



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
CARRERA TELECOMUNICACIONES

**“DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO DE
SISTEMA IOT PARA LA MONITORIZACIÓN DE DATOS DE
PRESIÓN ARTERIAL A TRAVÉS DE UN SENSOR
FOTOELECTRÓNICO”**

Trabajo de Titulación

Tipo: Propuesta Tecnológica.

Presentado para optar el grado académico de:

**INGENIERA EN ELECTRÓNICA, TELECOMUNICACIONES Y
REDES**

AUTORA:

MARCELA ESTEFANIA MORA CAMPANA

Riobamba – Ecuador

2022



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
CARRERA TELECOMUNICACIONES

**“DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO DE
SISTEMA IOT PARA LA MONITORIZACIÓN DE DATOS DE
PRESIÓN ARTERIAL A TRAVÉS DE UN SENSOR
FOTOELECTRÓNICO “**

Trabajo de Titulación

Tipo: Propuesta Tecnológica

Presentado para optar el grado académico de:

**INGENIERA EN ELECTRÓNICA, TELECOMUNICACIONES Y
REDES**

AUTORA: MARCELA ESTEFANIA MORA CAMPANA

DIRECTOR: Ing. DIEGO FERNANDO VELOZ CHÉRREZ. MSC.

Riobamba – Ecuador

2022

© 2022, **Marcela Estefanía Mora Campana**

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo cita bibliográfica del documento, siempre y cuando se reconozca el Derecho del Autor.

Yo, MARCELA ESTEFANIA MORA CAMPANA, declaro que el presente trabajo de titulación es de mi autoría y los resultados del mismo son auténticos. Los textos en el documento que provienen de otras fuentes están debidamente citados y referenciados.

Como autora asumo la responsabilidad legal y académica de los contenidos de este trabajo de titulación; el patrimonio intelectual pertenece a la Escuela Superior Politécnica de Chimborazo.

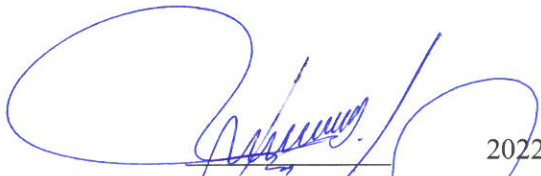

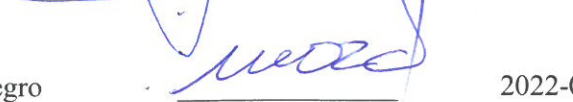
Riobamba, 19 de febrero de 2022



Marcela Estefania Mora Campana
060435840-8

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
CARRERA TELECOMUNICACIONES

El Tribunal del trabajo de titulación certifica que: El Trabajo de Titulación; tipo: Propuesta Tecnológica “**DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO DE SISTEMA IOT PARA LA MONITORIZACIÓN DE DATOS DE PRESIÓN ARTERIAL A TRAVÉS DE UN SENSOR FOTOELECTRÓNICO**” realizado por la señorita: **MARCELA ESTEFANIA MORA CAMPANA** ha sido minuciosamente revisado por los Miembros del Trabajo de Titulación, el mismo que cumple con los requisitos científicos, técnicos, legales, en tal virtud el Tribunal Autoriza su presentación.

	FIRMA	FECHA
Ing. Jonny Israel Guaiña Yungan PRESIDENTE DEL TRIBUNAL		2022-06-24
Ing. Diego Fernando Veloz Chérrez. MSC. DIRECTOR DE TRABAJO DE TITULACIÓN		2022-06-24
Ing. Franklin Giovanni Moreno Montenegro MIEMBRO DEL TRIBUNAL		2022-06-24

DEDICATORIA

El presente trabajo dedico con todo mi corazón a Dios quien es la fortaleza de mi vida y mis padres Lolo Mora y Teresa Campana quienes han sido el pilar fundamental para poder culminar este logro tan anhelado, mis padres me han brindado amor y paciencia en este camino estudiantil

Marcela.

AGRADECIMIENTO

Mi Agradecimiento también va dirigido a mis segundos padres mi abuelito Segundo Campana que, aunque ya no esté presente físicamente el a guiado mis pasos desde el cielo y a mi abuelita Evangelina Cordero que con sus bendiciones ha llenado mi corazón de fortaleza para continuar este camino

Para mí es gratificante haber culminado una meta en mi vida y a la vez pidiéndole a nuestro Padre Celestial que me ayude a continuar con muchas más metas profesionales

Marcela.

TABLA DE CONTENIDO

ÍNDICE DE TABLAS.....	xi
ÍNDICE DE FIGURAS.....	xii
ÍNDICE DE ANEXOS	xv
ÍNDICE DE ABREVIATURAS	xvi
RESUMEN.....	xvii
SUMMARY	xviii
INTRODUCCIÓN.....	1

CAPÍTULO I

1.	DIAGNÓSTICO DEL PROBLEMA.....	3
1.1.	Formulación del problema.....	3
1.2.	Sistematización del problema	3
1.3.	Justificación teórica.....	3
1.4.	Justificación aplicativa	4
1.5.	Objetivos.....	5
1.5.1.	<i>Objetivo general</i>	5
1.5.2.	<i>Objetivos específicos</i>	5

CAPÍTULO II

2.	REVISIÓN DE LA LITERATURA O FUNDAMENTOS TEÓRICOS.....	6
2.1.	Marco teórico referencial.....	6
2.1.1.	<i>Internet de las cosas</i>	6
2.1.2.	<i>Evolución del internet de las cosas</i>	6
2.2.	Características de IoT	7
2.3.	Componentes de IoT.....	8
2.4.	Un sistema IoT	10
2.4.1.	<i>La Arquitectura de tres capas</i>	10
2.4.2.	<i>Arquitectura de 4 Capas</i>	11
2.4.3.	<i>Arquitectura de 5 Capas</i>	11
2.4.4.	<i>Arquitectura Basada en SOA para sistemas de IoT</i>	12
2.5.	Interfaces de Programación de Aplicaciones (API).....	13
2.6.	REST.....	13
2.6.1.	<i>Principios de Rest</i>	14

2.7.	Seguridad de IoT	16
2.8.	Protocolo de Mensajería para Sistemas IoT.....	16
2.8.1.	<i>Protocolo HTTP.....</i>	17
2.8.2.	<i>Protocolo CoAP</i>	17
2.8.3.	<i>Protocolo MQTT.....</i>	18
2.8.4.	<i>Protocolo AMQP.....</i>	18
2.8.5.	<i>Protocolo WAMP</i>	18
2.8.6.	<i>Protocolo XMPP</i>	19
2.8.7.	<i>Protocolo STOMP.....</i>	19
2.9.	Elementos de IoT	20
2.9.1.	<i>Hardware.....</i>	20
2.9.2.	<i>Software para aplicación móvil.....</i>	20
2.9.2.1.	<i>React Native</i>	21
2.9.2.2.	<i>App Inventor.....</i>	21
2.9.3.	<i>Construcción de una App Inventor</i>	22
2.9.3.1.	<i>Native Script.....</i>	22
2.9.3.2.	<i>Android Studio.....</i>	23
2.9.3.3.	<i>Middleware.....</i>	24
2.9.4.	<i>Aplicaciones de IoT con análisis de la seguridad.....</i>	25
2.9.4.1.	<i>Sector Aeronáutica.....</i>	25
2.9.4.2.	<i>Sector Automoción</i>	25
2.9.4.3.	<i>Sector Telecomunicaciones</i>	25
2.9.4.4.	<i>Sector Edificios Inteligentes.....</i>	26
2.9.4.5.	<i>Sector Salud.....</i>	26
2.9.4.6.	<i>Sector Farmacéutico</i>	26
2.9.5.	<i>IOE</i>	26
2.9.6.	<i>IONT</i>	27
2.9.7.	<i>IoTTM o IoT de salud.....</i>	28
2.9.8.	<i>Medicina y Salud</i>	28
2.9.9.	<i>Telemedicina.....</i>	29
2.9.9.1.	<i>Aplicaciones de la telemedicina.....</i>	29
2.9.9.2.	<i>Escenarios típicos de aplicación del internet de las cosas en la salud.....</i>	30
2.9.10.	<i>Biosensores en el Internet de las Cosas asociadas al sector de la Salud.....</i>	32
2.9.10.1.	<i>Capacidad de los Biosensores.....</i>	32
2.9.11.	<i>Big Data</i>	32
2.9.11.1.	<i>LAS “V” DEL BIG DATA.....</i>	33
2.9.11.2.	<i>Volumen Masivo de Datos.....</i>	33

2.9.11.3.	<i>Velocidad Elevada en la generación y proceso de información</i>	33
2.9.11.4.	<i>Gran variedad en los Datos</i>	33
2.9.12.	<i>EDGE COMPUTING</i>	34
2.9.13.	<i>CLOUD COMPUTING</i>	35
2.9.13.1.	<i>Proveedor de Servicios Cloud Computing</i>	35
2.9.13.2.	<i>Nube de Alibaba</i>	36
2.9.13.3.	<i>Servicios web de Amazon</i>	37
2.9.13.4.	<i>Google Cloud</i>	37
2.9.13.5.	<i>IBM</i>	38
2.9.13.6.	<i>Microsoft Azure</i>	38
2.9.13.7.	<i>Oráculo</i>	38
2.9.14.	<i>Tipos de redes inalámbricas</i>	39
2.9.14.1.	<i>Redes de área corporal</i>	39
1.9.14.1.	<i>Redes de área local</i>	40
1.9.14.1.	<i>Redes de área Personal</i>	40
1.9.15.	<i>Tecnologías Inalámbricas</i>	41
1.9.15.1.	<i>WIFI</i>	41
1.9.15.2.	<i>BLUETOOTH</i>	43
1.9.16.	<i>ZIGBEE</i>	49
1.9.17.	<i>NFC</i>	51
1.9.18.	<i>RFID</i>	52
1.9.19.	<i>LORA</i>	52
1.9.19.1.	<i>LORAWAN</i>	52
1.9.20.	<i>Presión Arterial</i>	54
1.9.20.1.	<i>Presión Arterial Sistólica</i>	55
1.9.20.2.	<i>Presión Arterial Diastólica y Sistólica Saludable</i>	55
1.9.20.3.	<i>Método de Medición de la Presión Arterial</i>	56
1.9.21.	<i>Monitoreo Ambulatorio de Presión Arterial (MAPA)</i>	58
1.9.21.1.	<i>Funcionamiento</i>	58
1.9.21.2.	<i>Interpretación de Reportes</i>	58
1.9.22.	<i>Sensor Óptico PPG</i>	59
1.10.	<i>Tarjetas de Desarrollo</i>	60
1.10.1.	<i>Arduino</i>	60
1.10.2.	<i>NodeMCU Wifi & Bluetooth SoC Module</i>	61

CAPÍTULO III

3.	MARCO METODOLÓGICO.....	62
3.1.	Metodología de la investigación.....	62
3.1.1.	<i>Tipo de investigación</i>	<i>62</i>
3.1.2.	<i>Método de investigación</i>	<i>62</i>
3.1.2.1.	<i>Métodos Teóricos</i>	<i>63</i>
3.1.3.	<i>Técnicas de investigación</i>	<i>63</i>
3.2.	Desarrollo de la metodología	63
3.2.1.	<i>Matriz operacional de la metodología.....</i>	<i>64</i>
3.2.2.	<i>Dispositivo electrónico biomédico</i>	<i>65</i>
3.2.2.1.	<i>Sensor Biomédico.....</i>	<i>65</i>
3.2.3.	<i>Análisis de la Tarjeta de Desarrollo.....</i>	<i>67</i>
3.2.4.	<i>Análisis de las Tecnologías Inalámbricas.....</i>	<i>68</i>
3.3.	Diseño lógico del sistema	69
3.3.1.	<i>Sistema operativo</i>	<i>69</i>
3.3.2.	<i>Servidor de alojamiento.....</i>	<i>70</i>
3.3.3.	<i>Servidor Web.....</i>	<i>71</i>
3.3.3.1.	<i>Tipos de servidores web</i>	<i>71</i>
3.3.4.	<i>Base de Datos.....</i>	<i>72</i>
3.3.5.	<i>Protocolos de Comunicación cliente/servidor para IoT</i>	<i>73</i>
3.3.6.	<i>Aplicación Móvil.....</i>	<i>73</i>
3.4.	Diseño de la arquitectura del prototipo IoT	74
3.4.1.	<i>Sensor Biomédico</i>	<i>75</i>
3.4.2.	<i>Tarjeta de Desarrollo.....</i>	<i>77</i>
3.4.3.	<i>Plataforma IoT.....</i>	<i>81</i>
3.4.4.	<i>Instalación del sistema operativo</i>	<i>82</i>
3.4.5.	<i>Instalación del sistema operativo</i>	<i>83</i>
3.4.6.	<i>Instalación del servicio web.....</i>	<i>85</i>
3.4.7.	<i>Instalación de la base de datos</i>	<i>87</i>
3.4.8.	<i>Creación de la aplicación Android.....</i>	<i>91</i>
3.4.9.	<i>Implementación del prototipo biomédico.....</i>	<i>93</i>

CAPÍTULO IV

4.	RESULTADOS.....	95
4.1.	Lugar de estudio	95
4.1.1.	<i>Tamaño de la muestra</i>	<i>96</i>
4.1.2.	<i>Pruebas de funcionamiento.....</i>	<i>97</i>

4.1.2.1.	<i>Funcionamiento del Prototipo Biomédico</i>	97
4.1.3.	<i>Funcionamiento del Servidor VPS</i>	99
4.1.4.	<i>Funcionamiento de la Aplicación Android</i>	99
4.1.5.	<i>Comprobación de funcionamiento de los Cálculos</i>	100
4.2.	Resultados	101
4.2.1.	<i>Análisis del Prototipo Biomédico</i>	103
4.3.	Análisis Económico del Prototipo	106
CONCLUSIONES		108
RECOMENDACIONES		110
BIBLIOGRAFÍA		
ANEXOS		

ÍNDICE DE TABLAS

Tabla 1-2: Evolución de Bluetooth	44
Tabla 2-2: Tabla de Valores de Presión Arterial.....	56
Tabla 1-3: Matriz operacional de la metodología	64
Tabla 2-3: Tipos de Sensores Biomédicos	65
Tabla 3-3: Alternativas escala de Likert	66
Tabla 4-3: Comparación Likert del Sensor	66
Tabla 5-3: Análisis comparativo de las tarjetas de desarrollo.....	67
Tabla 6-3: Comparación Likert de la Tarjeta de Desarrollo	67
Tabla 7-3: Análisis comparativo de las tecnologías inalámbricas	68
Tabla 8-3: Comparación Likert Tecnologías Inalámbricas.....	68
Tabla 9-3: Características de los Sistemas Operativos.....	69
Tabla 10-3: Comparación Likert Sistema Operativo	69
Tabla 11-3: Características del Servidor de Alojamiento	70
Tabla 12-3: Comparación del Servidor del Servidor de Alojamiento.....	70
Tabla 13-3: Características de Servidor Web.....	71
Tabla 14-3: Comparación de Servidor Web.....	71
Tabla 15-3: Características de la Base de Datos	72
Tabla 16-3: Comparación de Base de Datos	72
Tabla 17-3: Características de los Protocolos de Comunicación cliente/servidor para IoT.....	73
Tabla 18-3: Comparación Likert de los Protocolos de Comunicación cliente/servidor para IoT	73
Tabla 19-3: Características de Aplicaciones Móviles	73
Tabla 20-3: Comparación Likert de las Aplicaciones Móviles.....	74
Tabla 21-3: Parámetros Requeridos por el profesional de la salud para diagnosticas al paciente luego del mapa.....	89
Tabla 22-3: Código del script procesamientos.php.....	90
Tabla 1-4: Localización Hospital Metropolitano	95
Tabla 2-4: Cantidad Hombres, Mujeres mayores de 65 años	96
Tabla 3-4: Análisis de Presión Arterial Equipo Médico, Prototipo Biomédico.....	104
Tabla 4-4: Análisis de Presión Arterial.....	105
Tabla 5-4: Análisis Económico del Prototipo	106

ÍNDICE DE FIGURAS

Figura 1-2:	IoT visto como red de redes	7
Figura 2-2:	Elementos principales de IoT	8
Figura 3-2:	Tipos de Arquitectura de IoT	10
Figura 4-2:	Arquitectura de IoT de 5 Capas	11
Figura 5-2:	Arquitectura Basada SOA para IoT	12
Figura 6-2:	Esquema de la Capa de Servicios SOA.....	13
Figura 7-2:	Modelo cliente/servidor con API REST	14
Figura 8-2:	Topología Cliente Servidor	15
Figura 9-2:	Estructura de un sistema REST basado en capas y uso de caché.....	15
Figura 10-2:	Pila de Familia de Protocolos para Sistemas IoT.....	16
Figura 11-2:	Arquitectura Básica de HTTP	17
Figura 12-2:	Entorno de CoAP	17
Figura 13-2:	Modelo de Protocolo AMQP	18
Figura 14-2:	Arquitectura XMPP para comunicación entre servidores y clientes.....	19
Figura 15-2:	Arquitectura del protocolo STOMP	20
Figura 16-2:	Visión Global App Inventor.....	21
Figura 17-2:	Visión Global App Inventor.....	22
Figura 18-2:	Native Script	23
Figura 19-2:	Android Studio.....	23
Figura 20-2:	Internet of Everything (IOE).....	27
Figura 21-2:	The Internet of Nano Things	28
Figura 22-2:	Estructura general de un sistema de telemedicina.....	30
Figura 23-2:	Salud centrada en el hogar	31
Figura 24-2:	Edge Computing	34
Figura 25-2:	Arquitectura de Ejemplo de Edge Computing	35
Figura 26-2:	Cuadrante mágico para servicios de plataforma e infraestructura en la nube ...	36
Figura 27-2:	Alibaba Group.....	37
Figura 28-2:	Google Cloud	37
Figura 29-2:	Servicios IaaS, PaaS de Azure	38
Figura 30-2:	Estándares Inalámbricos	39
Figura 31-2:	Ejemplo de Red de Área Local	40
Figura 32-2:	Redes de Área Personal.....	41
Figura 33-2:	Tecnología Wifi	41
Figura 34-2:	Topología punto a punto	42

Figura 35-2:	Topología Malla.....	42
Figura 36-2:	Topología Punto y Multipunto.....	43
Figura 37-2:	Tecnología Bluetooth.....	43
Figura 38-2:	Pila de Protocolo de Bluetooth.....	45
Figura 39-2:	Ejemplo de una Piconet.....	46
Figura 40-2:	Ejemplo de una Scatternet.....	46
Figura 41-2:	Formato de la Cabecera.....	47
Figura 42-2:	Estándar para el formato Acceso.....	48
Figura 43-2:	Tecnología Zigbee.....	49
Figura 44-2:	Conexión en la Topología Estrella.....	50
Figura 45-2:	Conexión en la Topología en Árbol.....	51
Figura 46-2:	Tecnologías NFC	51
Figura 47-2:	Arquitectura de red LoRaWAN	53
Figura 48-2:	Arquitectura de red LoRaWAN	54
Figura 49-2:	Instrumentos para la toma de la presión Arterial	55
Figura 50-2:	Puntos Anatómicos donde medir la presión Arterial	57
Figura 51-2:	Técnicas de Captura en el Fotoplesmitografía	59
Figura 52-2:	Placa Arduino Mega.....	60
Figura 53-2:	ESP 32 NODE MCU.....	61
Figura 1-3:	Esquema del prototipo del sistema IoT para el monitoreo de la presión arterial	75
Figura 2-3:	Arquitectura del sensor biométrico	76
Figura 3-3:	Sensor de presión arterial	76
Figura 4-3:	Líneas de código del algoritmo de calibración del sensor programado en el IDE de Arduino.....	77
Figura 5-3:	Diagrama de conexiones de los componentes electrónicos del prototipo biomédico	78
Figura 6-3:	Captura de pantalla de una parte del código configurado en la placa ESP32 en el IDE de Arduino.	78
Figura 7-3:	Líneas de código para envío de información a la tarjeta ESP32 por Bluetooth programado en el IDE de Arduino	79
Figura 8-3:	Líneas de código para toma de presión arterial programado en el IDE de Arduino	80
Figura 9-3:	Líneas de código para envío de información al servidor por HTTP programado en el IDE de Arduino	80

