial.println("Ubicacion: ");
169         Serial.print(latitud,5);
170         Serial.print(", ");
171         Serial.print(longitud,5);
172         Serial.println("\nLatidos: ");
173         Serial.print(beatAvg);
174         Serial.println("\nTemperatura: ");
175         Serial.println(temp);
176     }

```

Figura 3.33 Código para el envío del mensaje de texto de forma automática o manual

3.3.2. Programación del sistema de administración web

Una vez realizado el diseño de las interfaces, se realiza la etapa de la codificación, se buscó que funcionará de la mejor manera utilizando el lenguaje PHP, hojas de estilo CSS y JavaScript, en cada una de las figuras que se muestra a continuación se explica de forma general como se desarrollaron.

En la Figura 3.34 se muestra cómo se realizó el apartado del login, en este archivo se genera una condición if para validar los campos del usuario y la contraseña a través del método post, si los datos enviados están vacíos mostrará un mensaje de error, en caso contrario se conecta a la base de datos y valida si el usuario y contraseña son correctos, una vez validado los datos se obtendrá un resultado el cual se guarda en una variable result con otra condición se verifica si result es mayor a 0 entonces guardara toda la información en un array, después se crean una sesión con los datos obtenidos y nos direcciona al menú principal de la aplicación.

```
if(!empty($_POST))
{
    if(empty($_POST['usuario']) || empty($_POST['clave']))
    {
        $alert='<p class="msg_error">Ingresa tu usuario y contraseña</p>';
    }else{

        require_once "conexion.php";

        $user = $_POST['usuario'];
        $pass = $_POST['clave'];

        $query = mysqli_query($conexion, "SELECT * FROM cliente WHERE correo = '$user' AND
contrasena = '$pass'");
        $result = mysqli_num_rows($query);

        if($result > 0)
        {
            $data = mysqli_fetch_array($query);

            $_SESSION['active'] = true;
            $_SESSION['id'] = $data['id_cliente'];
            $_SESSION['curp'] = $data['curp'];
            $_SESSION['nombre'] = $data['nombre'];
            $_SESSION['direccion'] = $data['direccion'];
            $_SESSION['telefono'] = $data['telefono'];
            $_SESSION['user'] = $data['correo'];
            $_SESSION['tipo'] = $data['tipo'];

            header('location:asistente.php');
        }else{
            $alert='<p class="msg_error">Usuario y contraseña incorrectos</p>';
            session_destroy ();
        }
    }
}
```

Figura 3.34 Función de acceso al sistema

Al ingresar al sistema se muestra una interfaz en la que el administrador podrá registrar, consultar, actualizar asistentes, dispositivos y vincular estos datos para que sea posible que un asistente de tipo familiar, médico o amigo pueda monitorear los dispositivos que se vinculan en el sistema.

3.3.4. Insertar información

En la Figura 3.35 se muestra una parte de código el cual permite el almacenamiento de los datos de un asistente sin embargo la función es aplicada a los dispositivos y a la parte de la vinculación de los registros, lo primero que se realizó fue la validación con una condición if, donde los campos estén llenos, si está vacío algún campo retorna un mensaje de error a través de la variable alert, en caso contrario recibimos todos los datos que vienen del método post en diferentes variables, antes de guardar se realiza una validación del correo en la base de datos para evitar duplicados, después se guarda en un array, se almacena la información con un insert into, después se evalúa la variable para verificar si los datos se insertaron de forma correcta

```
if (!empty($_POST)) {  
    $alert='';  
    if (empty($_POST['curp']) || empty($_POST['nombre']) || empty($_POST['direccion']) || empty(  
$_POST['telefono']) || empty($_POST['correo']) || empty($_POST['clave']) || empty($_POST['tipo'])) {  
        $alert='<p class="msg_error"> Todos los campos son obligatorios</p>';  
    }else{  
        require_once "conexion.php";  
  
        $curp = $_POST['curp'];  
        $nombre = $_POST['nombre'];  
        $direccion = $_POST['direccion'];  
        $telefono = $_POST['telefono'];  
        $correo = $_POST['correo'];  
        $clave = $_POST['clave'];  
        $tipo = $_POST['tipo'];  
  
        $query = mysqli_query($conexion, "SELECT * FROM cliente WHERE correo = '$correo'");  
        $result =mysqli_fetch_array($query);  
  
        if ($result >0) {  
            $alert='<p class="msg_error">El correo ya existe</p>';  
        }else{  
            $query_insert = mysqli_query($conexion, "INSERT INTO cliente(curp,nombre,direccion,  
telefono,correo,contrasena,tipo) VALUES('$curp','$nombre','$direccion','$telefono','$correo','$clave'  
','$tipo')");  
  
            if ($query_insert) {  
                $alert='<p class="msg_save">Cliente creado correctamente</p>';  
            }else{  
                $alert='<p class="msg_error">Error al crear el cliente</p>';  
            }  
        }  
    }  
}
```

Figura 3.35 Insertar información en la base de datos

3.3.5. Consultar información

Como parte de la programación del sistema web tenemos el apartado de búsqueda de información, para generar la función de búsqueda dentro de nuestro sistema debemos tomar en cuenta los parámetros de que necesitamos encontrar en nuestra base de datos, para realizar este proceso en colocaremos la información en una caja de texto en la cual el usuario podrá introducir el valor de la búsqueda o dejar en blanco el campo, en caso de que el usuario decida dejar en blanco el campo se mostrarán todos los datos que se encuentran almacenados, en caso que el usuario introduzca información la búsqueda muestra los datos relacionados con los caracteres introducidos por el usuario tomando en cuenta todos los campos de la tabla que está ligada a la información del formulario ver Figura 3.36.