Figura 10-3:	Captura de pantalla del servidor privado virtual	82
Figura 11-3:	Captura de pantalla de Hostinger	83
Figura 12-3:	Captura de pantalla de las opciones configuradas en el servidor VPS.....	84
Figura 13-3:	SSH usada para la conexión cliente -servidor.....	84
Figura 14-3:	Registro de la llave SSH en el cliente	85
Figura 15-3:	Acceso al servidor por SSH	85
Figura 16-3:	Comprobación del servidor web funcionando correctamente en la dirección http://31.170.165.164/.....	87
Figura 17-3:	Ingreso y configuración de la base de datos, tabla y columnas para el almacenamiento de información.	88
Figura 18-3:	Código del scrip post-data.php.....	89
Figura 19-3:	Código del script procesamientos.php	90
Figura 20-3:	Diseño visual de la Aplicación en App Inventor.....	91
Figura 21-3:	Programación de los bloques del registro del paciente y red Wifi.....	92
Figura 22-3:	Programación de bloques para mostrar los resultados en la aplicación Android	92
Figura 23-3:	Scripts usados por la aplicación móvil para mostrar los resultados	93
Figura 24-3:	Proceso de implementación del prototipo biomédico	93
Figura 25-3:	Prototipo biomédico instalado en un guate de poliéster listo para su uso.....	94
Figura 26-3:	Explicación de uso e implementación del prototipo biomédico conjuntamente con el equipo médico a una paciente	94
Figura 1-4:	Localización Hospital Metropolitano.....	96
Figura 2-4:	Tarjeta ESP 32	97
Figura 3-4:	Wifi de Tarjeta de Desarrollo ESP 32.....	98
Figura 4-4:	Sensor Biomédico	98
Figura 5-4:	Funcionamiento del Servidor VPS.....	99
Figura 6-4:	Validación del funcionamiento de los botones y	100
Figura 7-4:	Comprobación de los cálculos realizados en la parte de Administración	101
Figura 8-4:	Equipo Médico CONTEC PM50	101
Figura 9-4:	Opciones de la parte de administración	102
Figura 10-4:	Tabla de valores de Presión Arterial del software ABPM utilizado por el doctor	103

ÍNDICE DE ANEXOS

ANEXO A: CÓDIGO USADO PARA LA PROGRAMACIÓN DEL DISPOSITIVO
BIOMÉDICO EN ARDUINO

ANEXO B: PHP DEL SERVIDOR WEB

ANEXO C: INFORMACIÓN DEL SENSOR

ÍNDICE DE ABREVIATURAS

IoT:	Internet of things (Internet de las Cosas)
IOE:	Internet of everything
IONT:	Internet de las Nano Cosas
LED:	Light-emitting diode (Diodo emisor de luz)
WIFI:	Wireless Fidelity (Fidelidad inalámbrica)
HTTP:	Protocolo de Transferencia de Hipertexto
MySQL:	My Structured Query Language (Lenguaje de Consulta Estructurado)
COAP:	Asset Liability Committee (Comité de Activos y Pasivos)
XMPP:	Protocolo extensible de mensajería y comunicación de presencia
BD:	Base de Datos
OMS:	Organización Mundial de la Salud
ECV:	(Enfermedad Cardiovascular)
PA:	Presión Arterial
PAS:	Presión Arterial Sistólica:
PAD:	Presión Arterial Diastólica
I²C:	(I ² C, del inglés Inter Integrated Circuit) Circuito Inter integrado
SQL:	Structured Query Language (Lenguaje de Consulta Estructurada)
AWS:	Amazon Web Services
RFID:	Radio Frequency Identification (Identificación por Radiofrecuencia)
WPAN:	Wireless Personal Área Network (Red Inalámbrica de Área Personal)
NFC:	Near Field Communication (Comunicación de campo cercano)
LORA:	Long Range (Largo alcance)

RESUMEN

El objetivo del presente trabajo de integración curricular fue diseñar e implementar un prototipo de sistema IoT para la monitorización de datos de presión arterial a través de un sensor fotoelectrónico. Este prototipo permite tomar los datos de presión arterial sistólica y diastólica del paciente mediante el sensor fotoelectrónico y la tarjeta de desarrollo, luego de ello procede a enviar la información a un servidor web almacenándola en un gestor de base de datos, por consiguiente, en la aplicación móvil se admite el registro del paciente en el sistema y el acceso a los resultados de la información de presión arterial. Para poder determinar el correcto funcionamiento del prototipo biomédico, se realizó la comparación conjunta de la toma de información arterial con la ayuda del equipo médico a mujeres mayores a 60 años, las pruebas se realizaron a 11 pacientes en la clínica metropolitana, siendo las muestras utilizadas para los cálculos y la comparación de resultados entre el equipo médico y el prototipo biomédico: Una vez realizada la medición y con los datos obtenidos indica que el prototipo implementado es confiable. Los resultados fueron presentados en la aplicación móvil. Se concluye que mediante la aplicación móvil es posible mostrar el diagnóstico del paciente conjuntamente con los cálculos de presión arterial además de los valores del monitoreo, donde el médico tratante será quien dictamine y valide el tipo de hipertensión y el tratamiento del paciente. Se recomienda el uso de más sensores en futuras investigaciones para el diseño y la implementación de un prototipo de sistema IoT para la monitorización de datos de presión arterial.

Palabras Clave: <INTERNET DE LAS COSAS (IoT)> <SENSOR FOTOELECTRÓNICO> <HIPERTENSIÓN> <PRESIÓN ARTERIAL> <MONITORIZACIÓN DE DATOS>.

1396-DBRA-UTP-2022

ESPOCH - DBRAI
PROCESOS TÉCNICOS Y ANÁLISIS
BIBLIOGRÁFICO Y DOCUMENTAL



01 JUL 2022

REVISIÓN DE RESUMEN Y BIBLIOGRAFÍA

Por: Rosales Hora: 15:08

ABSTRACT

The objective of this curricular integration work was to design and implement a prototype of an IoT system for monitoring blood pressure data through a photoelectronic sensor, it was decided to design and implement a prototype system in the internet of things, under Android mobile device environments for blood pressure data monitoring through a photoelectronic sensor. It allows taking the systolic and diastolic blood pressure data of the patient through the photoelectronic sensor and the development card, after which it proceeds to send the information to a web server, storing it in a database manager, therefore, in the mobile application patient registration in the system and access to blood pressure information results are supported. In order to determine the correct functioning of the biomedical prototype, the joint comparison of arterial information collection was carried out with the help of the medical team to women over 60 years of age, the tests were carried out on 11 patients in the metropolitan clinic, the samples used to be for calculations and comparison of results between the medical equipment and the biomedical prototype:

Once the measurement has been made and with the data gotten, it indicates that the implemented prototype is reliable. The results were presented in the mobile application. It is concluded that through the mobile application it is possible to show the patient's diagnosis together with the blood pressure calculations in addition to the monitoring values, where the treating physician will be the one who dictates and validates the type of hypertension and the patient's treatment: It is recommended that for the design and implementation of a prototype IoT system for monitoring blood pressure data, due to future research the use of more sensors to accurately measure blood pressure.

Keywords: <INTERNET OF THE THINGS>, <PHOTOELECTRONIC SENSOR>, <HYPERTENSION >, <BLOOD PRESSURE>, <ESP32>, <MAX32664 SENSOR>



MSc. Wilson G. Rojas
NOMBRE Y FIRMA PROFESOR

C.I 0602361842

INTRODUCCIÓN

Antecedentes

La enfermedad Cardio Vascular (ECV) es una de las principales causas de morbi-mortalidad a nivel mundial, ya que significa un tercio de la mortalidad y un décimo de la morbilidad anuales en todo el mundo: El desarrollo de la enfermedad Cardio Vascular están ligados directamente a factores como son la edad, sexo, antecedentes patológicos familiares, dieta inadecuada, consumo de tabaco y alcohol, obesidad, enfermedades como hipertensión arterial, dislipidemias y diabetes de esta manera incrementando el índice de mortalidad y morbilidad. (Alwan; Ala, 2010).

Según los estudios realizados en el Ecuador en marzo del 2020, La Organización Panamericana de la Salud (OPS), determino que las poblaciones con riesgo de padecer enfermedades cardiovasculares, mediante datos estadísticos, los resultados obtenidos fueron abrumadores al evaluar datos recogidos de 2231 personas entre 18 y 69 años de edad, siendo un 30% de la población adulta, 40 y 69 años, susceptible de correr riesgos de padecer enfermedades cardiovasculares. (OPS, 2016)

En Ecuador, los datos proporcionados en el 2019 por el Instituto Nacional de Estadísticas y Censos (INEC), determinan varios factores de riesgo que provocan el fallecimiento de 36.605 personas es decir aproximadamente el 50% de la cifra total de muertes registras en ese año. (Lugmaña et al., 2020). Dentro de las enfermedades cardiovasculares se encuentra la hipertensión (Tensión arterial alta, HTA) la cual es unas enfermedades crónicas con mayor importancia a nivel mundial. Debido a que esta enfermedad cobra 10 millones de vidas al año y las personas que padecen hipertensión después de ciertos años (alrededor de 10 años) tienden a sufrir daños importantes en el corazón, los riñones, cerebro, ojos y arterias

El monitoreo y registro de variables fisiológicas, resulta esencial en la medicina moderna. Tal es el caso de la presión arterial, una variable asociada a problemas graves de salud como ataques cardíacos y accidentes cerebro-vasculares. El diagnóstico y tratamiento de patologías asociadas a esta variable, requiere de medidas frecuentes que implican el desplazamiento regular del paciente a centros clínicos y hospitalarios.

Con estos antecedentes, el presente trabajo de titulación denominado diseño e implementación de un prototipo de sistema IoT para la monitorización de datos de presión arterial a través de un sensor foto electrónico. El sistema suministrará, información necesaria que permitirá garantizar

el bienestar de la persona que padece dicha enfermedad, la investigación busca como elemento adicional constituirse como una herramienta de soporte a las actividades de cuidados que realizan sus familiares o cuidadores.

CAPÍTULO I

1. DIAGNÓSTICO DEL PROBLEMA

1.1. Formulación del problema

¿El desarrollo de un prototipo IoT permitirá la detección temprana mediante el control de la presión arterial de forma remota y de manera confiable?

1.2. Sistematización del problema

¿Cuáles son las características y valores adecuados de la presión arterial en personas con enfermedades cardiovasculares?

¿Cuál es el estado del arte de los sistemas IoT?

¿Qué tipo de software y hardware serán utilizados en la implementación del prototipo?

¿Qué metodología es necesario utilizar para el desarrollo del proyecto?

¿Cómo diseñar un sistema IoT que recolecte información sobre la presión arterial?

¿Cuáles son los requerimientos del diseño para la implementación del sistema IoT a desarrollar?

¿Cuál es la arquitectura para un sistema IoT que permita medir eficientemente los parámetros de presión arterial?

1.3. Justificación teórica

Según estadísticas de la Organización Mundial de la Salud - OMS, en 2018 cerca de 1.130 millones de personas en el mundo presentaron algún tipo de alteración de la presión arterial, (OMS, 2020). Existen dos métodos para determinar la presión arterial, que se emplean según la condición del paciente y las necesidades de atención médica. La gran mayoría de dispositivos automáticos y semiautomáticos que se consiguen en el mercado emplean este último. El mecanismo óptico del fotoplethismografía (PPG) permite que los doctores y los investigadores ensanchen la extensión de la herramienta de la investigación y que habiliten más usos para el dispositivo. Por ejemplo, los

nuevos avances médicos han integrado PPG en la tecnología usable para en camino la supervisión y la administración de la actividad del corazón.

Las personas que necesitan llevar el control de la presión arterial comúnmente son aquellas que sufren hipertensión, la cual es una enfermedad que puede ser tratada si se toman las medidas adecuadas con responsabilidad, tales como la reducción de consumo de sal, una dieta adecuada, y el monitoreo de la presión arterial. Para ello es necesario que las personas hipertensas conozcan y controlen su presión arterial, para evitar complicaciones a largo plazo. Pudiendo ser la mayoría de estos factores aditivos, tal como ocurre con la obesidad y la ingesta de alcohol.

1.4. Justificación aplicativa

Los usuarios del sistema IoT, podrán hacer uso de manera remota a través de Internet, con la posibilidad de usar una aplicación móvil. Para facilitar la administración y procesamiento de la información los datos deben ser almacenados, un Framework brindará a cualquier dispositivo periférico que intentemos conectar, la capacidad de almacenamiento de información. Estas interfases serán implementadas de acuerdo a las características y necesidades del sensor con el que se vaya a interactuar. Posterior a ello, una Base de Datos brinda la posibilidad de recolectar la información; con ello proceder con la toma de decisiones. Otra ventaja que ofrece el sistema IoT es el manejo de servicios web, lo que permite armar las interfases gráficas para usuarios. Esto se debe a que los servicios web trabajan bajo el protocolo estándar WSDL10, que permite la interoperabilidad con diferentes plataformas que puedan conectarse a una red, contemplando de esta manera la configuración de diferentes tipos de alarmas, pudiendo ser las mismas ejecutadas en forma recurrente en caso que fuese necesario.

Al dar la importancia debida al control de la presión arterial surge la necesidad de que exista la posibilidad de monitorear continuamente este signo vital. El sistema IoT propone la recolección de datos y posterior a ello su análisis e interpretación de información mediante el profesional de la salud.

La señal biológica se obtendrá por medio del sensor fotoeléctrico que hace uso de la fotopleletismografía para la toma de valores de presión arterial y signos (Quiroz Estrada, Acosta Amaya y Torres Villa, 2021). Para el desarrollo del prototipo del sistema IoT se analizarán las redes de comunicaciones más adecuadas que interconectarán los diferentes dispositivos.

1.5. Objetivos

1.5.1. Objetivo general

Diseñar e Implementar de un prototipo de sistema IoT para la monitorización de datos de presión arterial a través de un sensor fotoelectrónico

1.5.2. Objetivos específicos

- Analizar las características y el comportamiento de la presión arterial en personas.
- Revisar el estado del arte de los sistemas IoT.
- Diseñar e Implementar un sistema IoT que permita recolectar y monitorizar datos de presión arterial.
- Desarrollar una aplicación para la monitorización remota.
- Evaluar el sistema IoT implementado para la monitorización de presión arterial.

CAPÍTULO II

2. REVISIÓN DE LA LITERATURA O FUNDAMENTOS TEÓRICOS

En el presente capítulo se investigó el comportamiento de la enfermedad de Presión Arterial y sus efectos en la salud humana, además se investigó las características del sensor de fotoplestitografía, también las tecnologías existentes para la comunicación. Se especifica las características de los elementos electrónicos, tarjetas de desarrollo y la tecnología de comunicación inalámbrica para el envío y recepción de datos

2.1. Marco teórico referencial

2.1.1. *Internet de las cosas*

El Internet de las Cosas es un término que se utiliza para describir la conectividad de la red, se extiende a objetos, sensores y equipos de uso diario de las personas, que no son computadoras con la capacidad de conexión como base para el desarrollo de servicios cooperativos y aplicaciones independientes (Rose, al et., 2015, p. 5), permitiendo que los elementos se puedan intercambiar y consumir datos sin tener mucha intervención humana y se los pueda identificar ya que de muchas aplicaciones se obtiene datos, el mismo que ha transformado radicalmente la comunicación, la tecnología permite disponer de aplicaciones para la conexión con el Internet, con el tiempo la red ha pasado de Internet de las Personas a la de Internet de las cosas (IoT) (Rose, al et., 2015, p. 5)

2.1.2. *Evolución del internet de las cosas*

La evolución del Internet tiene una gran capacidad de procesar los datos los cuales son enviados a partir de los sensores y las redes avanzadas de comunicaciones con procesos de análisis basados en el Big Data, siendo una gran evolución gracias a los avances tecnológicos que permiten otra forma de comunicación de máquina a máquina, posee un gran impacto en la vida de las personas, los negocios y el comercio. (Morales, 2019, p. 4). además de un gran crecimiento y mejora en sus modelos de negocio provenientes de la tecnología y los diferentes elementos que permiten conectar a Internet “cualquier cosa”, como lo muestra la Figura 1-1(Morales, 2019, p. 4).

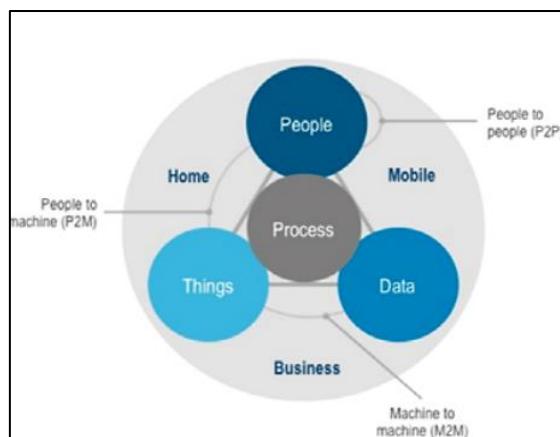


Figura 1-2: IoT visto como red de redes

Fuente: (Morales, 2019, p. 4)

2.2. Características de IoT

Se puede establecer algunas características fundamentales que identifican a IoT y que hacen de ella una infraestructura global para proporcionar servicios avanzados a los usuarios finales.

- **Objetos/Things/Dispositivos:** Los componentes Imperativo de IoT son los dispositivos denominada objetos o simplemente cosas. Como ya se ha mencionado, los objetos cotidianos no solo las computadoras son los principales actores de IoT. Tienen que ser identificables, reconocibles, localizables, direccionales, y/o controlables (Youngblood et al., 2005, p. 1513-1514).
- **Conectividad:** Una de las características ampliamente aceptada es la presencia de una infraestructura de red o conectividad de red, que habilite la comunicación de los dispositivos IoT entre su integración perfecta y un esquema de direccionamiento único
- **Datos:** El verdadero valor de IoT en los datos asociados a cada dispositivo se lo obtienen a partir de la interconexión entre ellos: Los datos deben ser el primer paso hacia la acción y la inteligencia, sin embargo, solo la información no es suficiente, el análisis y el uso inteligente de estos datos ya la capacidad de vincularlos con otros datos amplía su valor. (Shen et al., 2013)
- **Autonomía:** Es una característica recurrente que caracteriza a los dispositivos IoT, deben tener la capacidad de realizar la mayoría de sus tareas sin la intervención humana, o de otros dispositivos y un grado de control sobre sus propias acciones, y su propio estado interno según el contexto como las condiciones cambiantes por el entorno detectado. (Fortino, Guerrieri y Russo, 2012, p. 44-50)

- **Servicios:** Los servicios que pueden ser complejos o simples están disponibles para interactuar con los dispositivos IoT que permiten consultar su estado y cualquier información asociada a ellos, teniendo en cuenta las restricciones, como protección de la privacidad y coherencia semántica entre los dispositivos IoT y sus correspondientes objetos virtuales. (Gunturi, Kotha y Srinivasa Reddy, 2012, p. 20)
- **Heterogeneidad:** La heterogeneidad de los dispositivos IoT plantea problemas de interoperabilidad y es otra característica a menudo de primordial importancia a considerar para el desarrollo y adopción de esta tecnología, IoT aborda una gran adversidad de dispositivos generalmente con capacidades de comunicación y computación muy básicas que desafía la suposición de que cualquier dispositivo presenta una pila de protocolos completa.

2.3. Componentes de IoT

La funcionalidad de IoT, requiere de varios componentes complementarios que en conjunto ayudan a reducir la brecha entre el mundo físico y virtual. Entre los principales componentes que se han identificado y descrito se pueden resumir en la Figura 2-1 (Al-Fuqaha et al., 2015, p. 9)



Figura 2-2: Elementos principales de IoT
Fuente: (Yachirema Vargas, 2019, p. 21)

Identificación: Es la capacidad que permite identificar de forma única los dispositivos, por lo tanto, es fundamental para el éxito de IoT. No solo permitirá identificar miles de millones de dispositivos, sino también controlar estos dispositivos remotamente a través de Internet. (Prabhu, 2013, p. 26).

Detección/Actuación: Las tecnologías habilitadas, un tema clave para IoT es el desarrollo de medios apropiados para permitir interacciones con el entorno, se espera que los componentes de detección estén también representados por las tecnologías de redes de sensores inalámbricos (WSN), la capacidad de las WSN para detectar el entorno y auto organización en redes ad hoc representan una perspectiva importante de IoT.

Comunicación: Permite que los dispositivos se pueden enviar y recibir información, generalmente los dispositivos IoT operan con un bajo consumo de energía de enlaces de comunicación ruidosos y con altos niveles de pérdidas de información. Se requiere de protocolos de comunicación que se encuentren especializadas para realizar las tareas de comunicación. Existen diferentes protocolos de comunicación que facilitan la comunicación IoT como Zigbee, LoRA (Alliance, 2015,p 19) Z-WAVE, Wi-Fi, Bluetooth,(McDermott-Wells, 2005, p. 33-35) 6LowPAN y LTE-Advanced (Crosby, 2013, p. 31)

Computación: Entre las diferentes plataformas de hardware que permiten realizar el procesamiento en aplicaciones IoT se descartan Arduino, Intel Galileo, Raspberry Pi, Beaglebone, entre otros, el software de procesamiento, los sistemas operativos son vitales para permitir el procesamiento de un dispositivo, entre ellos se encuentran Contiki RTOS (Dunkels, Gronvall y Voigt, 2004, p. 455-462) TinyOS (Levis et al., 2005, p. 3) LiteOS (Cao et al., 2008, p. 235) y Riot OS (Baccelli et al., 2013, p. 3)

Servicios: Se pueden clasificar los servicios de IoT en cuatro clases (Gigli y Koo, 2011, p. 2) tales como: servicios relacionados con la identidad, servicios de agregación de la información, servicios basados en la colaboración y servicios únicos.

- **Arquitectura de Servicios relacionados con la identidad:** Es la base de todos los demás servicios, por lo tanto, se encarga de identificar la identidad de los dispositivos
- **Servicios de agregación de información:** Son aquellos que recopila y resume las medidas sensoriales sin procesar
- **Servicios de colaboración:** Actúan sobre los servicios de agregación de información y utilizan los datos obtenidos para toma acciones y reaccionar en consecuencia
- **Servicios ubicuos:** Estos servicios proporcionan servicios de colaboración en cualquier momento, en cualquier lugar, el objetivo final de toda aplicación IoT es alcanzar el nivel de servicios ubicuos

2.4. Un sistema IoT

En cuanto a la arquitectura de IoT, no existe actualmente un acuerdo sobre el modelo a utilizar, por ello existen diferentes instituciones y/o investigadores lo que han propuesto modelos equivalentes (Sethi y Sarangi, 2017, p. 2). Básicamente se pueden agrupar en los 3 tipos que podemos observar a continuación, como lo muestra la Figura 3-1.

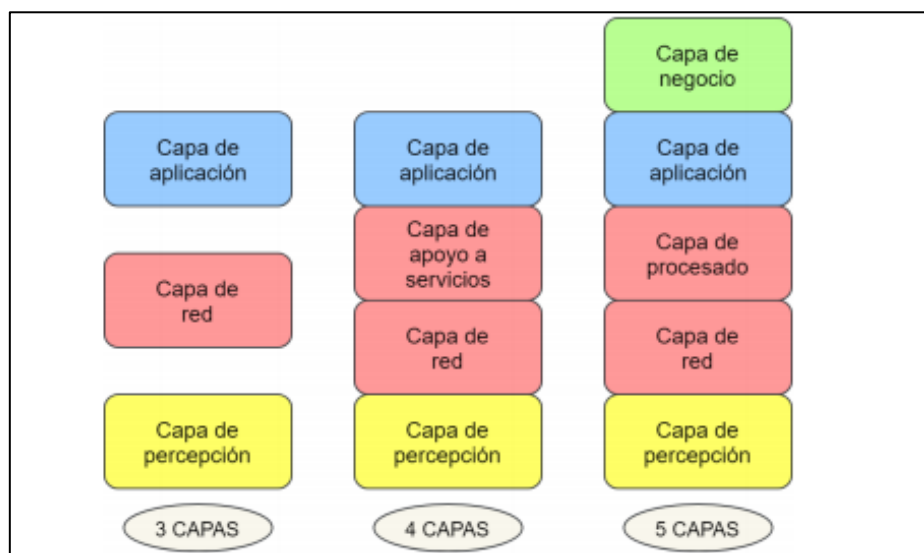


Figura 3-2: Tipos de Arquitectura de IoT

Fuente: (Ortiz Monet, 2019, p. 7)

2.4.1. La Arquitectura de tres capas

La arquitectura de tres capas tiene diferentes funciones específicas y disponen sus propios protocolos que son lenguajes de comunicación entre las capas, constan de tres capas que son Percepción, capa Red y capa Aplicación. (Yacchirema Vargas, 2019, p. 26)

- **Capa de Percepción:** Hace referencia a la capa física. Actúan en esta capa los sensores, captando la información requerida para cada tipo de aplicación, como pueden ser diferentes parámetros físicos o la identidad de otros objetos inteligentes del entorno.
- **Capa de red:** Es la que se encarga de la conexión entre objetos inteligentes, o de la conexión de estos con servidores y dispositivos de red. También se encarga de transmitir y procesar los datos recogidos por los sensores.
- **Capa de aplicación:** La capa de aplicación es la responsable de dar los servicios de cada tipo de aplicación a los usuarios. Entre los diferentes tipos de aplicaciones en las que se puede

implementar IoT podemos encontrar hogares, ciudades inteligentes que optimicen los recurso y tiempo de proceso.

2.4.2. *Arquitectura de 4 Capas*

La arquitectura de cuatro capas es bastante similar a la arquitectura de 3 capas. Su diferencia principalmente con la arquitectura de cinco capas, es que no cuenta con la denominada capa de negocio, en la cual se gestiona todo el sistema IoT, incluyendo tanto las aplicaciones como los modelos de negocio, ganancias y privacidad de los usuarios.

2.4.3. *Arquitectura de 5 Capas*

Es el modelo generalmente más utilizado, la arquitectura desde un punto de vista técnico describe a IoT, sin embargo, debido al gran desarrollo que está teniendo es necesario considerar también la gestión y el modelo de negocio en esta arquitectura. Así, se podría considerar que la arquitectura genérica del IoT consta de 5 capas, añadiendo la capa de Procesamiento y la capa de Negocio al modelo básico de tres capas, los cuales son: Capa de percepción o de objetos, Capa de red o de abstracción, Capa de procesado (middleware layer), Capa de aplicación Capa de negocio como lo muestra la Figura 4-1 (Domínguez Cobos, 2016, p. 7)

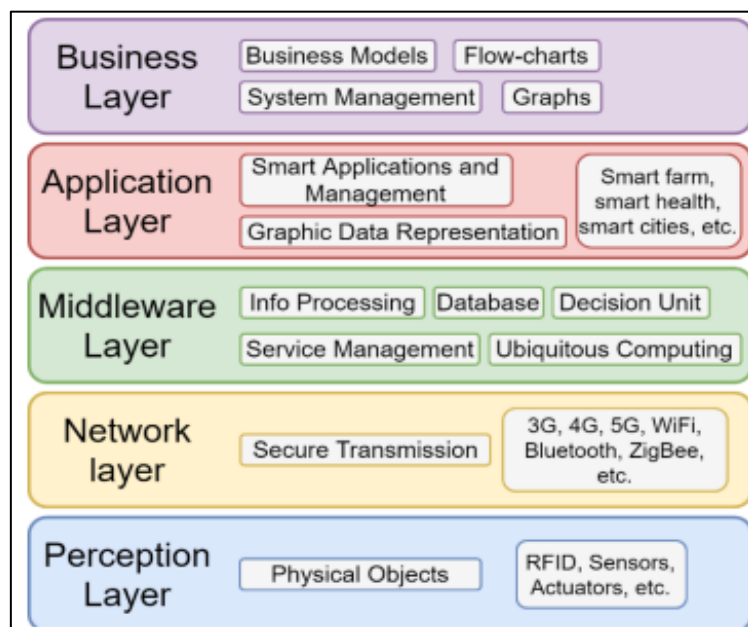


Figura 4-2: Arquitectura de IoT de 5 Capas
Fuente: (Ortiz Monet, 2019, p. 9)

2.4.4. Arquitectura Basada en SOA para sistemas de IoT

Para el desarrollo de sistemas de IoT la arquitectura basada en SOA propuesta se conforma de 4 capas, tal como se muestra en la Figura 5-1



Figura 5-2: Arquitectura Basada SOA para IoT

Fuente:(Sosa-Reyna, Tello-Leal y Lara-Alabazares, 2016, p. 3)

Capa de Objetos: La capa de objetos, los sistemas inteligentes mediante etiquetas (tags) o sensores, son aquellos capaces de detectar automáticamente el medio ambiente y el intercambio de datos entre los dispositivos. (Li, Xu y Zhao, 2015, p. 243-259).

Capa de Red: En el enfoque SOA, la capa de red es crucial en cualquier enfoque de IoT, permitiendo funcionalidades de QoS, gestión eficiente de energía en la red y en las cosas, procesamiento de señales y datos, seguridad y privacidad, entre otras.

Capa de Servicio: La capa de servicio se basa en la tecnología de middleware, es fundamental para consumir servicios y la ejecución de aplicaciones de IoT, donde las plataformas de hardware y software pueden ser reutilizables, como se muestra en la Figura 6-1



Figura 6-2: Esquema de la Capa de Servicios SOA
Fuente: (Bandyopadhyay y Sen, 2011, p. 49-69)

Capa de Aplicación: Capa responsable de la entrega de las aplicaciones a los diferentes usuarios de IoT. El desarrollo de aplicaciones en IoT se ha centrado en las áreas de salud, agricultura, transporte, ciudades inteligentes, automatización de casas, sistemas complejos para la toma de decisiones, gestión de uso de agua, etc (Fang et al., 2014, p. 1596-1605).

2.5. Interfaces de Programación de Aplicaciones (API)

Una API de REST, o API de RESTful, es una interfaz de programación de aplicaciones (API o API web) que se ajusta a los límites de la arquitectura REST y permite la interacción con los servicios web de RESTful. El informático Roy Fielding es el creador de la transferencia de estado representacional (REST) (Red Hat, 2020, p. 1)

Para la implementación de la API se usó la técnica de ingeniería de software REST, la cual hace uso sólo del protocolo HTTP para la entrega de recursos. Esto hace que operacionalmente la API sea eficiente en el uso de recursos y accesible a un amplio espectro de clientes. (Atencio Flores y Mamani Machaca, 2017, p. 25)

2.6. REST

REST es aquel que utiliza el protocolo HTTP como medio de transporte para crear servicios web ligeros, flexibles y escalables. (Valencia Altamirano, 2015, p. 13). Es un protocolo de comunicación creado para el intercambio y manipulación de datos en internet de forma muy simple, se basa su

funcionamiento en el modelo cliente-servidor, ha permitido que diferentes servicios en la web lo comiencen a utilizar, como las plataformas de internet de las cosas, como se muestra en la Figura 7-1 representa de manera gráfica el uso de la API dentro del modelo cliente/servidor.

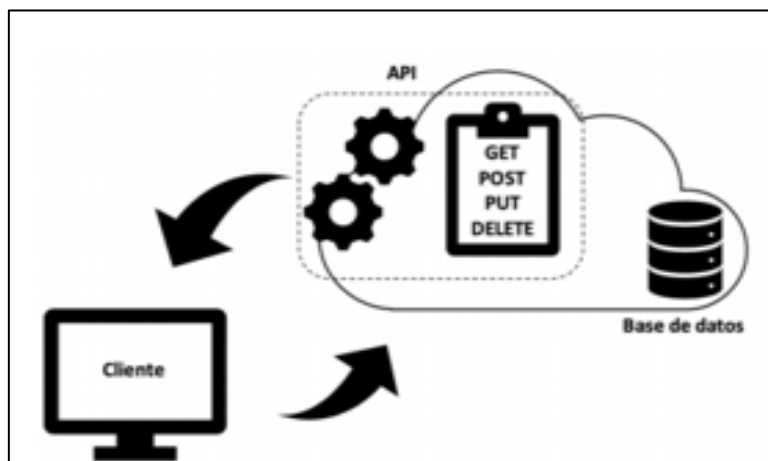


Figura 7-2: Modelo cliente/servidor con API REST
Fuente:(Benavidez Gómez y García Acevedo; Jesús David, 2019, p. 33)

Podemos identificar dos clases principales de Servicios Web:

Servicios Web compatibles con REST, en cuyo caso el objetivo principal del servicio es manipular representaciones XML de recursos web utilizando un conjunto uniforme de operaciones sin estado

Servicios Web arbitrarios, en el que el servicio puede exponer un conjunto arbitrario de operaciones. Ambas clases de servicios Web utilizan URIs para identificar los recursos y el uso de protocolos web (como HTTP y SOAP 1.2) y formatos de datos XML para la mensajería (Fernández, 2015, p. 16).

2.6.1. Principios de Rest

Cliente Servidor: Debe tener el sistema una arquitectura por capas de tipo cliente-servidor, la mayoría de los sistemas WEB ya que al separar las tareas de la interfaz de usuario de las de almacenamiento de datos, mejoramos la portabilidad de la interfaz de usuario a través de múltiples plataformas, como se muestra en la Figura 8-1 Topología Cliente Servidor

2.7. Seguridad de IoT

Uno de los aspectos más importantes es la seguridad en IoT. Están constantemente los dispositivos captando información personal y, por su naturaleza, llevando el mundo real a Internet y viceversa. Esto permite contar con tres categorías de riesgos entorno a la seguridad informática en dispositivos IoT, estos son:

La seguridad de la nube: La seguridad en la nube, las amenazas más urgentes vienen del entorno de la empresa o del entorno de la nube al que estos dispositivos inteligentes están conectados.

La seguridad del dispositivo: En la seguridad del dispositivo miles de millones de dispositivos se encuentran conectados y va a aumentar el uso de las aplicaciones de software y de los datos se encuentran en los recursos de las empresas, y también en los dispositivos de consumo, lo que implica nuevos puntos de ataque para los hackers maliciosos (Beltrán, 2016, p. 15-24).

La gestión del ciclo de vida de la seguridad: El agregado de nuevos dispositivos, el desmantelamiento del dispositivo al final de su vida útil, la integración del dispositivo en un nuevo ecosistema de la nube o viceversa, la gestión se descarga de firmware/software seguros son actividades que necesitan una gestión completa de las identidades, las claves y los tokens (Gemalto, 2016, p. 3).

2.8. Protocolo de Mensajería para Sistemas IoT

Presentan los cuatro protocolos de mensajería aceptados y de gran importancia para sistemas IoT como son MQTT, CoAP, CoAP y HTTP los cuales se encuentran en la parte superior de la pila de familia de protocolos como lo muestra en la Figura 10-1 se indica la Pila de familia de protocolos de sistemas IoT

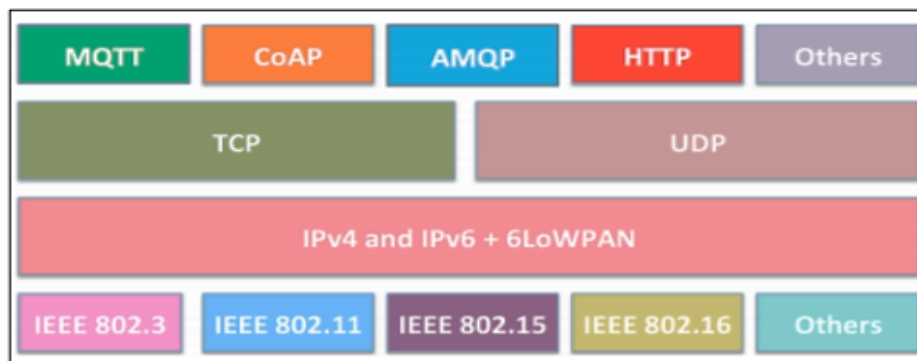


Figura 10-2: Pila de Familia de Protocolos para Sistemas IoT

Fuente: (Naik, 2017, p. 1-7)

2.8.1. Protocolo HTTP

Protocolo de Transferencia de Hipertexto: Es un sencillo protocolo cliente-servidor que articula los intercambios de información entre los clientes Web y los servidores HTTP. Se basa en sencillas operaciones de solicitud/respuesta. Envía datos el servidor a través del URI y el cliente recibe datos a través de un URI particular, como lo muestra la Figura 11-1 de la Arquitectura Básica de HTTP

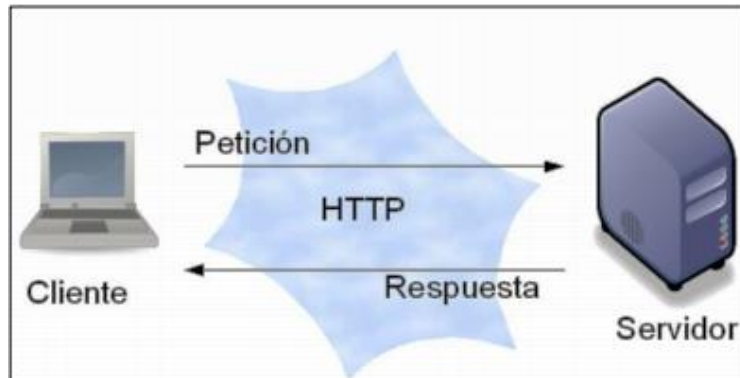


Figura 11-2: Arquitectura Básica de HTTP

Fuente: (Pérez Leones, 2019)

2.8.2. Protocolo CoAP

Esta desarrollado CoAP principalmente para interoperar con HTTP y RESTful Web a través de simples proxies. MQTT, a diferencia de CoAP usa el Identificador Universal de Recursos (URI) en lugar de los temas (Naik, 2017, p. 9) . Representa REST una forma simple de intercambiar datos entre clientes y servidores sobre HTTP, puede observarse como un protocolo de conexión que depende de la arquitectura cliente/servidor. Se utiliza en aplicaciones móviles y de redes sociales (Pérez Leones, 2019), como lo muestra la Figura 12-1 se indica un Entorno de CoAP

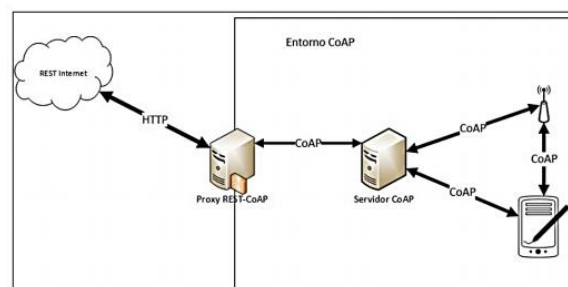


Figura 12-2: Entorno de CoAP

Fuente: (Pérez Leones, 2019)

2.8.3. *Protocolo MQTT*

El protocolo de transporte de telemetría de cola de mensajes (MQTT) es de código abierto que se desarrolló y optimizó para dispositivos restringidos y redes de bajo ancho de banda, alta latencia o poco confiables. Es un transporte de mensajería de publicación/suscripción que es extremadamente ligero e ideal para conectar dispositivos pequeños a redes con ancho de banda mínimo. Ha sido desde entonces utilizado por empresas e industrias para las comunicaciones entre máquinas o sensores. (Noguera Arnaldos, 2016)

2.8.4. *Protocolo AMQP*

Es otro protocolo que publica y suscribe, proviene del sector de servicios financieros. Su presencia la tiene en TIC (Tecnología de la información y comunicación). El beneficio mayor de AMQP es su modelo robusto de comunicaciones que soporta transacciones. AMQP es que puede garantizar transacciones completas, lo cual es útil. AMQP se agrupa a menudo con protocolos IoT, pero es bastante limitada en la industria (Pérez Leones, 2019) como lo muestra la Figura 13-1 indica el Modelo de Protocolo AMQP.