```
48 <table class="table-bordered border">
49 <tr>
50 <th>Id</th>
51 <th>Curp</th>
52 <th>Nombre</th>
53 <th>Direccion</th>
54 <th>Telefono</th>
55 <th>Correo</th>
56 <th>Tipo</th>
57 <th>Acciones</th>
58 </tr>
59 <?php
60
61 $sql_registe =mysql_query($connection, "SELECT count(*) as total_registro FROM cliente ");
62
63 $result_register =mysql_fetch_array($sql_registe);
64
65 $total_registro = $result_register['total_registro'];
66
67 $por_pagina = 5;
68
69 if (empty($_GET['pagina'])) {
70     $pagina = 1;
71 }
72 }else{
73     $pagina= $_GET['pagina'];
74 }
75
76
77 $desde =($pagina-1)* $por_pagina;
78 $total_paginas = ceil($total_registro / $por_pagina);
79
80 $query= mysql_query($connection, "SELECT c.id_cliente,c.curp, c.nombre, c.direccion, c.telefono,
81
82 $result = mysql_num_rows($query);
83
84 if ($result >0) {
85     while ($data= mysql_fetch_array($query)) {
86
87
88 ?>
89 <tr>
90 <td><?php echo $data["id_cliente"];?></td>
91 <td><?php echo $data["curp"];?></td>
92 <td><?php echo $data["nombre"];?></td>
93 <td><?php echo $data["direccion"];?></td>
94 <td><?php echo $data["telefono"];?></td>
95 <td><?php echo $data["correo"];?></td>
96 <td><?php echo $data["tipo"];?></td>
```

Figura 3.36 Código que muestra los resultados de búsqueda

Una vez realizada la búsqueda se mostrarán una lista de resultados encontrados en los cuales el usuario podría eliminar o modificar la información referente a cada registro, los datos se muestran en un paginador el cual contiene las columnas correspondientes a la información y en la parte final se agregan dos campos eliminar o actualizar ver Figura 3.37 para decidir que se acción ejecutara el administrador.

```

if ($result >0) {
    while ($data= mysqli_fetch_array($query)) {
?>
        <tr>
            <td><?php echo $data["id_cliente"];?></td>
            <td><?php echo $data["curp"];?></td>
            <td><?php echo $data["nombre"];?></td>
            <td><?php echo $data["direccion"];?></td>
            <td><?php echo $data["telefono"];?></td>
            <td><?php echo $data["correo"];?></td>
            <td><?php echo $data["tipo"];?></td>
            <td>
                <a class="link_edit" href="actualizarasistente.php?id=<?php echo $data["id_cliente"];?>">
                    |
                <a class="link_delete" href="eliminarasistente.php?id=<?php echo $data["id_cliente"];?>">
            </td>
        </tr>
    </tr>
?>
    }
}
?>
</table>
<div class="paginador">
    <ul>
        <?php
            if ($pagina !=1) {
?>
                <li><a href="?pagina=<?php echo 1;?>"><</a></li>
                <li><a href="?pagina=<?php echo $pagina - 1; ?>"><<</a></li>
            <?php
            }
        </ul>
    </div>

```

Figura 3.37 Paginador y columnas de eliminar y modificar

Las funciones básicas que realiza el sistema son registrar, buscar, modificar y eliminar este CRUD se repite en todos las pantallas del sistema presentadas en el formulario, recordemos que la finalidad del sistema es dar de alta dispositivos electrónicos que son los encargados de enviar las lecturas de ubicación, ritmo cardiaco y temperatura, registrar asistentes que administraran estos dispositivos desde la App y vincular dispositivos con asistentes

cuidadores para conceder permisos de monitoreo entre ellos y de esta forma garantizar la seguridad del sistema en el intercambio de información.

3.3.6. Programación de la App Móvil

La App móvil es de gran importancia para las personas que estarán a cargo del cuidado y monitoreo de la salud del paciente por esta razón se desarrollara en el lenguaje de programación java utilizando el IDE de android studio ver Figura 3.38 el cual garantiza la compatibilidad con dispositivos móviles de gama baja y alta existentes en el mercado a partir de android 5 hasta la versión más actual.

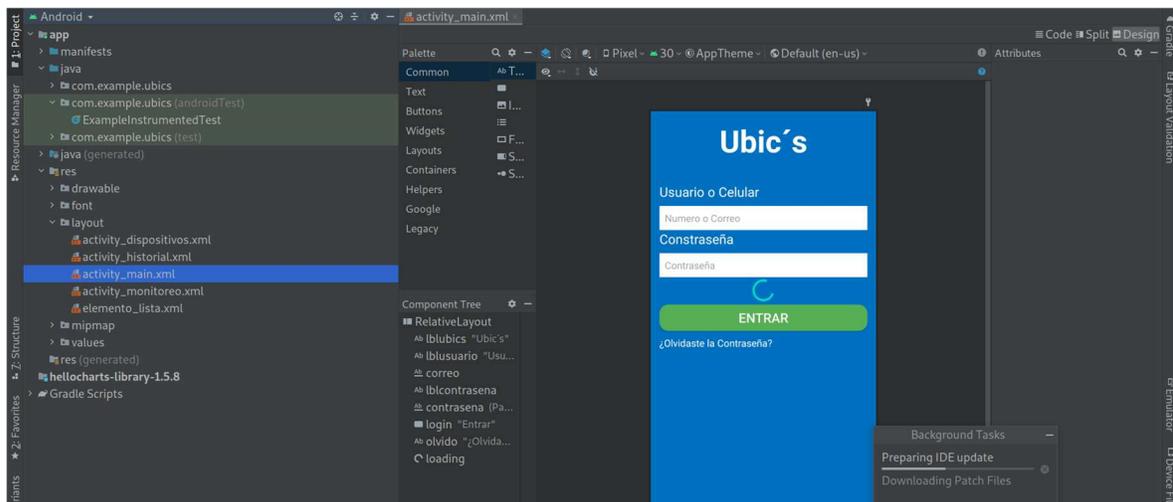


Figura 3.38 Programando pantalla de acceso

Para la creación de la app móvil se consideraron varios factores, el primero proporcionar seguridad a la aplicación mediante el uso de API KEYS desarrolladas en PHP, un lenguaje que trabaja del lado del servidor para recibir o enviar peticiones por medio de la red de internet utilizando servicios web. Los datos se enviarán por medio de un JSON que viajará hacia el servidor para proporcionar la información ver Figura 3.39 y esta sea validada se envíe una respuesta automática hacia la app que mostrará si el usuario es válido o invalido dando acceso a la pantalla principal de administración de la App.

```
41 private void loginAccesso(final String correo, final String contraseña){
42     cargando.setVisibility(View.VISIBLE);
43     final StringRequest stringRequest=new StringRequest(Request.Method.POST, URL_LOGIN, new Response.Listener<String>() {
44         @Override
45         public void onResponse(String response) {
46             try {
47                 JSONObject jsonObject = new JSONObject(response);
48                 boolean success= jsonObject.getBoolean("success");
49                 if(success){
50                     String id_cliente=jsonObject.getString("id_cliente");
51                     String nombre=jsonObject.getString("nombre");
52                     Intent intent=new Intent(getApplicationContext(), MainActivity.this, dispositivos.class);
53                     intent.putExtra("id_cliente", id_cliente);
54                     intent.putExtra("nombre", nombre);
55                     MainActivity.this.startActivity(intent);
56                     cargando.setVisibility(View.GONE);
57                     Limpiar();
58                 } else{
59                     Toast.makeText(getApplicationContext(), "Usuario o contraseña incorrectos", Toast.LENGTH_SHORT).show();
60                     Limpiar();
61                     cargando.setVisibility(View.GONE);
62                 }
63             } catch (JSONException e) {
64                 e.printStackTrace();
65                 cargando.setVisibility(View.GONE);
66                 Login.setVisibility(View.VISIBLE);
67                 Toast.makeText(getApplicationContext(), "Error 1: "+e.toString(), Toast.LENGTH_SHORT).show();
68             }
69         }, new Response.ErrorListener() {
70             @Override
71             public void onErrorResponse(VolleyError error) {
72                 cargando.setVisibility(View.GONE);
73                 Login.setVisibility(View.VISIBLE);
74                 Toast.makeText(getApplicationContext(), "Error de red verifica t
75             }
76         })
77     });
78 }
79 }
80 }
81 }
82 }
83 }
84 }
85 }
86 }
87 }
88 }
89 }
90 }
91 }
92 }
93 }
94 }
95 }
96 }
97 }
98 }
99 }
100 }
```

Figura 3.39 Código de validación de claves de acceso

Una vez ubicado en la pantalla de administración se mostrará una lista de dispositivos los cuales fueron vinculados al celular de la persona que se encargará del monitoreo, cada uno de estos ítems tiene un botón que permitirá acceder a los datos generados por el portador como ritmo cardíaco, temperatura y ubicación del último registro generado por el dispositivo y un historial en forma de gráfica que garantiza un análisis de las lecturas en tiempo real. Para obtener la lista de dispositivos se realiza una petición en formato JSON a un web service almacenado en el servidor el cual se encargará de mostrar todos los dispositivos vinculados a la cuenta del cuidador ver Figura 3.40.

```
public void verDispositivos(final String id_cliente){
    listadispositivos=new ArrayList<>();
    final String request=new StringRequest(Request.Method.POST, URL_CONSULTA, new Response.Listener<String>(){
        @Override
        public void onResponse(String response) {
            try{
                JSONObject jsonObject=new JSONObject(response);
                JSONArray jsonArray = jsonObject.getJSONArray(jsonObject.names());
                for (int i = 0; i < jsonArray.length(); i++) {
                    JSONObject jsonObject1 = jsonArray.getJSONObject(i);
                    listadispositivos.add(
                        new Dispositivo(
                            jsonObject1.getString("id_usuario"),
                            jsonObject1.getString("nombre")
                        )
                    );
                }
                adaptador = new AdaptadorDispositivos( context, this, listadispositivos);
                lista.setAdapter(adaptador);
            }catch (JSONException e){
                e.printStackTrace();
                Toast.makeText(context, "Error: "+e.toString(), Toast.LENGTH_SHORT).show();
            }
        }
    }, new Response.ErrorListener() {
        @Override
        public void onErrorResponse(VolleyError error) {
            Toast.makeText(context, "Error de registro"+error.toString(), Toast.LENGTH_SHORT).show();
            error.printStackTrace();
        }
    });
}

@Override
protected Map<String, String> getParams() throws AuthFailureError {
    Map<String, String> params=new HashMap<>();
    params.put("id_cliente", id_cliente);
    return params;
}