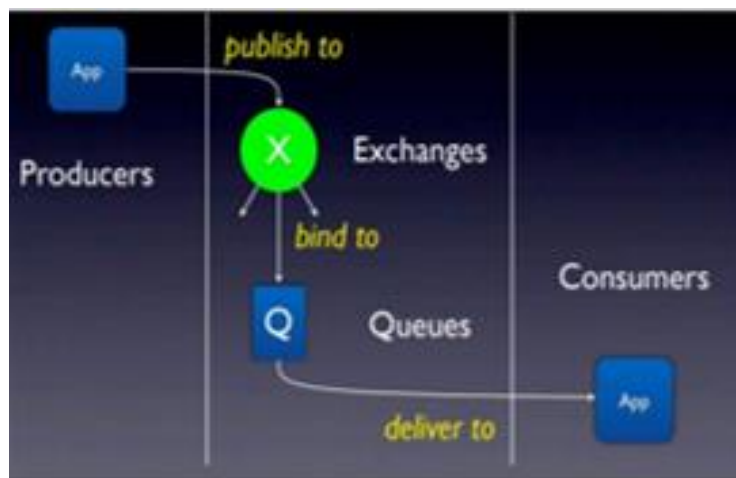


Figura 13-2: Modelo de Protocolo AMQP

Fuente: (Garnock-Jones, 2010, p. 4)

2.8.5. *Protocolo WAMP*

El protocolo WAMP es un subprotocolo de Websockets, que se ejecuta sobre TCP confiable y no implementa mecanismos de confiabilidad propios. Si es necesario, se pueden asegurar usando Websockets sobre TSL/SS: Puede Soportar dos modos de comunicación: Llamadas de Procedimiento Remotas (RPC) y publicador/suscriptor (PUB/SUB). WebSocket no se encuentra diseñado para dispositivos con recursos limitados, sin embargo, está diseñado para la

comunicación casi tiempo real, por lo cual es seguro y con el uso de WMAP puede proporcionar sistemas de mensajería eficientes.

2.8.6. *Protocolo XMPP*

Protocolo extensible de mensajería y presencia es un excelente ejemplo de una tecnología Web existente que encuentra un uso nuevo en el espacio de IoT. El protocolo en general está implementado para una arquitectura cliente-servidor descentralizada, pero con la extensión comentada XMPP para establecer una comunicación directa, de extremo a extremo (también conocida como peer-to-peer (P2P)) y entre los clientes como lo muestra la Figura 14-1 indica la arquitectura XMPP para comunicación entre servidores y clientes. (Castro Cuasapaz, 2018)



Figura 14-2: Arquitectura XMPP para comunicación entre servidores y clientes
Fuente: (Noguera Arnaldos, 2016, p. 39)

2.8.7. *Protocolo STOMP*

EL protocolo STOMP orientado a texto, opera de forma similar a los marcos HTTP, contiene un marco de comando, encabezados opcionales y un cuerpo opcional, también funciona con cadenas de texto tales como JSON o XML, es de código abierto, fácil de implementar y concede interoperabilidad, ya que permite a los usuarios comunicarse con todos los intermediarios STOMP bróker disponibles, sin importar el lenguaje de programación en que están escritos, y además funciona sobre TCP (Carolina et al., 2018, p. 79)

No existen muchas limitaciones en la arquitectura de los servidores y las funciones, respecto al manejo de la semántica y la asignación del nombre del destino, como lo muestra la Figura 15-1 indica la arquitectura del protocolo STOMP.

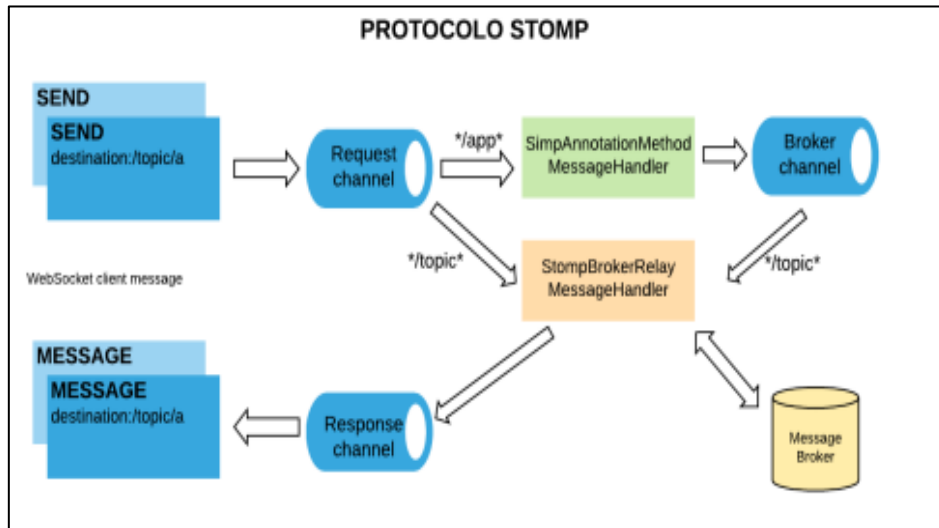


Figura 15-2: Arquitectura del protocolo STOMP
Fuente: (Nayyar, 2017,p79)

2.9. Elementos de IoT

La evolución de IoT desde un punto de vista estructural, no habría podido llevarse a cabo sin la concurrencia de tres áreas concretas, que han permitido el desarrollo continuo de cada una de las aplicaciones desarrolladas para plataformas IoT (Bonilla-Fabela et al., 2016, p. 7).

2.9.1. Hardware

Los dispositivos fabricados en los inicios de la electrónica ocupaban espacios excesivamente grandes para la función que desempeñaban, en la actualidad la tendencia de la miniaturización de componentes electrónicos ha permitido fabricar dispositivos de tamaño reducido con capacidades similares e incluso superiores a la de una computadora personal.

Los sensores, actuadores (dispositivos que controlan los sistemas) y también otros dispositivos de comunicación alojados en los objetos.

2.9.2. Software para aplicación móvil

2.9.2.1. React Native

Es un framework basado en JavaScript que sirve para el desarrollo de aplicaciones móviles de presentación nativa para iOS y Android (Fernández, 2020) Su núcleo se basa en React14, pero usa módulos nativos en lugar de módulos web como bloques de construcción. (Fernández, 2020, p. 5). Usa una técnica que realiza llamadas asincrónicas al sistema operativo donde se ejecuta la aplicación, que llama a las API15 de widget nativas. Hay un motor de JavaScript, y la API de React es parecida a React para diseño de aplicaciones web. Los desarrolladores web pueden construir aplicaciones móviles que se ven como aplicaciones verdaderamente nativas usando una biblioteca de JavaScript. Debido a que el código en su gran mayoría sirve para otras plataformas dando como resultado un desarrollo simultaneo tanto como para iOS como para Android. (Boduch, 2017, p. 4).

2.9.2.2. App Inventor

App Inventor es una aplicación web de Google que permite crear aplicaciones para el sistema operativo de dispositivos móviles Android. Utiliza un editor Drag and Drop (Arrastrar y soltar) para la generación de interfaces gráficas y un sistema de bloques para gestionar el comportamiento de la aplicación. Los proyectos generados a través de esta herramienta se almacenan automáticamente los servidores App Inventor, permitiendo llevar en todo momento un seguimiento y control de todo nuestro trabajo: se observa en la Figura 16-1 la visión global de App Inventor

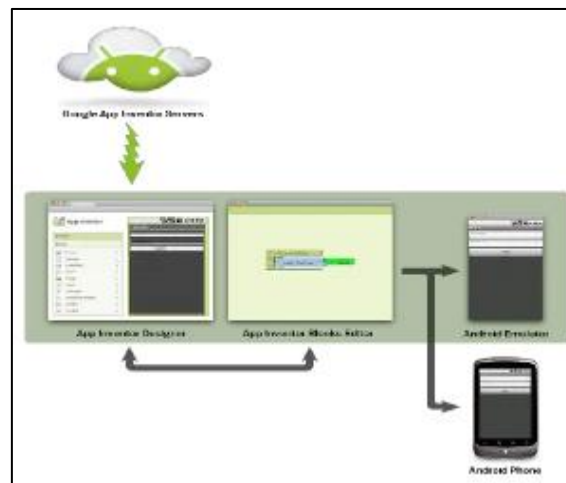


Figura 16-2: Visión Global App Inventor
Fuente: (Creative Commons Atribucion 4.0, 2015, p. 2)

La principal característica de App Inventor no es necesario tener ningún conocimiento de programación para desarrollar las aplicaciones. Simplemente basta con disponer de un navegador web y una cuenta de usuario de Google.

2.9.3. Construcción de una App Inventor

Las aplicaciones construidas mediante App Inventor están compuestas por los elementos que se muestran en el siguiente diagrama

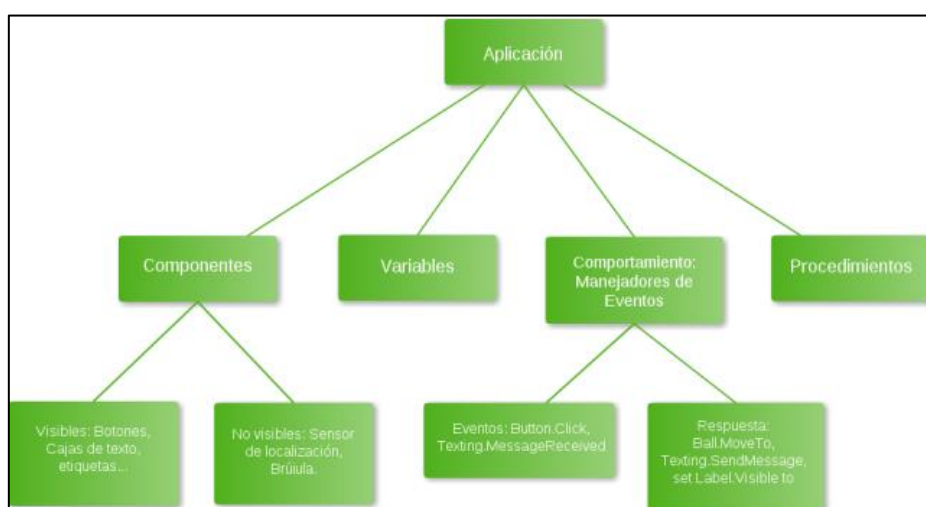


Figura 17-2: Visión Global App Inventor

Fuente:(Creative Commons Atribucion 4.0, 2015, p. 7)

Una buena manera de entender una aplicación, es descomponerla en dos partes, por un lado, los componentes y por otro los comportamientos.

2.9.3.1. Native Script

NativeScript es un framework creado por Telerik para crear aplicaciones multiplataforma nativas para iOS y Android sin vistas web (aplicaciones móviles interpretadas). Permite utilizar Angular11, TypeScript o JavaScript moderno para obtener una interfaz de usuario y rendimiento verdaderamente nativos, al tiempo que reutiliza las habilidades y el código de nuestros proyectos web. (Medio, 2017, p. 6)

La aplicación es compilada, parte del código multiplataforma es traducido a código nativo, mientras que el resto es interpretado en tiempo de ejecución (Delía, L. N, 2018,p 7) como lo muestra la Figura 18-1 Native Script.



Figura 18-2: Native Script
Fuente: (Medio, 2017, p. 6)

2.9.3.2. *Android Studio*

Es el IDE oficial para desarrollar aplicaciones móviles para el Sistema Operativo Android, permite construir aplicaciones para cualquier dispositivo Android sea smartphone, Tablet, televisión, etc. Android Studio utiliza una licencia de software libre Apache 2.0, está programado en Java y es multiplataforma. Incluye un editor de código, un emulador, así como herramientas de testeo, como se muestra en la Figura 19-1 Android Studio (Altamirano Andrade, 2017, p. 9)



Figura 19-2: Android Studio
Fuente: (Quiñones Lopez, 2017, p. 6)

Características

- Interfaz específica para el desarrollo en Android.
- Herramientas Lint para detectar problemas de rendimiento, usabilidad y compatibilidad de versiones.
- Utiliza ProGuard para optimizar y reducir el código del proyecto al exportar a APK.
- Integración con Google Cloud Platform, para el acceso a los diferentes servicios que proporciona Google en la nube. (Altamirano Andrade, 2017, p. 9)

2.9.3.3. Middleware

Un middleware es aquel proporciona una capa de software entre aplicaciones, el sistema operativo y las capas de comunicaciones de la red, que facilita y coordina algún aspecto del procesamiento cooperativo. Desde la perspectiva informática, un middleware proporciona una capa entre software de aplicación y software del sistema. (Neely et al., 2006,p 1009-118)

Requisitos para que un middleware sea compatible con IoT

- **Requisitos del servicio de middleware:** Los requisitos para IoT se pueden clasificar como funcionales y no funcional. Los requisitos funcionales capturan el servicio vicios o funciones (p. ej., abstracciones, gestión de recursos) proporciona requisitos no funcionales (p. ej. Confiabilidad, seguridad y disponibilidad) capturan el soporte de QoS o problemas de desempeño.
- **Descubrimiento de recursos:** Los recursos de IoT incluyen heterogéneos dispositivos de hardware (p. ej., etiquetas RFID, sensores, sensor mote, y smartphones), potencia y memoria de los dispositivos, analógica a dispositivos convertidores digitales (A / D), las comunicaciones módulo disponible en esos dispositivos, y la infraestructura o información a nivel de red (p. ej., topología de red y protocolos), y los servicios prestados por estos dispositivos.
- **Gestión de recursos:** Se espera una QoS aceptable para todas las aplicaciones y en un entorno donde. En arquitecturas de IoT, especialmente en servicios arquitecturas orientadas o basadas en máquinas virtuales (VM), middleware necesita facilitar potencialmente espontáneo recurso (servicio) (re) composición, para satisfacer la aplicación necesidades
- **Gestión de datos:** En IoT, los datos se refieren principalmente a datos detectados o cualquier red información de infraestructura de interés para las aplicaciones. Un middleware de IoT debe proporcionar gestión de datos, servicios a aplicaciones, incluida la adquisición de datos, datos procesamiento (incluido el preprocesamiento) y almacenamiento de datos.
- **Gestión de código:** En particular, asignación de código y código Se requieren servicios de migración. Selección de asignación de código el conjunto de dispositivos o nodos sensores que se utilizarán para realizar una tarea a nivel de usuario o aplicación. (Mohammed, 2016, p. 73)

Plataforma de Middleware

Al momento de seleccionar una plataforma de comunicaciones para la transmisión de información de vídeo bruto se ha tenido en cuenta que las aplicaciones desarrolladas sean distribuidas, de tal manera que el servidor (facilita el servicio) y el/los clientes/-s (consumen ese servicio) se encuentre/-n en equipos distintos y conectados de manera remota. Es el software que permite el intercambio de información entre las aplicaciones, así como las herramientas computacionales que permitan el análisis de datos.

Entre los diferentes tipos de paquetes de software que cumplen en menor o mayor medida estas propiedades pueden incluirse el middleware basado en intermediarios o brokers (ORB, Object Request Broker). Dentro de dichos paquetes de software destacan tres: CORBA (Common Object Request Broker Addressing (García et al., 2000, p. 9), JavaRMI y .NET Remoting.

2.9.4. Aplicaciones de IoT con análisis de la seguridad

De una manera más amplia y diferenciada se van a desglosar cada uno de los campos en los que IoT aportará aplicaciones y ayudará en el desarrollo de las actividades de los mismos. Se va a realizar desde el punto de vista de la seguridad sobre los usuarios. (Sola, 2016, p. 34)

2.9.4.1. Sector Aeronáutica

En el sector Aeronáutico como es el de la aviación sus efectos que podría tener con la inclusión de IoT son variadas, tales como reducir los tiempos de viaje, aumentar la seguridad y comodidad de los pasajeros, reducir costes y gastos de las compañías y aeropuertos, hacer más eficiente el viaje y abaratar costes.

2.9.4.2. Sector Automoción

En el sector de la aviación, va a ser uno de los que más se beneficiarán de las nuevas tecnologías, tal que millones de usuarios que utilizan los automóviles, trenes y el transporte público, en general. Varias empresas importantes del mundo de la automoción desarrollan autos inteligentes, otras siguen trabajando en coches automáticos sin conductor.

2.9.4.3. Sector Telecomunicaciones

Lo más importante de IoT está principalmente en su definición: Interconexión y comunicación.

El sector de la comunicación, observamos que las empresas operadores de telecomunicaciones van a tener un gran protagonismo con respecto al uso de IoT ya que en su poder están todas las infraestructuras que recogen los datos que tienen las capacidades comunicativas.

El servicio primordial que deberán cumplir todas las operadoras será el de proporcionar la conectividad necesaria para que los dispositivos puedan conectarse a internet

2.9.4.4. Sector Edificios Inteligentes

Es el sector que más rápido ha crecido con respecto a esta tecnología y que es sin duda una realidad presente. Las casas se convertirán en un sistema inteligente capaz de adaptarse al usuario y a su forma de vida, de modo que el usuario dejara de prestar atención a las tareas domésticas, hacer la comida o apagar las luces cuando vaya a dormir. La seguridad también tendrá un factor importante ya que no funcionaría como no estaría previsto, también produciría algún accidente doméstico (Sola, 2016, p. 34).

2.9.4.5. Sector Salud

Tendrá IoT muchas aplicaciones en el sector de la salud. La posibilidad que nos ofrece el de poder usar el teléfono móvil con sensores que nos permitirá poder medir las variables corporales y estar conectado con nuestro médico en todo momento. Realizar un diagnóstico rápido de un paciente, prevención de enfermedades o monitorear accidentes son las funciones principales que se esperan en el sector

2.9.4.6. Sector Farmacéutico

El Sector Farmacéutico va de la mano, con el sector de la salud y por lo que va a ser afectado por esta tecnología. En el desarrollo de la industria farmacéutica la tecnología de IoT está todavía llegando, el potencial que se puede prever es enorme, sobre todo en la transmisión de datos de pacientes y seguimiento de medicamentos. Esta tecnología va a cambiar la manera de recoger los datos clínicos durante todo el proceso.

2.9.5. IOE

El aumento del poder de procesamiento y la independencia energética, y a medida que más personas y nuevos tipos de información estén conectados, rápidamente ingresaremos a Internet of Everything (IoE) (Evans, 2012). Los conceptos que se desarrolló la empresa Cisco para definir

una nueva evolución del IoT donde se reúne a las personas, procesos, datos y cosas para que la conexiones en la red sean mucho más valiosas, transformando la información en acciones que creen nuevas capacidades, experiencias más ricas, así como oportunidades económicas únicas a las empresas, individuos y países.

Actualmente las personas se conectan a internet por medio de dispositivos como notebook, Tablet o smartphone mediante redes sociales (Facebook, Twitter, LinkedIn, etc.). IoE transformará la manera de interactuar, como lo muestra la Figura 20-1 indica el Internet of Everything (IOE).