RequestQueue requestQueue= Volley.newRequestQueue(context);
requestQueue.add(stringRequest);
}
```

Figura 3.40 Código para cargar los dispositivos vinculados a la cuenta de asistente cuidador

Una vez mostrados los dispositivos vinculados se puede acceder a los datos generados de cada uno de ellos, y se muestra la última actualización o almacenamiento registrado en la base de datos, si necesitas conocer más detalles desde la aplicación puedes ir al historial de lecturas y te mostrará una gráfica de las ultimas lecturas registradas del portador en el sistema.

CAPITULO 4

RESULTADOS Y DISCUSIÓN

4.1. Resultados y Discusión

4.1.1. Desarrollo del prototipo

Para validar el correcto funcionamiento del prototipo se realizan las pruebas para evaluar el funcionamiento de cada uno de las secciones que lo componen. Como primera parte al iniciar el programa realizamos una conexión serial a la computadora para visualizar las lecturas de los sensores, ver Figura 4.1, el programa al iniciar nos proporciona los datos de latitud y longitud que permitirán obtener la ubicación del dispositivo, posteriormente se realizan 1500 lecturas de IR la cual tarda alrededor de un minuto, una vez finalizada se ejecuta la función de temperatura y se muestran los parámetros de localización, frecuencia cardiaca y temperatura.

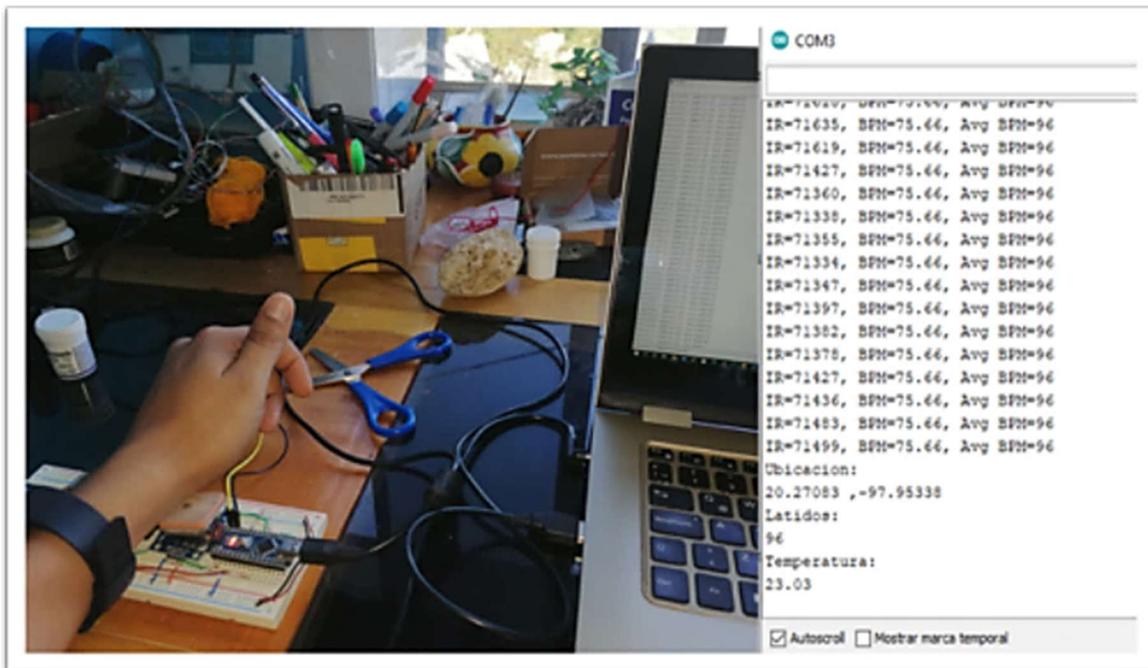


Figura 41 Lecturas del dispositivo

La segunda prueba se realiza al presionar el botón de ayuda en el dispositivo y se envían las lecturas al dispositivo móvil indicando si los parámetros están dentro del rango normal ver Figura 4.2 o fuera de rango.

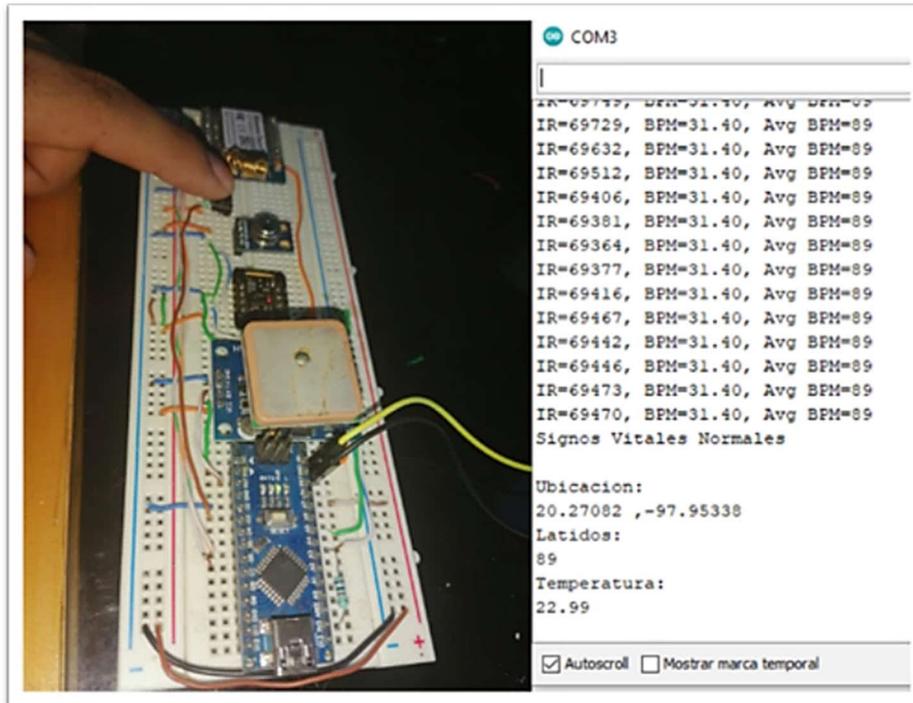


Figura 42 Lecturas en rango normal

La siguiente prueba realizada al dispositivo se utilizó un cautín y se acercó al sensor de temperatura, ver Figura 4.3, con la finalidad de sobrepasar el rango normal de los parámetros leídos y automáticamente al detectar esta variación se muestra y envía el mensaje de signos vitales fuera de rango al dispositivo móvil vinculado.

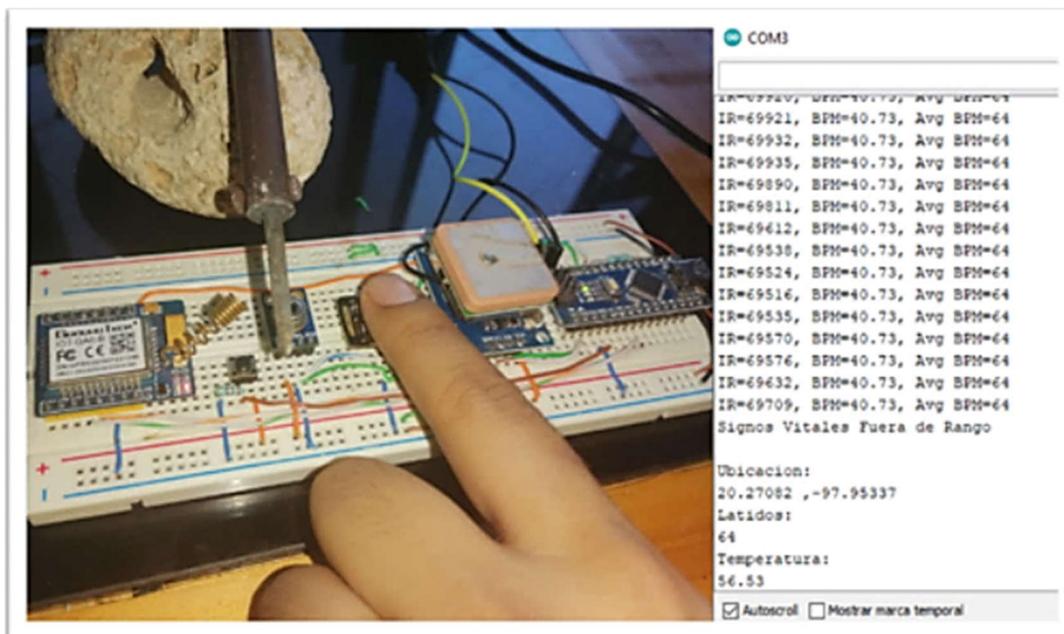


Figura 43 Lecturas fuera de rango

4.1.2. App Móvil.

Para el desarrollo de la aplicación móvil se utilizó Android estudio y programación orientada a objetos, la estructura de navegación de la app consta de diferentes vistas como la pantalla de acceso ver Figura 4.4 donde los usuarios que estén registrados en la app podrían entrar para realizar tareas de monitoreo de los dispositivos vinculados por el sistema web, seleccionar un dispositivo y visualizar las ultimas lecturas de ubicación, temperatura y ritmo cardiaco, ver Figura 4.5 además si quisiera conocer más a detalle las lecturas en el transcurso del día o la semana se incorpora un historial con gráficos que podríamos analizar por medio de representación de colores indicando con un color rojo el ritmo cardiaco, amarillo la temperatura.



Figura 44 Acceso y lista de dispositivos vinculados al cuidador

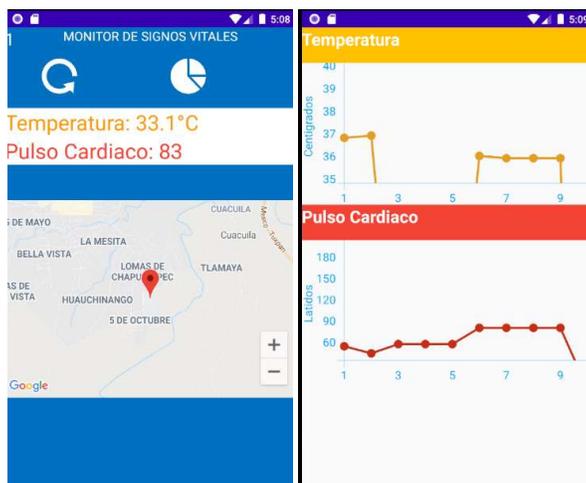


Figura 4.5 Interfaz de monitoreo e interfaz de historial semanal

Una vez finalizadas las pruebas se mostraron las notificaciones, ver Figura 4.6, en estas imágenes se muestra el mensaje de texto enviado por el dispositivo electrónico vinculado al número telefónico registrado en la función de mensaje del código.

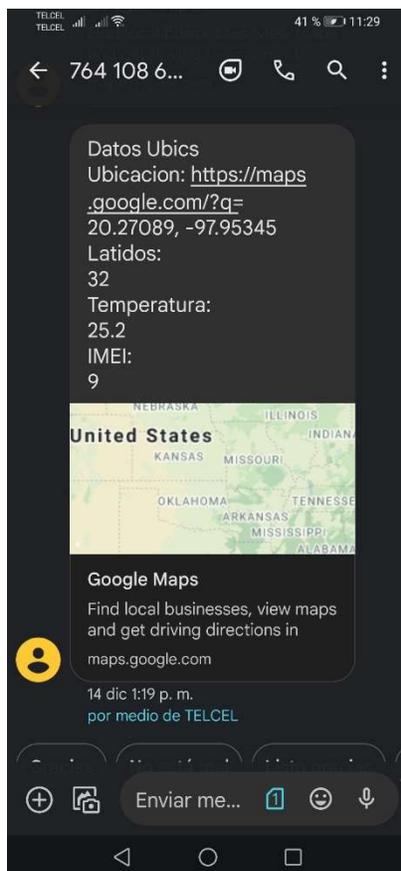


Figura 45.6 Notificaciones en mensaje de texto

4.1.3. Sistema de administración de usuarios, cuidadores y dispositivos.

Se realizó el registro del dispositivo en el sistema tomando en cuenta los datos de IMEI, numero, CURP, nombre del paciente y dirección, de igual forma se procedió a realizar el registro de los asistentes cuidadores en el que se agregaron datos como CURP, nombre, dirección, teléfono, correo, contraseña y tipo de registro. Una vez registrados los datos de la persona que porta el dispositivo y la que se encarga del monitoreo ver Figura 4.7, procedemos a realizar el registro de vinculación del registro del paciente con el cuidador y de esta forma se activa el monitoreo en la app.

Control de ubicacion y signos vitales								
Asistente		Dispositivos		Vincular		Salir		
Dispositivos Vinculados								
<input type="text" value="Buscar"/> <input type="button" value="Buscar"/>								
CUIDADOR	Curp	Nombre	DISPOSITIVO	Imei	Numero	Curp Del Portador	Status	Acciones
CUIDADOR	LUSR850819HPLNVC08	Ricardo Luna Santos	DISPOSITIVO	869892044250181	7641112200	LURD202807HPLNVC01	Activo	Editar Eliminar
CUIDADOR	ROFJ850716PLNDCN02	Juanita Rodriguez Francisco	DISPOSITIVO	2222222222222	7761124321	SAOCA65084LDASDU02	Activo	Editar Eliminar
CUIDADOR	LUSR850819HPLNVC08	Ricardo Luna Santos	DISPOSITIVO	11111111111111111	7657643551	LUMM65016LNVC08	Activo	Editar Eliminar

Figura 4.7 Lista de dispositivos y asistentes vinculados

Una vez vinculados los cuidadores con los dispositivos el usuario cuidador podrá instalar la app y entrar para poder monitorear los datos que se envían desde el dispositivo a la nube de internet en tiempo real ver Figura 4.8.

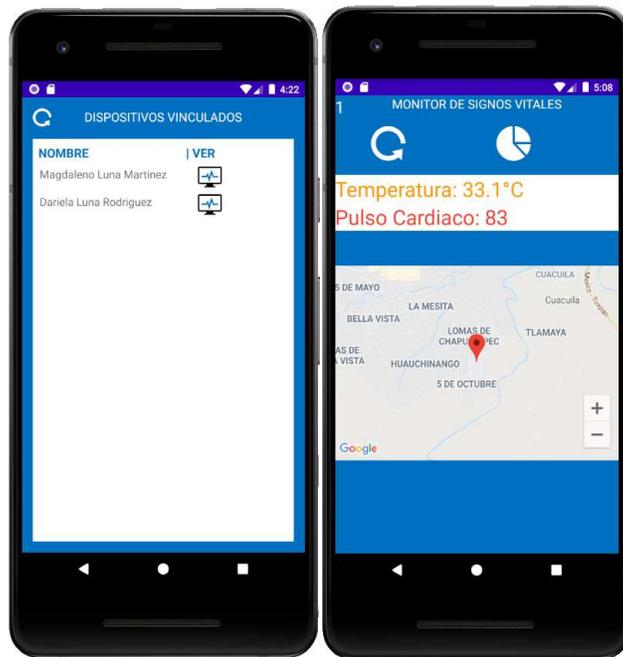


Figura 468 Monitoreo desde la App móvil

CAPITULO 5

CONCLUSIONES Y TRABAJOS FUTUROS

5.1. Conclusiones

En el presente trabajo se presentó el desarrollo de un dispositivo electrónico que permite medir la temperatura y ritmo cardiaco del portador, así como localizar su ubicación. El dispositivo constituye una herramienta útil para las personas encargadas de monitorear la ubicación y el estado de salud de personas que padecen alguna morbilidad crónica, o de adultos o infantes que necesiten ser vigilados para que a cualquier situación anormal puedan recibir atención inmediata y oportuna. Además, el dispositivo propuesto tiene la característica de permitir al portador, presionar un botón de ayuda el cual enviará una notificación en forma de texto que se visualizará en el teléfono celular que se encuentre vinculado. A las secciones que componen el dispositivo electrónico se le realizaron diversas pruebas para evaluar su correcto funcionamiento logrando resultados prometedores.

Además de presentar el dispositivo se desarrolla una App que se instala en un teléfono inteligente con sistema operativo Android y permite visualizar las estadísticas generadas por el dispositivo electrónico vinculados al celular, mostrando de forma gráfica la lectura de pulso cardiaco, temperatura y ubicación. Esta App permite visualizar de forma detallada un historial de lecturas en forma de gráficas y actualizadas cada determinado tiempo.

El módulo de administración del sistema se desarrolla con la finalidad de vincular los dispositivos del paciente con el celular de la persona encargada de su cuidado y de esta forma dar acceso a la información solo al personal autorizado.

5.2. Recomendaciones

El prototipo electrónico para envío de ubicación, temperatura y ritmo cardiaco está realizado con materiales económicos, se sugiere realizar una inversión superior para poder explotar todas sus funcionalidades y minimizar el consumo de energía en el dispositivo.