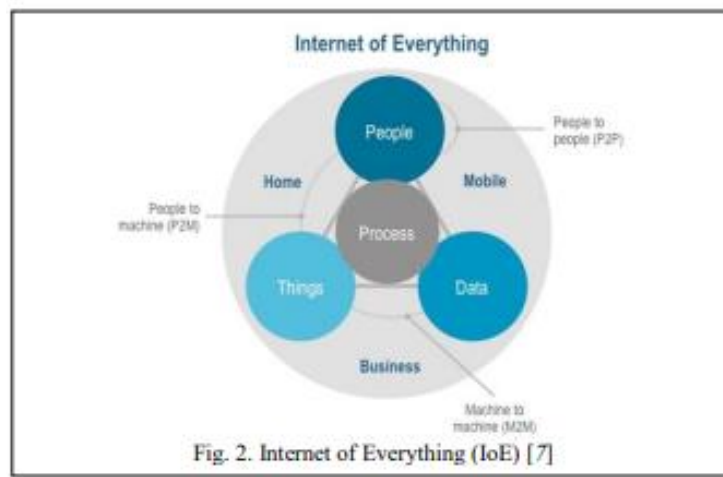


Figura 20-2: Internet of Everything (IOE)
Fuente: (Evans, 2012, p. 1)

2.9.6. IONT

El concepto de IOE se está ampliando al máximo por la implementación del Internet de las Nano Cosas (IoNT). Este ha logrado mediante la incorporación de nano-sensores en diversos objetos y mediante el uso de nano-redes. La aplicación médica es aquello que proporciona acceso a datos de lugares que son previamente imposibles de detectar o de instrumentos inaccesibles debido al tamaño del sensor, (Sasitharan Balasubramaniam, 2013, p. 62-68), La interfaz de IONT con micro dispositivos existentes es importante para que se convierta en omnipresente y un mayor estudio debe centrarse en esta tarea especialmente en el ámbito industrial, biomédico y áreas industriales (Akyildiz y Jornet, 2010, p. 58-63), como lo muestra la Figura 21-1 indica The Internet of Nano Things.

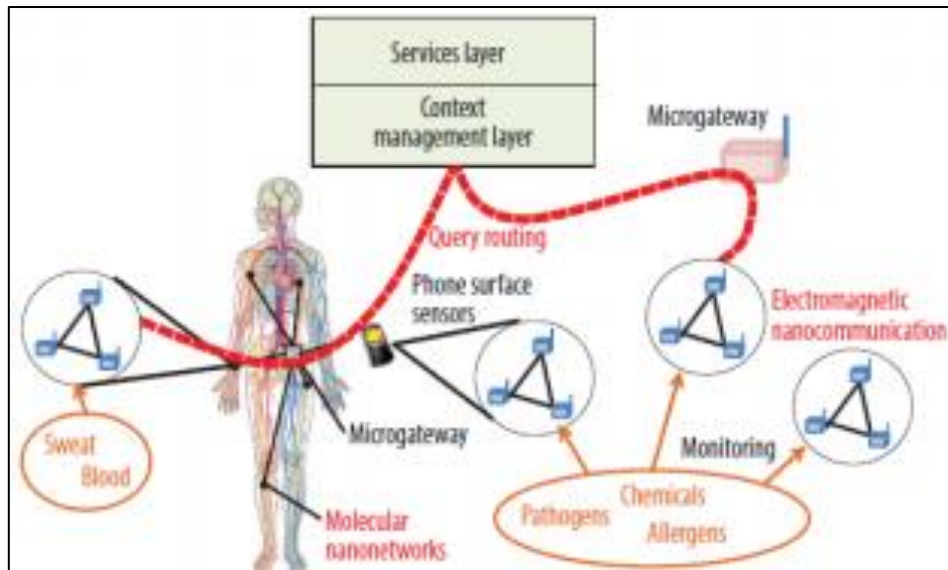


Figura 21-2: The Internet of Nano Things
Fuente: (Sasitharan Balasubramaniam, 2013, p. 62-68)

2.9.7. *IoTM o IoT de salud*

Es una serie de dispositivos y aplicaciones interconectados a través de redes informáticas que permiten aprovechar la tecnología IoT acerca de temas médicos por lo que se puede generar soluciones a necesidades que podrían estar surgiendo en el hogar y en el diario, A nivel mundial los sistemas de salud tienen una tendencia a cambiar y es allí donde herramientas como el internet intervienen. Debido a IoMT se hacen más accesibles los servicios de atención, aumentando la cobertura, mejorando los diagnósticos y facilitando la prevención. Existen soluciones se encuentran justificadas en el sector de la salud (Sanmartin Mendoza et al., 2016, p. 5).

2.9.8. *Medicina y Salud*

Fuertemente se verán afectados los médicos y el sector sanitario por la IoT. La avanzada detección de dispositivos permite la monitorización en tiempo real de parámetros médicos y funciones vitales (por Ejemplo, temperatura, presión sanguínea segura, la frecuencia cardíaca, el nivel de colesterol). La recopilación de datos se transmite a través de tecnologías de comunicación estándar o específicas y personal médico dispuesto para el diagnóstico y de control de la salud de los pacientes.

Observamos en los últimos años que los wearables (usables), tales como relojes inteligentes y pulseras, se han vuelto cada vez más populares y la industria sanitaria ha apreciado en ellos un activo importante para optimizar el tratamiento del paciente y la productividad de los hospitales. Como ayuda, para los pacientes, los relojes inteligentes (Smart watches) proporcionaran la

capacidad de controlar desde sus constantes vitales hasta la calidad del sueño, enviando recordatorios para tomar la medicación o incluso permitiéndoles reportar síntomas sobre alguna dolencia. (Elizalde, 2020, p. 4).

2.9.9. Telemedicina

Con el avance de las tecnologías de la comunicación que ha impulsado a los sistemas de salud a implementar sistemas de intercambio de información médica. Las tecnologías de la comunicación con el constante impulso han permitido ir más lejos que el objetivo inicial de la telemedicina, facultando a los galenos ayudar a colegas a distancia en toma de decisiones, además de brindar formaciones y capacitaciones. Estas características permiten a los sistemas de salud a compartir información cuando la distancia es un problema, así mejorando la atención a los pacientes y el ambiente laboral del personal sanitario (Cardier et al., 2016, p. 840-847).

Según la organización mundial de la salud (OMS) la telemedicina se define como “Contribuir servicios de salud, donde la distancia es un factor crítico, por cualquier profesional de la salud, usando tecnologías nuevas de la comunicación para el intercambio válido de información en el diagnóstico, el tratamiento y la prevención de enfermedades o lesiones, investigación y evaluación, y educación continua de los proveedores de salud, con el interés de mejorar la salud de los individuos y sus comunidades”

2.9.9.1. Aplicaciones de la telemedicina

Pueden variar las aplicaciones de la telemedicina según el caso médico que el profesional de la salud este atendiendo, pero estas pueden cubrir la mayor parte de áreas de la salud

Tele-prevención: La Tele-Prevención trata de prever que el paciente sufra algún daño en su salud, además de informar de la existencia de algún riesgo, utilizando las TIC (Tecnologías de la información y comunicación) (Cardier et al., 2016, p. 840-847)

Tele-diagnóstico: Es aquella que ayuda a los galenos en la toma de decisiones en casos médicos, ya sea por teleconferencia o mediante el uso de bases de datos (Cardier et al., 2016, p. 840-847)

Tele-consulta: Este tipo puede ser Tele-junta médica, donde médicos de dos o más instituciones médicas se ponen en contacto para el intercambio de información para un caso en particular. Puede ser de tipo Telepresencia, donde un médico se pone en contacto con un paciente,

permitiéndole discutir síntomas, prescripciones o tratamientos con otro médico, asistiéndolo a distancia (Cardier et al., 2016, p. 840-847)

Tele-emergencias: Utilizan equipos médicos en los cuales se realizan diagnósticos, monitoreando signos vitales con la ayuda de profesionales de la salud. Esta información es compartida con el médico a distancia para la toma de decisiones necesarias en cada caso. (Cardier et al., 2016, p. 840-847)

Tele-monitorización: Son aquellos que se basa en la toma de datos vitales del paciente para el diagnóstico médico, los datos pueden ser radiografías, signos vitales, electrocardiogramas, etc. (Cardier et al., 2016, p. 840-847); Funciona un sistema de telemedicina gracias a la ayuda de las comunicaciones y de la transmisión de la información por este medio, siendo el más común internet. La información del usuario o pacientes es tomada por diferentes métodos, después esta es enviada por el internet hasta el médico que brinda una segunda opinión o un diagnóstico de algún caso médico a un colega o a un paciente de forma directa. (Cardier et al., 2016, p. 841) como lo muestra la Figura 22-1 Estructura general de un sistema de telemedicina.

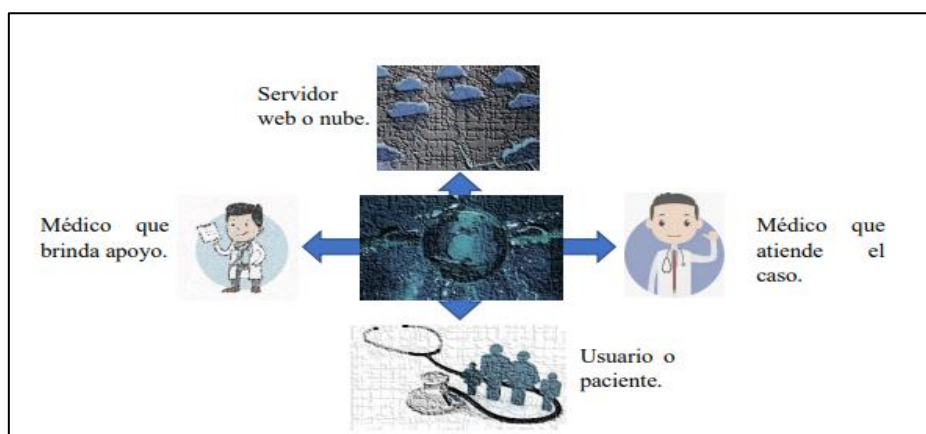


Figura 22-2: Estructura general de un sistema de telemedicina
Fuente: (Cardier et al., 2016, p. 841)

2.9.9.2. Escenarios típicos de aplicación del internet de las cosas en la salud

Servicio de Emergencia Inteligente: Los servicios de emergencia (SE) es muy importante, puesto que muchas vidas dependen de su buena y oportuna gestión; muchas de las veces la responsabilidad recae en el servicio prehospitalario y calidad del servicio médico de emergencia. Existen hasta el momento varias soluciones que ayudan a una oportuna gestión del servicio prehospitalario, a través de las tecnologías inalámbricas y utilización de IoT como base. (Beltrame, 1997)

El principal objetivo de estos sistemas es tener un mejor control en la atención de la salud del paciente, lo cual puede reducir las visitas al hospital y posibilitar el mejoramiento de la calidad de vida (Lin et al., 2007, p 203) como lo muestra la Figura 23-1 Salud centrada en el hogar.

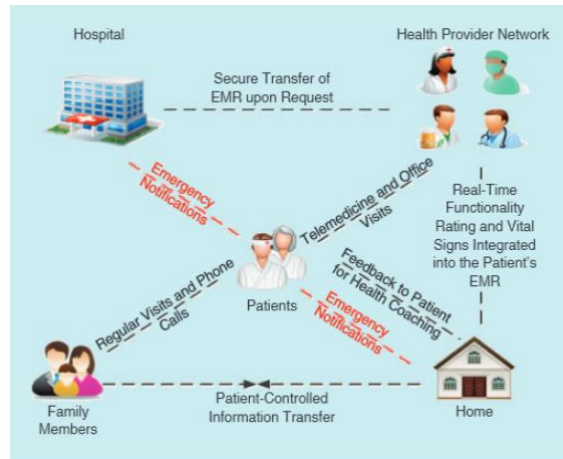


Figura 23-2: Salud centrada en el hogar
Fuente: (Islam et al., 2014, p. 5)

Servicio de Medicación Inteligente: Para cualquier enfermedad tenga éxito depende en gran medida del cumplimiento al tomar la medicina prescrita en el momento adecuado. Los factores asociados al paciente pueden llevar al incumplimiento en la toma de medicina, ya sea por factores psicológicos asociados con la ansiedad, motivación de recuperación, actitud hacia la enfermedad, o el olvido del horario en las personas de la tercera edad, con el objetivo de controlar la medicación del paciente en el hogar (Pang, Tian y Chen, 2014, pp. 352-360).

Dispositivos Biomédicos Portátiles Inteligentes: El desarrollo y la investigación de dispositivos inteligentes portátiles (SWD, Smart Wearable Devices) para servicios personalizados (la salud es un ejemplo de ello) han motivado a los gobiernos y las agencias de investigación de todo el mundo en los últimos años (Cerutti, Magenes y Bonato, 2010, pp. 691-693)

Se propone una arquitectura de red basada en 3G, GSM, infraestructura de Internet y en redes de sensores inalámbricos de área corporal (WBASN, Wireless Body Area Sensor Network), llamada Redes de Médicos Comunitarios, CMN por sus siglas en inglés (Community Medical Network). La arquitectura propuesta por You está dividida en 5 grandes partes: WBASN, dispositivos móviles, redes de comunicación móviles, backbone y servidores en los hospitales locales.

Servicio de Redes Sociales enfocados en la Salud: La aparición de la World Wide Web, el Internet ha sido protagonista de numerosos avances en la sociedad, incluso con la explosión de la llamada Web 2.0, el cual empezó el fenómeno de las redes sociales. Los sitios de medios sociales

(redes sociales), tales como Facebook, Twitter, Youtube, entre otros, se han convertido en poderosas herramientas a través de Internet, tales como aplicaciones de generación de contenido, blogs, wikis, han fortalecido en gran parte la forma de hacer negocios en la Internet, siendo la fuente de ingreso de muchas empresas y familias a nivel mundial (Hwang et al., 2010, p. 3).

2.9.10. Biosensores en el Internet de las Cosas asociadas al sector de la Salud

Dispositivo de análisis está conformado por un elemento biológico de reconocimiento (célula, tejido, receptor, ácido nucleico, enzima, anticuerpo, entre otros), nano materiales (nano partículas, nano compuestos), materiales inteligentes u otros componentes.

2.9.10.1. Capacidad de los Biosensores

Los relojes inteligentes no son el único wearables utilizado por sanitarios, existen también los auriculares, anillos y parches inteligentes; gafas de realidad aumentada (AR) e incluso camisetas inteligentes. Los parches inteligentes capturan los niveles de hidratación, la temperatura corporal y el ritmo cardíaco, entre otros datos biométricos.

La solución de control de la presión sanguínea del mundo para prendas de vestir en forma de auriculares se anunció en el CES de este año. Trata de un hito, que permitirá a médicos y enfermeras medir la tensión de los pacientes de forma continuada y no invasiva

Sería posible la primera aproximación, para distinguir entre biosensores no invasivos e invasivos. Los primeros son los preferidos para el desarrollo de la IoMT y, en particular, para el despliegue de servicios de tele monitorización socio-sanitarios. Sin embargo, los últimos también resultan de interés como, por ejemplo, para los pacientes en las prescripciones farmacológicas (Edge Computing Consortium, 2017, p. 4).

2.9.11. Big Data

Definiciones de Big Data y de Big Data como Servicio (BDaaS): Es un paradigma que hace posible la recopilación, el almacenamiento, la gestión, el análisis y la visualización, potencialmente en condiciones de tiempo real, de grandes conjuntos de datos con características heterogéneas.

BDaaS categoría de servicio en la nube en la que las capacidades que se ponen a disposición del cliente del servicio en la nube permiten recopilar, almacenar, analizar y visualizar los datos utilizando tecnologías Big Data

2.9.11.1. *LAS “V” DEL BIG DATA*

No existía hace muy poco una definición del concepto Big Data y mucho menos de su aplicación al mundo de la salud, se podría explicar el concepto mediante 3 características principales conocidas como las ‘3 vs’ (velocidad, volumen y variedad). (Urueña et al., 2016, p. 14) .

2.9.11.2. *Volumen Masivo de Datos*

La principal característica que define Big Data es su gran cantidad de volumen de información que maneja. Esta cantidad masiva de datos se acumula con un crecimiento exponencial, ampliando continuamente la estructura de almacenamiento de datos existentes., Es importante en este sentido remarcar que los costes asociados a la recogida, almacenamiento y proceso de datos derivados de la proliferación de dispositivos conectados entre sí y a Internet se reducen continuamente. Actualmente cuando se habla de bases de datos masivas se refiere a magnitudes del orden de petabytes (10¹⁵ bytes) o exabytes (10¹⁸ bytes).

2.9.11.3. *Velocidad Elevada en la generación y proceso de información*

Las características esenciales de Big Data es la enorme velocidad en la generación, recogida y proceso de la información. La tecnología Big Data es capaz de almacenar y trabajar en tiempo real con las fuentes generadoras de información como sensores, cámaras de videos, redes sociales, blogs, páginas webs y otras fuentes que generan millones de datos al segundo. La capacidad de análisis por otro lado de dichos datos ha de ser muy veloz reduciendo los tiempos de procesamiento que presentaban las herramientas tradicionales de análisis

2.9.11.4. *Gran variedad en los Datos*

La tercera de las “v” que explica Big Data es la elevada capacidad de agregar información procedente de una amplia variedad de fuentes de información independientes, como redes sociales, sensores, máquinas o personas individuales. Se precisan las nuevas tecnologías para analizar este tipo de datos con el fin de obtener una ventaja competitiva., los sistemas Big Data en este sentido permiten la integración de datos de origen cuantitativo naturalmente desestructurados, así como gráficos, texto, sonido o imágenes.

2.9.12. EDGE COMPUTING

Edge Computing es un conjunto de dispositivos, sensores, recursos informáticos y ordenadores que producen y recogen datos que luego se envían a los centros de nube, sin embargo, EC también puede ser abordado en términos de su arquitectura, retos, tecnologías de software, beneficios y capacidades. como lo muestra la Figura 24-1 Edge Computing.

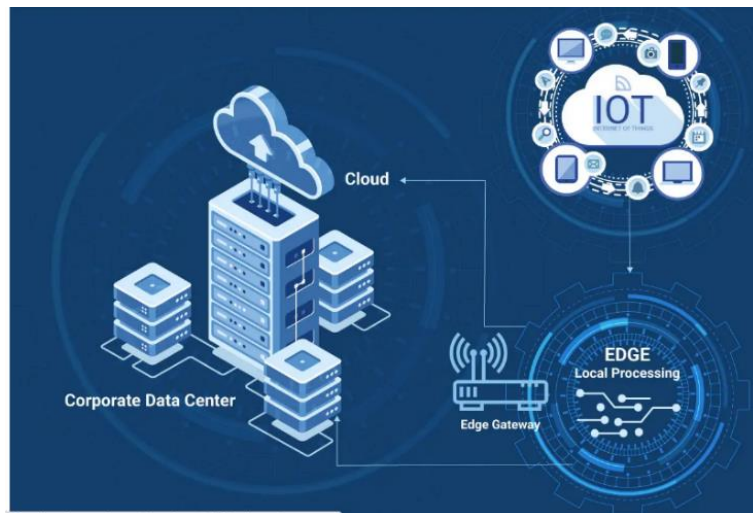


Figura 24-2: Edge Computing
Fuente:(Edge Computing Consortium, 2017)

El beneficio de este tipo de enfoque de computación distribuida ha sido demostrado, por ejemplo, en hogares inteligentes, atención médica, redes de suministro eléctrico inteligentes, ciudades inteligentes, logística inteligente y monitoreo del medio ambiente (Chamoso et al., 2018, p. 2).