El sistema de administración web fue desarrollado en PHP se recomienda utilizar un framework para garantizar la estabilidad y aprovechar las características de la arquitectura que se ofrece en la implementación.

La app móvil está en una versión demo y trabaja solo en dispositivos Android studio se recomienda realizar la app en un lenguaje diferente para ejecutar en múltiples sistemas operativos y de esta forma no limitar su funcionamiento solo a unos dispositivos si no aprovechar la implementación en multiplataforma.

El prototipo está en una la primera etapa de desarrollo, pueden existir muchas mejoras en la programación y el envío de los datos, así como en la arquitectura empleada y la reingeniería en sus componentes y aplicaciones de software que interactúan en la implementación y el seguimiento del proyecto.

5.3. Trabajos Futuros

Como parte del proceso de desarrollo de cada proyecto de investigación, una o varias líneas tienen opciones de mejora en las cuales es posible seguir trabajando. Durante el desarrollo de la presente tesis surgen nuevas líneas de investigación las cuales se espera seguir retroalimentando con información nueva, aplicando constantes mejoras y retomarla para proyectos futuros.

A continuación, se presentan algunas áreas de oportunidad para realizar trabajos futuros que pueden mejorar el trabajo presentado en esta tesis y que, por exceder el alcance de tiempo, no han podido ser implementados y ejecutados en este proyecto. Además, se aportan algunas ideas específicas para apoyar y mejorar el prototipo. Entre los posibles trabajos futuros se destacan:

- Mejorar el diseño del prototipo, incluir materiales flexibles y ergonómicos que permitan un mejor ajuste a la muñeca, reducción del espacio y de esta forma garantizar que las lecturas sean más exactas.
- Analizar el uso de nuevos componentes y módulos utilizando equipos con mayor calidad. seguridad y reducción de espacio implementando nano electrónica, de esta forma aprovechar el tamaño del dispositivo para un mejor diseño y funcionamiento además de alargar la vida útil de la batería.
- Adaptar el prototipo a diferentes tipos de usuarios incorporando componentes necesarios para realizar una tarea en específico reduciendo el tamaño del prototipo y aprovechando al máximo su potencial en el control de cierto parámetros fisiológicos y la sincronización en la nube.
- Pensar en una idea de negocio para la venta del servicio de monitoreo hacia personas de diferentes edades, genero e incluso mascotas que forman parte de las familias en algunos hogares.
- Implementar machine learning para que el dispositivo logre aprender los valores normales de temperatura, pulso cardiaco de cada persona en un periodo de tiempo y de esta forma identificar un parámetro fuera del rango normal de un portador del dispositivo.

- Adaptar el dispositivo al tipo de conectividad más adecuado considerando que en algunos lugares no existe señal GPRS o red móvil, buscar alternativas como bluetooth, almacenamiento local en un dispositivo móvil y sincronización en la nube cuando exista conectividad.
- Implementar un módulo de cargador de batería para que en cualquier momento sea posible recargar las baterías y de esta forma aumentar el tiempo de uso.

ANEXO A

A.1. Publicación de un artículo en revista arbitrada e indexada



Monitor de temperatura, ritmo cardíaco y ubicación enviando mensajes de texto a un teléfono celular

Monitor temperature, heart rate and location by sending text messages to a cell phone

Ricardo Luna Santos

Tecnológico Nacional de México/Instituto Tecnológico Superior de Huauchinango, Huauchinango, México
ricardolunasan@gmail.com.mx
ORCID: 0001-8280-6769

Aldo Hernández Luna

Tecnológico Nacional de México/Instituto Tecnológico Superior de Huauchinango, Huauchinango, México
aldo.hdz@huauchinango.tecnm.mx
ORCID: 0000-0002-7717-5314

Jacinto Torres Jiménez

Tecnológico Nacional de México/Instituto Tecnológico Superior de Huauchinango, Huauchinango, México
jacinto.torres@huauchinango.tecnm.mx

Manuel Cruz Luna

Tecnológico Nacional de México/Instituto Tecnológico Superior de Huauchinango, Huauchinango, México
manuelcl@huauchinango.tecnm.mx
ORCID: 0000-0002-0640-8926

doi: <https://doi.org/10.36825/RITI.09.18.007>

Recibido: Enero 21, 2021

Aceptado: Mayo 25, 2021

Resumen: En el presente trabajo de investigación se propone el desarrollo de un dispositivo electrónico que permite realizar la lectura de temperatura, ritmo cardíaco y ubicación de una persona y enviar los datos en forma de mensaje de texto a un teléfono celular (*smartphone*) vinculado por medio de la red móvil. El dispositivo de monitoreo que se propone utiliza sensores que nos ayudan a detectar diferentes tipos de señales y que se traducen a lenguaje humano utilizando microcontroladores, electrónica y diversos tipos de programación para el desarrollo de funciones que permitan la obtención de los datos. Como resultado el dispositivo enviará un mensaje de texto solo si las lecturas devuelven un parámetro diferente a los rangos normales del ser humano o cuando el portador del dispositivo presione un botón de ayuda, de esa forma los mensajes de texto llegan al número telefónico del dispositivo móvil vinculado con la información del paciente o familiar.

Palabras clave: Monitoreo, Señales, Tiempo Real, Sensores, Red Móvil.

Referencias

- [1] Comparación de monitores de signos vitales del ECRI Institute. (2016, Agosto). Recuperado 5 de mayo de 2020, de <http://www.elhospital.com/temas/Monitores-de-signos-vitales-Parte-1+114421?pagina=1>
- [2] Balaguer, M. (2010). Revisando Técnicas: Control de signos vitales. *Revista de Enfermería*, 10, 5-10.
- [3] Penagos, S. P., Salazar, L. D., de Enfermería, E. C., Vera, F. E., & Cardioinfantil, F. (2005). Control de signos vitales. *Guías para manejo de Urgencias*. Bogotá (Colombia): Fundación Cardioinfantil, 1465-1473.
- [4] Gutiérrez Cisternas, C. A. (2016). Sistema de monitoreo continuo de signos vitales con sensores no invasivos y transmisión inalámbrica de datos (Doctoral dissertation, Universidad de Concepción. Facultad de Ingeniería. Departamento de Ingeniería Eléctrica).
- [5] Ballesteros, D. M., Melo, H. E., & Maya-Quintero, A. J. (2014). Sistema de transmisión inalámbrica de señales ECG y de temperatura para ambientes hospitalarios (SINHO).
- [6] Otero, J., & Gómez, A. (2007). Integración de dispositivos biomédicos en sistemas de teleasistencia. Galicia: Centro de Supercomputación de Galicia.
- [7] López, L. J. R., López, A. F. M., & Sanabria, Y. P. C. (2015). Aplicación de la biotelemedicina para tres signos vitales. *Ciencia y Poder Aéreo*, 10(1), 179-186.
- [8] Cruz, J., Villegas, G., Uyá, J., Hernández, A., Passariello, G., & Mora, F. (2001). Sistema telemédico de electrocardiografía basado en internet embebido. *Memorias del Taller de Telemedicina e Informática Médica (CD)*. Mérida. Venezuela.