Una arquitectura de Edge Computing puede resultar compleja y se enfrenta a grandes desafíos derivados de la heterogeneidad de los dispositivos que la forman. puede verse la comunicación limitada debido a la movilidad de sus componentes, se deben gestionar servicios que pueden residir en parte en dispositivos locales y en parte en la nube, resulta necesario una capa intermedia de red que interactúe con los dispositivos Edge y con la nube, con la flexibilidad necesaria para adaptarse a diferentes tipos de tráfico, seguridad y niveles de servicio requeridos como lo muestra la Figura 25-1 Arquitectura de Ejemplo de Edge Computing

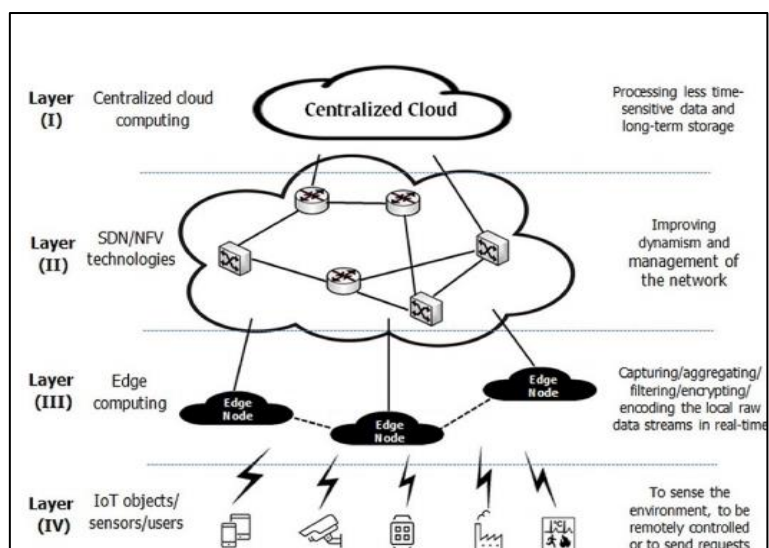


Figura 25-2: Arquitectura de Ejemplo de Edge Computing
Fuente: (Tahezizadeh et al., 2018)

2.9.13. CLOUD COMPUTING

Cloud Computing describe aplicaciones que son extendidas para ser accesibles a través de Internet. Cloud usan grandes servidores para el almacenamiento de datos y poderosos servidores que alojan aplicaciones y servicios Web. (Mestas , 2015, p. 67)

Cloud Computing representa un cambio importante en cómo pueden las empresas y Organismos Públicos procesar la información y gestionar las áreas TIC; apreciándose que con la gestión TIC tradicional las empresas realizan cuantiosas inversiones en recursos, incluyendo hardware, software, centros de procesamiento de datos, redes, personal, seguridad, etc.; mientras que con los modelos de soluciones en la nube se elimina la necesidad de grandes inversiones y costes fijos, transformando a los proveedores en empresas de servicios que ofrecen de forma flexible e instantánea la capacidad de computación bajo demanda (Avanade, 2011, p. 5)

2.9.13.1. Proveedor de Servicios Cloud Computing

Los proveedores de servicio de Cloud Computing son grandes empresas que ponen a disposición una solución en base a servicios de adopción y consumo de recursos, los cuales proporcionan diversos servicios para aplicaciones, infraestructura y plataformas, de tal manera que sea escalable y accesible para las diferentes empresas y negocios. Actualmente existen muchos proveedores de servicios, en base al cuadrante de Gartner para el año 2020, los proveedores que se encuentran liderando este cuadrante son: AWS, Azure de Microsoft, Google Cloud plataforma, así también, compiten por liderar el mercado Alibaba Cloud, IBM, Oracle, Nube Tencent como lo muestra la Figura 26-1 Cuadrante mágico para servicios de plataforma e infraestructura en la nube.



Figura 26-2: Cuadrante mágico para servicios de plataforma e infraestructura en la nube
Fuente: (Raj Bala, y otros, 2020,p 2)

2.9.13.2. Nube de Alibaba

Alibaba Cloud (conocido también como Aliyun en chino) siendo un jugador de nicho en este Cuadrante Mágico. En la evaluación del Cuadrante Mágico se centra en el negocio internacional de Alibaba Cloud: Es el líder en participación de mercado de CIPS en China, donde tiene una alta satisfacción del cliente y fuertes vínculos con el sector público chino.

Aprovecha el dominio de su empresa matriz en las operaciones de comercio electrónico para ayudar a las empresas tradicionales a crear nuevos canales digitales y comenzar a obtener ingresos digitales. Tencent como lo muestra la Figura 27-1 Alibaba Group



Figura 27-2: Alibaba Group
Fuente: (Avanade, 2011,p5)

2.9.13.3. *Servicios web de Amazon*

Amazon Web Services (AWS), una subsidiaria de Amazon, es líder en este Cuadrante Mágico Continúa teniendo un liderazgo dominante en muchas de las dimensiones críticas del mercado de CIPS, incluida la participación de mercado total y las capacidades de las ofertas. (Raj Bala, y otros, 2020,p 5): Es una plataforma que domina en el mundo de las TI. Entre sus principales beneficios para elegir AWS son los siguiente: flexibilidad rigurosa, aplicaciones rápidas y sencillas, escalabilidad de la solución, mantenimiento de datos seguros y alcance masivo de las operaciones (Rossemery, 2019, p. 33)

2.9.13.4. *Google Cloud*

Google es líder en este Cuadrante Mágico: Conjunto de servicios públicos alojados en la nube ofrecidos por la empresa Google. Esta plataforma incluye una gran variedad de servicios para el cálculo, almacenamiento y desarrollo de aplicativos a ejecutarse dentro de hardware de Google: Los desarrolladores de software, administradores de la nube pública y demás profesionales de TI de la organización pueden acceder a dichos servicios ofrecidos por Google Cloud Platform a través de una conexión a internet o red dedicada. (Raj Bala, y otros, 2020,p 6) como lo muestra la Figura 28-1 Google Cloud



Figura 28-2: Google Cloud
Fuente: (López Pachay, 2018,p 82)

2.9.13.5. *IBM*

IBM es un jugador de nicho en este Cuadrante Mágico: IBM ha realizado la transición a la mensajería en torno a la informática híbrida y multinube. El cual ofrece capacidades y herramientas a las empresas, lo que permite elegir el entorno de nube que se adapte a sus requisitos de aplicación específicos, posicionando a IBM Cloud como una oferta especializada o de nicho durante el proceso de selección.

2.9.13.6. *Microsoft Azure*

Microsoft es líder en este Cuadrante Mágico: Ofrece un conjunto completo de soluciones integrales relacionadas con una amplia gama de cargas de trabajo y aplicaciones. Es una plataforma de servicios de Microsoft en una nube pública. Se podría decir, que es cualquiera con una conexión a Internet puede entrar al portal 8 de Microsoft Azure y disponer de los servicios ofertados para crear sus aplicaciones: También una plataforma en la nube que, integrada con Visual Studio Team Services 9 (VSTS), permite realizar Application Lifecycle Management (ALM), administrar el ciclo de vida de nuestra aplicación (Alejandro, 2016,p 22) como lo muestra la Figura 29-1 Servicios IaaS, PaaS de Azure

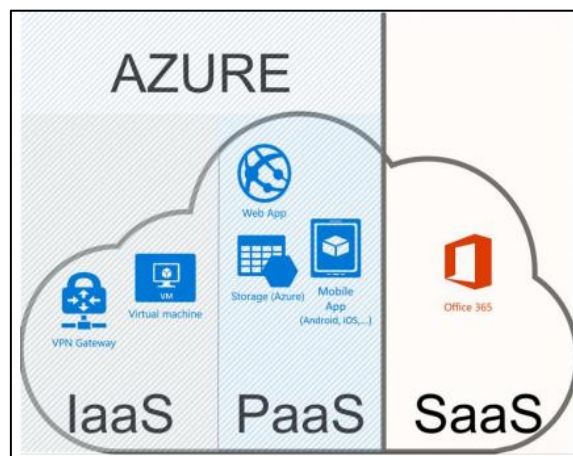


Figura 29-2: Servicios IaaS, PaaS de Azure
Fuente: (L. Bass et al., 2012,p 8)

2.9.13.7. *Oráculo*

Oracle es un jugador de nicho en este Cuadrante Mágico: Demostrando su compromiso en el mercado con una presencia mundial significativamente mayor durante el año pasado en términos de regiones OCI con una única zona de disponibilidad y nuevas capacidades PaaS. Expandingo sus servicios en la nube a regiones como el Medio Oriente, donde un número significativo de clientes ya son clientes de otros productos de Oracle. En particular, OCI es la única plataforma

en la nube a hiperescala en este Cuadrante Mágico. Ofrece todas las capacidades simultáneamente en todas las regiones del mundo, a diferencia de sus competidores. (Raj Bala, y otros, 2020,p 6)

2.9.14. Tipos de redes inalámbricas

2.9.14.1. Redes de área corporal

Los dispositivos inalámbricos se encuentran integrados por sensores es posible que tengan aplicaciones para adquirir y monitorear signos vitales en cualquier lugar y en cualquier momento; hoy en día los sensores tienen poca participación en áreas como la salud debido a los diferentes desafíos que representa desarrollar una red con unidades en miniatura y alta vida útil que operen en el interior o en la proximidad del cuerpo humano. Por ello surgen las Redes de Área Corporal (BAN) que constituyen una nueva generación de las Redes Inalámbricas de Sensores (WSN), adecuadas para la monitorización del cuerpo humano.

WPAN: Enfocan su uso en la optimización de aplicaciones industriales, tal es el caso de ZigBee para aplicaciones de redes de sensores y Wi-Fi para la red de datos con rangos de transmisión más altos; por lo tanto, los sistemas WBAN deben basarse en un estándar diferente, desarrollado y optimizado específicamente para una red de nodos de baja potencia, de corto alcance, alrededor del cuerpo humano, como lo muestra la Figura 30-1 Estándares Inalámbricos.

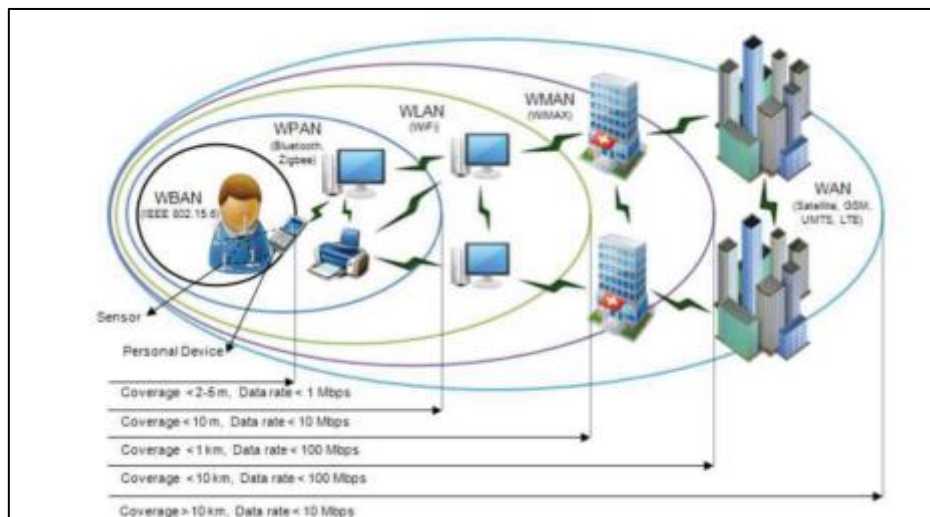


Figura 30-2: Estándares Inalámbricos
Fuente: (Averos , 2017,p 19)

1.9.14.1. Redes de área local

Las redes de área Local son redes de propiedad privada los cuales se encuentran en un sólo edificio o campus de pocos kilómetros de longitud. Utilizan ampliamente para conectar computadores personales, estaciones de trabajo impresoras y otros equipos de red, los cuales compartirán recursos e información: se deberán considerar tres aspectos: tamaño, tecnología de transmisión y topologías.

- Posibilidad de compartir equipos periféricos tales como impresoras, módems, fax, etc.
 - Posibilidad de compartir información a través de bases de datos centralizadas en servidores
 - Reduce y elimina la duplicidad de trabajos
 - Permite mejorar la seguridad y control de la información, como lo muestra la Figura 31-1
- Ejemplo de Red de Área Local

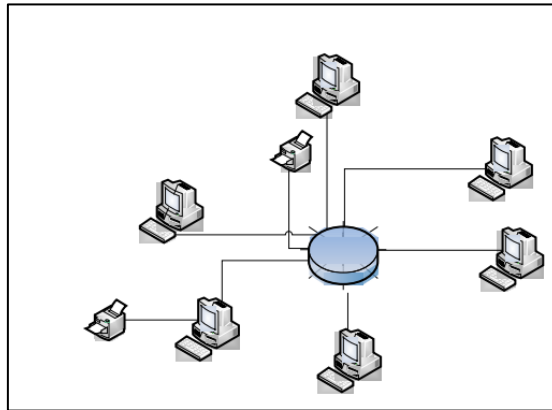


Figura 31-2: Ejemplo de Red de Área Local
Fuente: (López, 2007,p 14)

1.9.14.1. Redes de área Personal

Es una red de computadoras usadas para la comunicación entre dispositivos de la computadora y un usuario. Se pueden conectar con cables hacia los buses de la computadora tales como USB. Una red de área personal inalámbrica (WPAN) puede también realizarse con tecnologías de red como IrDA y Bluetooth: En particular cuando se usa el medio inalámbrico, se refiere a las PANs inalámbricas. La WPAN forma una burbuja alrededor del usuario, que se le denomina ESPACIO DE OPERACIÓN PERSONAL (POS). como lo muestra la Figura 32-1 Redes de Personal.



Figura 32-2: Redes de Área Personal
Fuente: (Garzón Carrión, 2018,p. 16)

1.9.15. Tecnologías Inalámbricas

1.9.15.1. WIFI

Tecnología que permite la interconexión de diversos dispositivos electrónicos de manera inalámbrica, SE lo realiza mediante la transmisión de ondas de radio en dos posibles bandas de frecuencia, 2.4 GHz y 5GHz. Existen varios tipos de Wifi, se basa cada uno de ellos en un estándar proporcionado por el IEEE, algunos de esos estándares son IEEE 802.11b, IEEE 802.11g e IEEE 802.11n, trabajan en la banda de 2.4 GHz con una velocidad de hasta 11 Mbit/s, 54 Mbit/s y 300 Mbit/s, respectivamente. (Reinoso, 2008,p 3) como lo muestra la Figura 33-1 Tecnología Wifi.



Figura 33-2: Tecnología Wifi
Fuente: (Shutterstock, 2018,p 1)

Topología de una red

La topología de una red es aquella ubicación lógica o física de los componentes principales de la misma, a continuación, se relacionan las más conocidas: (Rodríguez, J. E., 2016,p 3) como lo muestra la Figura 34-1 Topología punto a punto.



Figura 34-2: Topología punto a punto
Fuente: (Rodríguez, J. E., 2016,p 3)

Topología malla

Está compuesta por nodos, se puede utilizar en redes inalámbricas y tiene la ventaja de no ser dependiente de un nodo para el buen funcionamiento en la red, es decir que en el momento que se caiga un nodo la red sigue funcionando (Rodríguez, J. E., 2016,p 3), como lo muestra la Figura 35-1 Topología Malla.

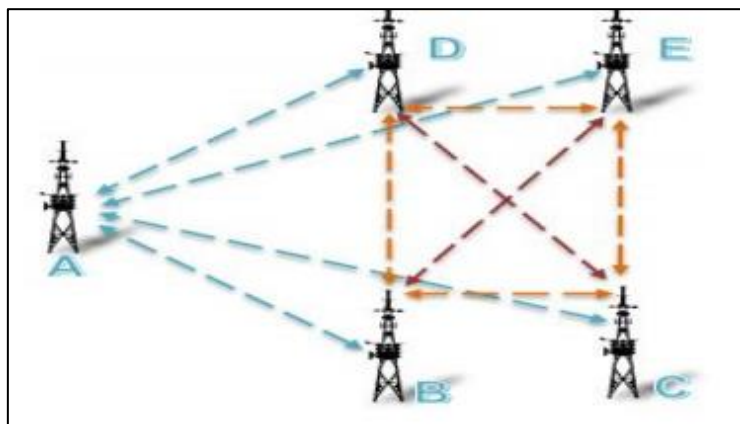


Figura 35-2: Topología Malla
Fuente: (Rodríguez, J. E., 2016,p 3)

Topología punto y multipunto

Topología muy utilizada que permite mantener la comunicación de un punto de un terminal a otro al mismo tiempo en simultáneo. (Rodríguez, J. E., 2016,p 3) como lo muestra la Figura 36-1 Topología Punto y Multipunto.

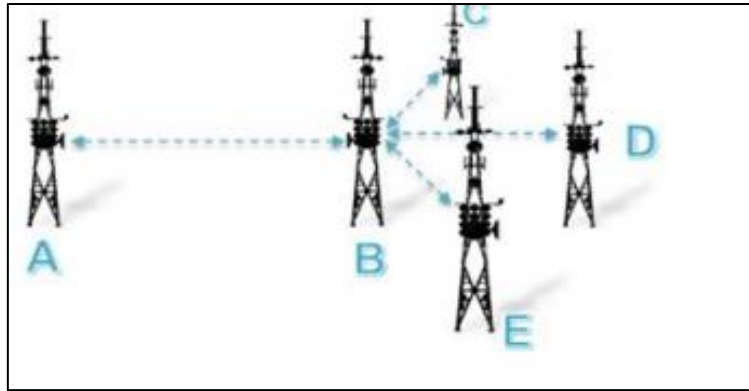


Figura 36-2: Topología Punto y Multipunto

Fuente: (Rodríguez, J. E., 2016,p 3)

1.9.15.2. BLUETOOTH

La tecnología inalámbrica bluetooth se ha convertido en una especificación global de carácter tecnológico para el establecimiento de comunicaciones inalámbricas entre dispositivos portátiles, equipos de escritorio y periféricos. La tecnología bluetooth es de pequeña escala y de bajo costo, opera en la banda ISM de 2.4 GHz, universalmente disponible y que no requiere licencia (Albert et al., 2012,p 41)

Bluetooth se ha diseñado para operar en un ambiente multi usuario. Hasta ocho usuarios o dispositivos pueden formar una piconet y hasta diez piconets pueden coexistir en la misma área de cobertura. Bluetooth puede considerarse como una red inalámbrica de corto alcance muy segura, como lo muestra la Figura 37-1 Tecnología Bluetooth



Figura 37-2: Tecnología Bluetooth

Fuente: (Valverde, 2015,p 52)

A continuación, recogemos las características principales de esta tecnología:

- Frecuencia libre de 2.4 GHz.
- Una red básica de dispositivos conectados por Bluetooth permite como máximo 8 dispositivos conectados a la vez, uno de estos dispositivos debe ser el principal, es decir quien inicia la conexión y los otros los secundarios
- Adicionalmente en estándar actuales se permite la conexión de más dispositivos, pero con otra denominación

- Utiliza la siguiente modulación de frecuencia GFSK.
- Tiene disponibles 79 circuitos disponibles para la conexión
- Se necesita aceptar la conexión desde el otro dispositivo para establecer el contacto.
- No admite cifrado (Valverde, 2015,p. 52).

Evolución de bluetooth

En la Tabla 1-1 representa la Evolución de Bluetooth

Tabla 1-2: Evolución de Bluetooth

Versiones de Bluetooth	Máxima Velocidad de Datos	Características
Bluetooth 1. 0a	732.2kb/s	Primera versión oficial
Bluetooth 1.0b	732.2kb/s	Mejoras Generales
Bluetooth 1.1	732.2kb/s	Problemas de conexión y de seguridad resueltos; primera versión comercializable; cifrado; hasta siete conexiones simultánea
Bluetooth 1.2	1 Mb/s	Compatibilidad descendente con Bluetooth 1.1; menos susceptible a las interferencias gracias al AFH (Adaptative Frequency Hopping)
Bluetooth 2.0+EDR	2.1 Mb/s	Tasas de transmisión de datos tres veces mayores gracias al EDR (Enhanced Data Rate); diversos métodos para el ahorro de energía; uso adicional de NFC (Near Field Communication) para el emparejamiento
Bluetooth 2.1+EDR	2.1 Mb/s	Conexión automática sin PIN gracias al Secure Simple Pairing
Bluetooth 3.0+HS	24 Mb/s	Canal adicional de alta velocidad (HS) basado en Wi-Fi y UWB (banda ultra ancha)
Bluetooth 4.1	25 Mb/s	Los dispositivos más pequeños ya no necesitan intermediarios; IPv6
Bluetooth 4.2	25 Mb/s	Mejoras generales
Bluetooth 5	50 Mb/s	Aumento considerable del alcance y de las tasas de transmisión de datos

Fuente: (Suquilanda, 2013,p 9)

Realizado por: Mora, Marcela, 2022

ARQUITECTURA DE BLUETOOTH

- Bluetooth es una arquitectura de protocolo de capas
- Está conformado por reemplazo de cable y protocolo de control de telefonía (RFCOMM), Protocolos adoptados PPP, TCP/UDP/IP
- Protocolos Centrales
- Radio
- Banda base
- Protocolo de Administrador de enlaces (LMP)
- Protocolo de Control y Adaptación de Enlace Lógico(L2CAP)
- Protocolo de descubrimiento de servicio (SDP) como lo muestra la Figura 38-1 Tecnología Bluetooth

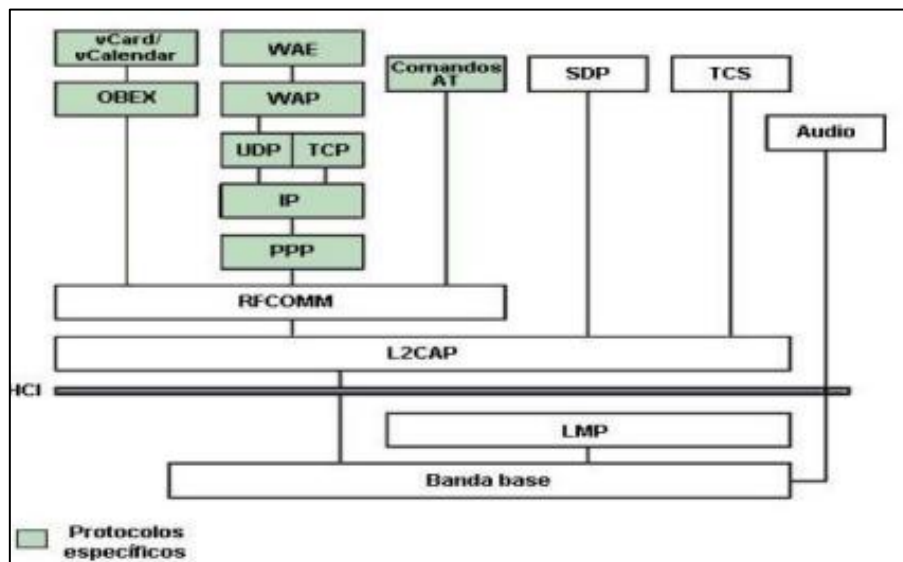


Figura 38-2: Pila de Protocolo de Bluetooth

Fuente: (De León Paredes, 2007,p 2)

PICONETS BLUETOOTH

Los participantes en la piconet pueden intercambiar los papeles es decir si un esclavo quisiera asumir el papel de maestro. Sólo puede haber un maestro en la piconet al mismo tiempo. Cada unidad de la piconet utiliza su identidad maestra y reloj nativo para seguir en el canal de salto. Cuando se establece la conexión, se añade un ajuste de reloj a la propia frecuencia de reloj nativa de la unidad esclava para poder sincronizarse con el reloj nativo del maestro. como lo muestra la Figura 39-1 Ejemplo de una Piconet.

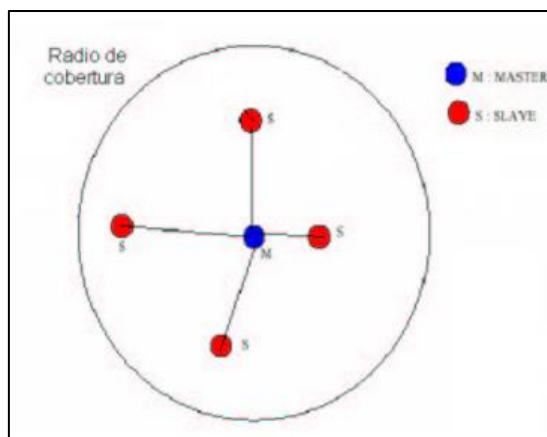


Figura 39-2: Ejemplo de una Piconet
Fuente: (S et al., 2010,p 6)

Las maestras controlan en tráfico del canal, por lo que estas tienen la capacidad para reservar slots en los enlaces SCO. Para los enlaces ACL, se utiliza un esquema de sondeo. A un esclavo sólo se le permite enviar un slot a un maestro cuando éste se ha dirigido por su dirección MAC (medio de control de acceso) en el procedimiento de slot maestro-esclavo.

SCATTERNET BLUETOOTH

Es un dispositivo de una de las dos piconets tiene que pertenecer a dos de éstas. Para esto hay dos formas de hacerlo: ser esclavo de las dos piconets o ser esclavo de una piconet y maestro de la otra. Se encuentran las unidades en el mismo radio de cobertura pueden establecer potencialmente comunicaciones entre ellas. Sin embargo, sólo aquellas unidades que quieran intercambiar información comparten un mismo canal creando la piconet como lo muestra la Figura 40-1

Ejemplo de una Scatternet.

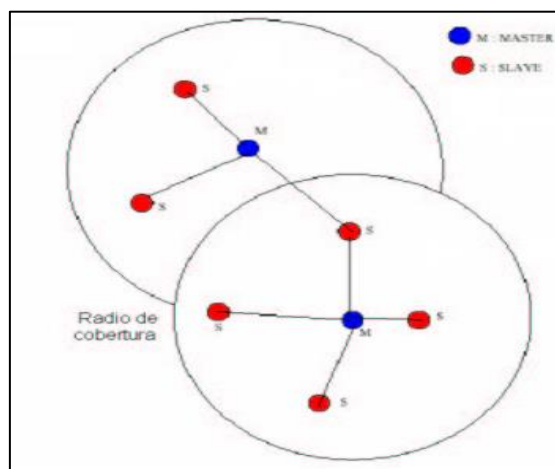


Figura 40-2: Ejemplo de una Scatternet
Fuente: (S et al., 2010,p 6)

CABECERA DE BLUETOOTH

Está constituida por 54 bits contiene información para el reconocimiento de paquete, numeración de paquetes, reordenación de paquetes fuera de orden, el 8 control de flujo, dirección de esclavo y control de errores de la cabecera, repartidos en 6 campos principales como lo muestra la Figura 28-1 Formato de la Cabecera

AM_ADDR: 3 bits de dirección de miembro activo, se utiliza para distinguir los dispositivos activos en una piconet, siendo la dirección 000 la dirección broadcast.

TIPO: 4 bits de tipo de código de paquete, Es aquel que define qué tipo de paquete es enviado y cuántos slots va a ocupar.

FLUJO: 1 bits de control de flujo. El bit de control de flujo es aquel que es usado para notificar al emisor cuándo el buffer del receptor está lleno y que debe dejar de transmitir, en ese caso el bit tendrá el valor 0

ARQN: 1 bit de reconocimiento de paquetes recibidos, paquetes correcto o incorrecto (último paquete recibido). Si es un 1 es un ACK, y con 0 un NAK

SEQN: 1 bit de número de secuencia. El cual se va invirtiendo para evitar retransmisiones en el receptor.

HEC: 8 bits de chequeo de error de cabecera. como lo muestra la Figura 41-1 Formato de la Cabecera

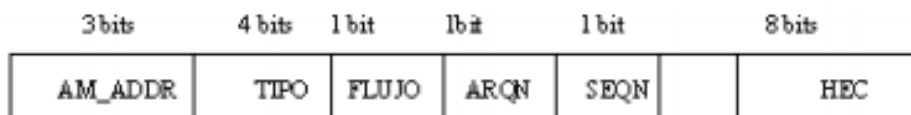


Figura 41-2: Formato de la Cabecera

Fuente: (Salazar, Danilo, 2011,p 8)

Carga útil: Se encuentra formada por bits desde 0 hasta 2745 el cual puede contener campos de voz, campos de datos o ambos. Es aquel que contiene el conjunto de datos que supone la información a transmitir.

CAMPO DE PAQUETES BLUETOOTH

Cada paquete consta de tres partes: el código de acceso, el encabezado, y los datos útiles. como lo muestra la Figura 42-1 Estándar para el formato de Acceso.

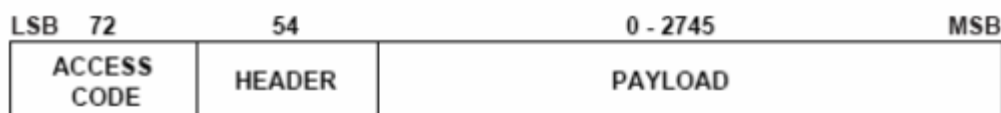


Figura 42-2: Estándar para el formato Acceso

Fuente: (Salazar, 2011,p 8)

Código de acceso: Empieza cada paquete con un código de acceso que puede tener una longitud de 72 bits (en otro caso, de 68 bits).

Existen tres tipos diferentes de código de acceso:

Channel Access Code o código de acceso al canal (CAC): identifica una piconet.

Device Access Code o Código de acceso de dispositivo (DAC): utilizado para procedimientos especiales de señalización como el paging

Inquiry Access Code o Código de Acceso de Búsqueda (IAC): utilizado para procedimientos como inquiry. Se llamará IAC general cuando se quiere descubrir a otras unidades Bluetooth dentro del rango, o IAC dedicado cuando se desea descubrir unidades de un tipo específico. (Acosta Ponce, 2006,p 81)

Cabecera: Contiene información acerca del control del enlace (LC). Para proteger la cabecera de posibles errores en la transmisión, cada bit es repetido 3 veces en fila produciendo así un total de 54 bits. La cabecera consiste de seis campos:

- 3 bits que designan la dirección de miembro activo.
- 4 bits para el tipo de paquete.
- 1 bit para control de flujo.
- 1 bit para indicar al transmisor una transferencia con éxito. 1 bit indicador de número de secuencia. 8 bits para comprobación de errores.

Carga útil (Payload): Puede constar de 0 a 2745 bits y se compone de tres segmentos: cabecera (existente solo en paquetes ACL), Información y CRC (existente solo en paquetes ACL)

Carga útil de voz

- Voz (0-2745 bits)

Carga útil de datos

- Cabecera (8 bits).
- Datos (0 a 2721 bits)
- CRC (16 bits). (Acosta, 2006,p 81)

1.9.16. ZIGBEE

Es un protocolo de comunicaciones inalámbricas similar a Bluetooth, está basado en el estándar para Redes Inalámbricas de Área Personal (WPANS, Wireless Personal Area Networks). Fue diseñado este estándar pensando en la sencillez de la implementación y el bajo consumo, sin perder potencia ni posibilidades. Zigbee amplía el estándar IEEE 802.14.4 aportando una capa de red (NWK, Network Layer) que gestiona las tareas de enrutado y de mantenimiento de los nodos de red (Ortega , 2008, p 66) como lo muestra la Figura 43-1 Tecnología Zigbee

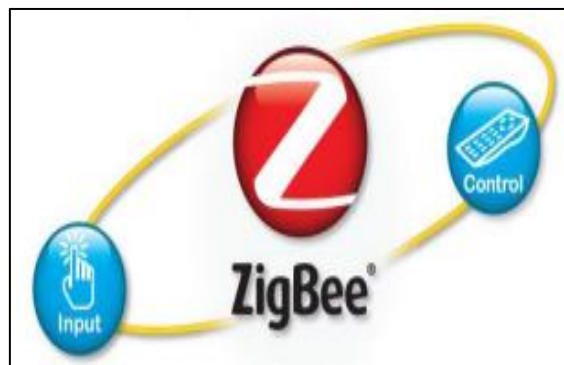


Figura 43-2: Tecnología Zigbee
Fuente: (Valverde, 2015,p 52)

Las comunicaciones Zigbee se realizan en la banda libre de 2.4GHz. A diferencia de Bluetooth no utiliza FHSS (Frequency Hooping), sino que realiza las comunicaciones a través de una única frecuencia, es decir, de un canal. Normalmente puede escogerse un canal de entre 16 posibles. El alcance depende de la potencia de emisión del dispositivo, así como el tipo de antenas utilizadas (cerámicas, dipolos).

El alcance normal con antena dipolo en visión directa suele ser aproximadamente (tomando como ejemplo el caso de Max Stream, en la versión de 1mW de potencia) de 100m y en interiores de unos 30m.

La velocidad de transmisión de datos de una red Zigbee es de hasta 256kbps. Por último, decir que una red Zigbee la pueden formar, teóricamente, hasta 65535 equipos, es decir, el protocolo está preparado para poder controlar en la misma red esta cantidad enorme de dispositivos. La realidad es menor, siendo, de todas formas, de miles de equipos.

Nodos y topología

En una red Zigbee puede haber hasta 254 nodos, no obstante, según la agrupación que se haga, se pueden crear hasta 255 conjuntos/clúster de nodos con lo cual se puede llegar a tener 64770 nodos para lo que existe la posibilidad de utilizar varias topologías de red: en estrella, en malla o en grupos de árboles, como puede observar a continuación:

Topología en estrella: Cada nodo está conectado únicamente al controlador central, Entre su principal ventaja es la facilidad para añadir nuevos dispositivos y, sobre todo, el hecho de que un fallo de un nodo no compromete el funcionamiento del sistema en su totalidad, como lo muestra la Figura 44-1 Conexión en la Topología Estrella

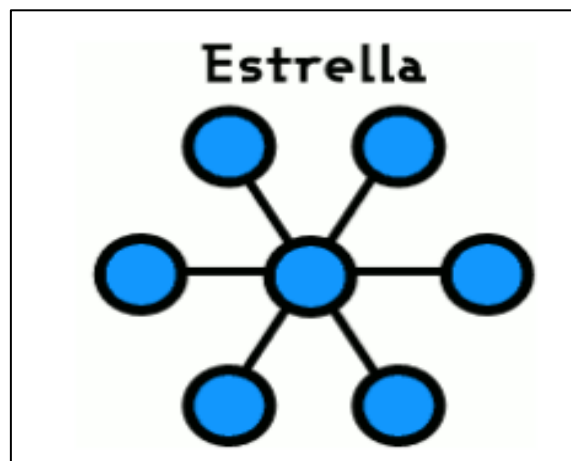


Figura 44-2: Conexión en la Topología Estrella
Fuente: (Sifuentes, 2018.p20)

Topología en árbol: La topología en árbol puede verse como una combinación de varias topologías en estrella. Tanto la de árbol como la de estrella son similares a la de bus cuando el nodo de interconexión trabaja en modo difusión, pues la información se propaga hacia todas las estaciones, como lo muestra la Figura 45-1 Conexión en Topología en Árbol.

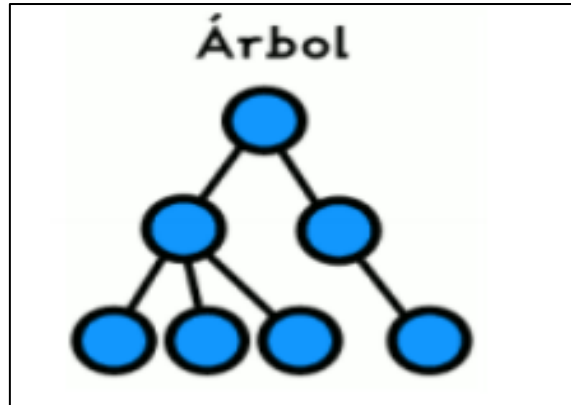


Figura 45-2: Conexión en la Topología en Árbol
Fuente: (Sifuentes, 2018.p20)

1.9.17. NFC

NFC (Near Field Communication o Comunicación de Campo Cercano), es una tecnología de comunicación inalámbrica de corto alcance que permite el intercambio bidireccional de datos entre dispositivos a una distancia corta inferior a 10 centímetros.

Esta tecnología facilita la comunicación en dos direcciones entre dos dispositivos compatibles con la tecnología, de esta manera uno puede ser el emisor y otro el receptor, e inclusive intercambiar roles sin problemas, esto significa que NFC rompe la limitación funcional del RFID, donde tenía que haber un receptor y un emisor y no se podía cambiar los roles, esto hace de NFC una tecnología más versátil y dinámica como lo muestra la Figura 46-1 Tecnologías NFC.



Figura 46-2: Tecnologías NFC
Fuente: (Valverde, 2015.p. 71)

NFC trabaja en la frecuencia de 16,56 MHz, esta frecuencia es abierta y no necesita licencias para ser utilizada y la velocidad de transmisión de datos que ofrece es muy buena oscila entre los 106 Kbps. 212 kbps y 424 Kbps, NFC está pensando para transmitir pequeñas porciones de

información en forma segura, rápida y ahorrando recursos. NFC, puede trabajar en varios modos, entre los que están activo y pasivo, daremos más detalles de estos modos más adelante.

1.9.18. RFID

La RFID (identificación por radiofrecuencia) es una tecnología que se emplea en todas las áreas de captura automática de datos, que permite la identificación sin contacto físico de objetos mediante radiofrecuencia (RF). Entre sus aplicaciones actuales abarcan desde sistemas industriales automatizados, control de acceso, identificación de animales y pasaportes electrónicos hasta aplicaciones médicas, emisión de billetes y seguimiento de existencias (Canestri, 2007,p. 6)

Un primer dispositivo conocido similar a RFID pudo haber sido una herramienta de espionaje inventada por León Theremin para el gobierno soviético en 1945. El dispositivo de Theremin era un dispositivo de escucha secreto pasivo, no una etiqueta de identificación, por lo que esta aplicación es dudosa, (Dargan et al., 2004,p 7)

1.9.19. LORA

Sistema de comunicaciones inalámbricas de largo alcance, promovido por LoRa Alliance. Tiene como objetivo ser utilizable en dispositivos a batería de larga duración, donde el consumo de energía es de suma importancia. puede referirse comúnmente a dos capas distintas: una capa física que utiliza la técnica de modulación de radio CSS (Chirp Spread Spectrum); y un protocolo de capa MAC (LoRaWAN), aunque el sistema de comunicaciones LoRa también implica una arquitectura de red de acceso específica (Augustin et al., 2016, p. 3)

Una de las ventajas de LoRa radica en la capacidad de largo alcance de la tecnología. Una única puerta de enlace o estación base puede cubrir ciudades enteras o cientos de kilómetros cuadrados. El alcance depende en gran medida del entorno u obstrucciones en una ubicación determinada, pero LoRa y LoRaWAN tienen un presupuesto de enlace mayor que cualquier otra tecnología de comunicación estandarizada (Semtech, 2019, p. 3).

1.9.19.1. LORAWAN

Es un protocolo de red de área amplia de baja potencia (LPWA) está diseñado para conectar de forma inalámbrica "cosas" operadas por batería a Internet en redes regionales, nacionales o globales. Incluye funciones que admiten comunicaciones bidireccionales de bajo costo, móviles

y seguras para Internet de las cosas (IoT), máquina a máquina (M2M), ciudad inteligente y aplicaciones industriales (Lora Alliance, 2015, p. 10).

Topología LORAWAN

Define los parámetros de la capa física del dispositivo a la infraestructura (LoRa) y el protocolo (LoRaWAN), Las puertas de enlace están conectadas al servidor de red a través de conexiones IP estándar y actúan como un puente transparente, simplemente convirtiendo paquetes RF en paquetes IP y viceversa. como lo muestra la Figura 47-1. Arquitectura de red LoRaWAN

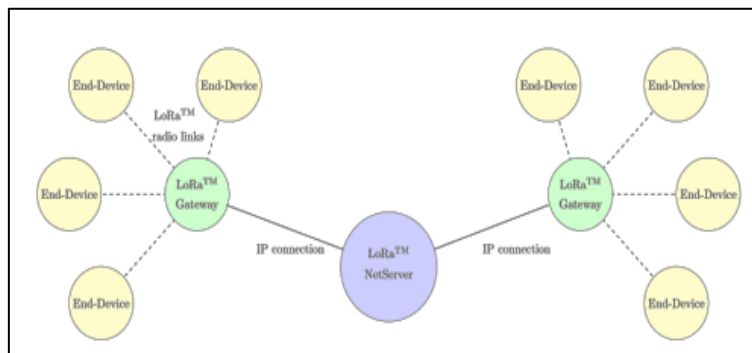


Figura 47-2: Arquitectura de red LoRaWAN
Fuente: (Vangelista et al., 2015, pp. 51-58)

La comunicación entre dispositivos finales y gateways utiliza diferentes frecuencias y canales, utilizando las siguientes reglas:

El dispositivo final cambia de canal de manera pseudoaleatoria en cada transmisión.