- [9] Anchaluisa Abril, D. M. (2018). Sistema ambulatorio para el control de signos vitales y prevención de la diabetes mellitus (Bachelor's thesis, Universidad Técnica de Ambato. Facultad de Ingeniería en Sistemas, Electrónica e Industrial. Carrera Ingeniería en Electrónica y Comunicaciones).
- [10] Claudionor J. Coelho, Antonio O. Fernandes, Julio C. Conway, Fábio L. Correa, Hervaldo S. Carvalho, José M. Mata, "A biomedical wearable device for remote monitoring of physiological signals", *IEEE*, ISSN: 7803-7937, March 2003.

- [11] Cataldi, Z. (2000). Una metodología para el diseño, desarrollo y evaluación de software educativo (Doctoral dissertation, Facultad de Informática).
- [12] Relojes inteligentes: de dispositivos inútiles a guardianes de la salud. (2019, marzo Recuperado de <https://www.technologyreview.es/s/10982/relojes-inteligentes-de-dispositivos-inutiles-guardianes-de-la-salud>).
- [13] Decaro Santiago, L. G., Soriano Hernández, M. G., Ocaña Delgado, R. (2017). Los modelos de negocios y las tecnologías de la información y comunicación. Tlatemoani, 8, (26), 170-188. Recuperado de: <https://www.eumed.net/rev/tlatemoani/26/modelos-negocios.pdf>
- [14] Francia Huambachano, J. (2017). ¿Qué es Scrum?. Recuperado de: <https://www.scrum.org/resources/blog/que-es-scrum>
- [15] Proyectosagiles.org. (2021). ¿Qué es SCRUM?. Recuperado de: <https://proyectosagiles.org/que-es-scrum/>
- [16] Canive, T. (2020). Metodología SCRUM: ¿qué es y cómo aplicarlo en tu trabajo?. Recuperado de: <https://www.sinnaps.com/blog-gestion-proyectos/metodologia-scrum>
- [17] Schwaber, K., Sutherland, J. (2016). La Guía de Scrum.La Guía Definitiva de Scrum: Las Reglas del Juego. Recuperado de: <https://scrumguides.org/docs/scrumguide/v2016/2016-Scrum-Guide-Spanish.pdf#zoom=100>
- [18] Master, G. D. P. (2021). Roles y Responsabilidades en un Proyecto Scrum. Gestión de Proyectos Master. Recuperado de: <https://www.gestiondeproyectos-master.com/roles-y-responsabilidades-en-un-proyecto-scrum/>
- [19] Marcano, A. (2020). ASP.NET CORE 3.1 MVC Y WEB API: Ejemplos y Ejercicios en C# (Spanish Edition). Independently published.
- [20] Tamayo, A., Katrib, M. (2021). ASP.NET MVC y jQuery. Más poder, más control, ¿más esfuerzo? DotNetManía, 12-19.
- [21] Ramírez Guzmán, L. (2018). Desarrollo e implementación de un sistema de información de tipo móvil en donde se pueda agilizar y facilitar el servicio prestado en los lavaderos de autos en la ciudad de Bogotá. Colombia.
- [22] Rubio, J. C. (2019, 25 febrero). Qué es GIT y para qué sirve. OpenWebinars. <https://openwebinars.net/blog/que-es-git-y-para-que-sirve/>

- [23] Fuentes, J. R. (2015). Desarrollo de software ágil: Extreme Programming y Scrum. IT Campus Academy.
- [24] González, Y., Romero, Y. (2012). Patrón Modelo-Vista-Controlador. Telemática, (vol. 11, no 1, pp. 47-57).
- [25] Sagredo, J. G. (2012). Automatización de la codificación del patrón modelo vista controlador (MVC) en proyectos orientados a la Web. Ciencia ergo-sum. Revista Científica Multidisciplinaria de Prospectiva, (vol. 19, no 3, pp. 239-250).
- [26] Mistry, J. (2020). MVC Architecture [Figura]. 10 Reasons to Use Laravel PHP Framework For Web Development in 2021. Recuperado de: <https://www.monocubed.com/why-laravel-php-framework/>
- [27] Tepavac, I., Valjevac, K., Kliba, S., Mijač, M. (2015). Version Control Systems. Tools and Best Practices: Case Git.
- [28] Mudholkar, M., Mudholkar, P. (2017). A Beginner's Guide to Git and GitHub. 10.13140/RG.2.2.20126.89927.