Respetar el dispositivo final el máximo ciclo de trabajo de transmisión relativo a la subbanda y a las regulaciones locales. El dispositivo final respeta la máxima duración de transmisión como lo muestra la Figura 48-1. Arquitectura de red LoRaWAN.

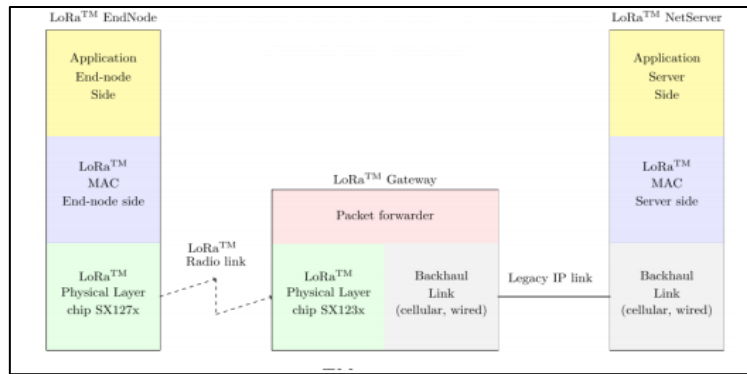


Figura 48-2: Arquitectura de red LoRaWAN

Fuente: (Vangelista et al., 2015, pp. 51-58)

1.9.20. Presión Arterial

Se conoce como presión arterial a la presión de la sangre que circula por las venas, cuando la presión no es adecuada y sobrepasa los valores considerados como normales esta presión va dañando lentamente las paredes de los vasos esto conlleva a un daño colateral en un futuro y padecer de enfermedades cardíacas y vasculares como; infarto de miocardio, insuficiencia cardíaca, angina de pecho, trombosis, insuficiencia renal, etc. (Rodés et al., 2007, p. 40)

La Tensión o Presión Arterial es la fuerza que ejerce la sangre sobre las paredes de los vasos sanguíneos, y se encuentra determinada por dos factores principales: la cantidad de sangre que tenemos y el calibre de los vasos sanguíneos por los que circula. Se usan para medirla dos valores o cifras, el valor más alto es la presión sistólica o Máxima, el menor valor es la presión diastólica o Mínima (Urgiles y Colcha, 2015, pp. 33-36).

Entre los principales instrumentos para medir la presión arterial tenemos: nanómetro de mercurio conocido también como tensiómetro, podemos mencionar el manómetro que es de menor costo pero sus mediciones son idénticas a la del tensiómetro Redes de Sensores Inalámbricas enfocadas a la Medicina cuando existe una diferencia mayor a 4 mm Hg el manómetro debe ser reparado, los tensiómetros electrónicos son mucho más eficientes a la hora de valorar al paciente ya que los mismos trabajan por medio de energía eléctrica o por baterías y por medio de sensores como lo muestra la Figura 49-1 Instrumentos para la toma de la presión Arterial.



Figura 49-2: Instrumentos para la toma de la presión Arterial
Fuente: (Ruiz Sanchez, 2016, p. 66)

1.9.20.1. Presión Arterial Sistólica

Se refiere a la fuerza que se crea por la presión ejercida sobre las arterias cuando el corazón se contrae y bombea la sangre a través de las arterias. La presión arterial sistólica generalmente es considerada normal si está por debajo de 120, en tanto que una presión sistólica mayor que 140 es considerada como presión sanguínea alta o hipertensión. Si una persona tiene una cifra de presión arterial sistólica entre estos dos números, por lo general se considera como límite para la presión arterial alta. Es aconsejable en este caso que se realice una monitorización regular la presión arterial

1.9.20.2. Presión Arterial Diastólica y Sistólica Saludable

En la mayoría de los casos se considera normal que la presión sistólica y diastólica tenga un valor menor de 130/85. Cuando estas cifras son mayores es aconsejable revisar la presión arterial con cierta regularidad, esto se lo puede realizar desde la casa con la ayuda de un medidor de presión arterial o visitando a su médico de confianza para que le ayude con la medición. La presión arterial tiene un valor superior a 140/90 indica que se tiene hipertensión y se debe seguir un tratamiento.

A continuación, se muestra tabla de valores de presión arterial de forma general.

Tabla 2-2: Tabla de Valores de Presión Arterial

Categoría	Sistólica (mm Hg)		Diastólica (mm Hg)
PA Normal	Menor de 120	Y	Menor a 80
PA Elevada	120-129	y/o	Menor a 80
HTA Grado 1	130-139	y/o	80-89
HTA Grado 2	Al menos 140	y/o	Al menos 90

Fuente: (Rosero Caiza, 2018, p. 39)

Realizado por: Mora, Marcela, 2022

1.9.20.3. Método de Medición de la Presión Arterial

Existen algunos métodos para medir la presión arterial que se clasifican en:

Pulso: Consiste en hacer desaparecer y aparecer de nuevo el pulso de una arteria que ha sido colapsada con un manguito de goma

Procedimiento: Se coloca un manguito de goma alrededor de un miembro mientras se observa el pulso en su zona distal (por debajo del manguito) (Wieck et al., 1989, p. 117).

Auscultatorio: Consiste en hacer desaparecer los sonidos del pulso tras colapsar una arteria con un manguito de goma y, posteriormente volverlos a escuchar determinando la tensión arterial máxima y mínima según la naturaleza de dichos sonidos. Es el método más usado y también el más investigado, por tanto, profundizaremos en él más adelante. (Rosero Caiza, 2018, p. 8)

Oscilométrica: Consiste en observar las oscilaciones que se manifiestan en un oscilómetro al colapsar una arteria y posteriormente determinamos la tensión máxima y mínima en función de las características de las oscilaciones

Ultrasónico / Doppler: Este método utiliza el ultrasonido para determinar las presiones máxima y mínima de la sangre a su paso por las arterias. Los valores quedan registrados digitalmente y solamente se utiliza en servicios especializados por su elevado precio (Rosero Caiza, 2018, p. 4)

Intraarterial o Directo: Consiste en medir la presión máxima, mínima y media dinámica de la sangre al paso por la arteria canalizada a través de un transductor que hace transformar dichas presiones en dígitos observables (Rosero Caiza, 2018, p. 72)

La presión máxima y mínima ejerce la sangre a su paso por las arterias, para lo cual se puede utilizar métodos internos y externos. Externos como el manguito de goma e internos como el catéter intraarterial.

La presión arterial pueden ser medida en diferentes partes del cuerpo; por la arteria radial que se encuentra ubicada por encima de la muñeca, la arteria pedía dorsal se la coloca por la mitad de la pantorrilla y en esta zona se encuentra la arteria tibial, por la arteria braquial el dispositivo se lo coloca en la parte superior del brazo, arteria poplitea que se la coloca en la parte superior por encima de la rodilla podemos observar de mejor manera las diferentes formas de colocar correctamente un dispositivo de medición de la presión arterial, cabe recordar que cuando se van a realizar este tipo de exámenes el especialista de la salud recomienda estar en reposo por lo menos 10 minutos antes de realizarse para obtener resultados más reales, el paciente debe estar en estado de relajación, es importante que el al momento de que la bomba empieza a trabajar el paciente no hable ya que esto afectaría los valores, es importante a los valores no redondear, si es necesario se recomienda que se realice dos a tres veces la misma evaluación con el fin de tener un valor mucho más real, las mediciones se las pueden realizar de diferentes posiciones ya sea sentado con el brazo doblado hasta la altura del corazón o acostado para medirlo a nivel de pierna o brazo como lo muestra la Figura 50-1 Puntos Anatómicos donde medir la presión Arterial.

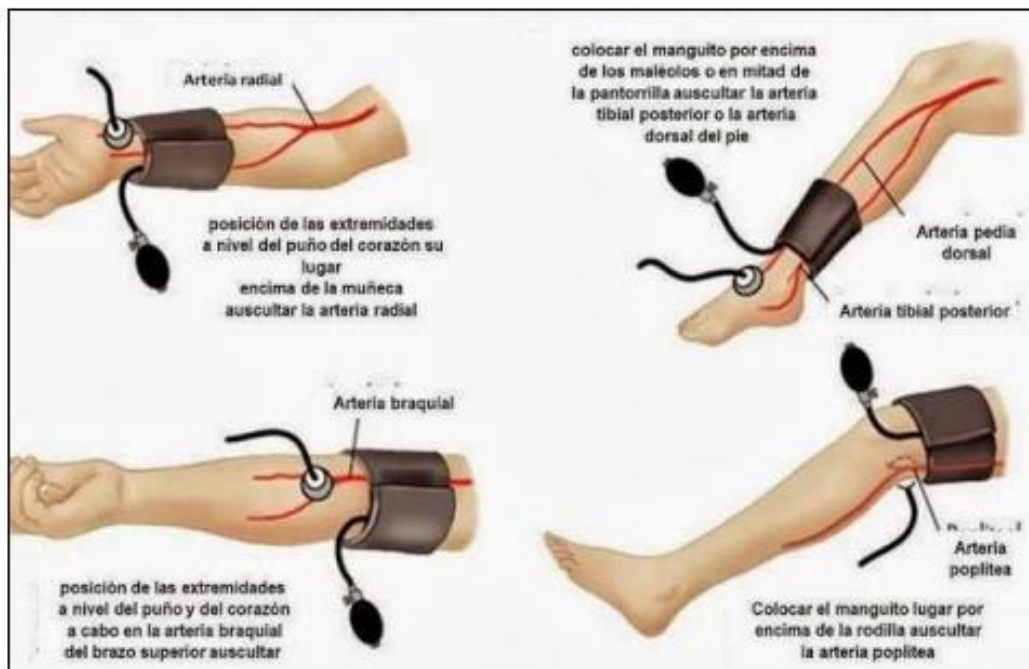


Figura 50-2: Puntos Anatómicos donde medir la presión Arterial

Fuente: (Ruiz Sanchez, 2016, p. 66)

1.9.21. Monitoreo Ambulatorio de Presión Arterial (MAPA)

El monitoreo ambulatorio de la presión arterial (MAPA), es la técnica no invasiva mediante la cual múltiples mediciones de la presión arterial pueden ser obtenidas automáticamente por periodos de 1 a 3 días con intrusión mínima en las actividades del individuo. (Hernández Paz, 1998, p. 6)

Es un procedimiento en el que un dispositivo portátil sobre el paciente registra la PA por lo general en un período de 24 horas, permite el cálculo de la media de PA durante el día, la noche y las 24 horas, así como la PA de carga, que es el porcentaje de mediciones de PA que están en un rango de HTA, en un período de tiempo dado

1.9.21.1. Funcionamiento

El aparato típico utilizado para el MAPA funciona con una batería, y consiste en un mango para el brazo que puede ser programado para inflarse automáticamente durante el período del monitoreo (usualmente 24 a 48 horas)

La técnica oscilométrica consiste en detectar las oscilaciones transmitidas por la arteria braquial al mango; este método mide la presión arterial sistólica y la media, y utiliza diversos algoritmos para calcular la diastólica: Una calibración rigurosa del equipo es necesaria con cada individuo en posiciones de pie, sentado y en decúbito. Para esto, se recomienda hacer mediciones simultáneas de la presión arterial.

1.9.21.2. Interpretación de Reportes

Los datos reportados en el MAPA son la media y la desviación estándar de las presiones arteriales sistólica, diastólica y media de 24 horas, así como el porcentaje de lecturas sistólicas por encima de 140 mm Hg y de lecturas diastólicas por encima de 90 mm Hg (carga de presión arterial), 18 En estudios poblacionales grandes, se han propuesto umbrales de normalidad de la presión ambulatoria de dos maneras: calculándose la presión arterial media más dos desviaciones estándar, y según los percentiles 95

Al conocer los métodos para medir la presión mediante el monitoreo con el prototipo del sistema IoT se describe el tipo de sensor foto electrónico. (Hernández Paz, 1998, p. 7)

1.9.22. Sensor Óptico PPG

Presenta el sensor dos elementos emisores, diodos LEDs (uno infrarrojo y otro rojo). Convierten la energía eléctrica en energía óptica, son compactos, presentan una larga duración (tiempo de vida 105 h) trabajan en un amplio rango de temperaturas con pequeños desplazamientos en la longitud de onda del pico de emisión: También consiste en utilizar un diodo que emite haces de luz infrarroja y un foto-sensor acoplado, ambos tocando la piel humana. La mayor parte de la luz que emite el diodo es absorbida por los tejidos, y sólo un porcentaje del 5 al 10% alcanza los vasos sanguíneos, cuya magnitud de luz reflejada depende de la densidad de hematíes en su interior (Marinel-lo i Roura y Juan Samsó, 2003, p. 5) (Nieto y Vega, 2017)

Funcionamiento del Sensor Óptico FPG

Una fuente de iluminación emite luz hacia, una parte de la luz atraviesa la piel y es absorbida mientras que otra parte se refleja en los tejidos sanguíneos y vuelve al exterior de la piel. La cantidad de sangre varia la cantidad de luz reflejada, un mayor flujo de sangre implica una mayor absorción y por lo tanto menos luz reflejada. Analizando la luz reflejada se puede obtener una señal que contiene la información pulsátil

Técnicas de Captura

Foto pletismógrafo obtiene la señal de las variaciones aplicando la fotopletismografía en los dedos; con este propósito se describen los tres tipos como lo muestra la Figura 51-1 Técnicas de captura en la Fotopletismografía. Las tres técnicas son: Trans-iluminado, reflexión y Fibra Óptica

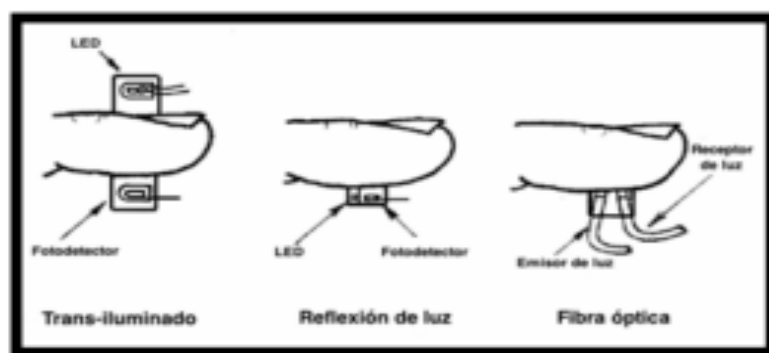


Figura 51-2: Técnicas de Captura en el Fotopletismografía
Fuente: (Galo et al., 2011, p 2)

1.10. Tarjetas de Desarrollo

Son conocidas como placas electrónicas, las Tarjetas de Desarrollo desde su aparición son de gran utilidad y hacen más sencilla la creación de aplicaciones, implementaciones o desarrollo de prototipos brindándole así a la sociedad la capacidad de crear o recrear todo tipo de ideas, también se creó con la finalidad de que sea de uso de fabricantes, educadores, innovadores y estudiantes, en la actualidad se puede encontrar en el mercado una gran diversidad de dispositivos que son compatibles con sensores, módulos y una infinidad de dispositivos electrónicos. Actualmente las tarjetas de desarrollo más destacables son:

1.10.1. Arduino

Es una plataforma de hardware y software de código abierto, basada en una sencilla placa con entradas y salidas, analógicas y digitales, desarrollado en un entorno que está basado en el lenguaje de programación Processing. (AMANGANDI, Jonathan., 2017, p 16): Su principal característica es tener una plataforma de hardware de código abierto. También un software amigable, requiere un nivel mínimo de conocimientos en programación, es similar a C/C++, compatible con Windows, Linux y Mac OS, el circuito interno está compuesto por un micro controlador ATMEGA (Aranda, 2014, p. 148-149) como lo muestra la Figura 52-1 Placa Arduino Mega.

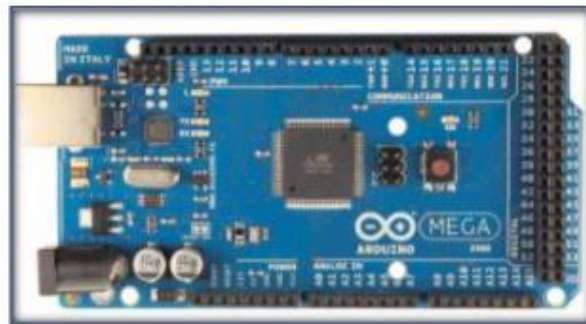


Figura 52-2: Placa Arduino Mega

Fuente: (Aranda, 2014, p. 64)

a) Puerto de serie UART

Es el puerto serie hardware que todos los microcontroladores tienen al menos uno y la comunicación serie es la base de casi cualquiera de las comunicaciones de los microcontroladores (Crespo, E. 2014., 2017, p 2)

1.10.2. NodeMCU Wifi & Bluetooth SoC Module

Creado por Espressif Systems, ESP32 es un sistema de bajo consumo y bajo costo en un chip SoC (System On Chip) con Wi-Fi y modo dual con Bluetooth. En el fondo, hay un microprocesador Tensilica Xtensa LX6 de doble núcleo o de un solo núcleo con una frecuencia de reloj de hasta 240MHz.

ESP32 está altamente integrado con switch de antena, balun para RF, amplificador de potencia, amplificador de recepción con bajo nivel de ruido, filtros y módulos de administración de energía, totalmente integrados dentro del mismo chip: Diseñado para dispositivos móviles; tanto en las aplicaciones de electrónica, y las de IoT (Internet de las cosas), ESP32 logran un consumo de energía ultra bajo a través de funciones de ahorro de energía (Sarabia, 2019 ,p 2), como lo muestra la Figura 53-1 Placa Arduino Mega.



Figura 53-2: ESP 32 NODE MCU
Fuente: (Sarabia, 2019 ,p 2)

CAPÍTULO III

3. MARCO METODOLÓGICO

En el presente capítulo se detallará el tipo de metodologías, las técnicas y el método utilizado para cumplir con los objetivos planteados del prototipo para la monitorización de datos de presión arterial a través de un sensor foto electrónico

Se analizará y se describirá el trasfondo del proyecto planteado, a través de procedimientos específicos en donde abarcará técnicas de observación, recolección de datos, monitoreo y presentación de resultados. Además, se basará su metodología en el desarrollo de las propuestas en cada entorno definido, el cual necesitará estudiar software y hardware que permitan la conexión e interacción de los dispositivos del sistema IoT.

3.1. Metodología de la investigación

Para el desarrollo del presente trabajo se establecen los requerimientos necesarios para llevarlo a cabo. A continuación, se describen los métodos y técnicas de investigación.

3.1.1. *Tipo de investigación*

Se establece la presente investigación como una propuesta tecnológica, se necesitó de un estudio mediante la aplicación de métodos y técnicas de investigación de tipo básica/aplicativa. Además, se eligió ese tipo de investigación ya que centra su idea en aplicar y utilizar los conocimientos adquiridos durante la investigación sobre el problema establecido, siguiendo así una investigación básica. Se busco con el uso de la investigación aplicada mejorar el bienestar de las personas con Presión Arterial y así poder determinar el hardware y software necesario para la implementación del prototipo.

3.1.2. *Método de investigación*

Se decidió emplear para el presente trabajo de investigación la combinación de los métodos teóricos y empíricos, los cuales son:

3.1.2.1. Métodos Teóricos

Revisión de documentación. - Esto con el fin de tener información que se relacione con la enfermedad de Presión Arterial y dispositivos que ayuden al bienestar de cuidadores y pacientes, sobre todo información sobre los avances tecnológicos que existen en la electrónica y en especial en el campo de tarjetas de desarrollo, sensores, módulos para la transmisión y recepción de datos y etiquetas para almacenamiento de datos.

Sistematización. - Para poder aplicar la información recolectada diseño e implementación de un prototipo de sistema IoT para la monitorización de datos de presión arterial a través de un sensor foto electrónico

Análisis y síntesis. - Evaluar los resultados obtenidos y dar conclusiones y recomendaciones sobre el trabajo realizada

3.1.3. Técnicas de investigación

Con la finalidad de compilar la información que será necesaria para la implementación del prototipo para personas con Presión Arterial, se usará las siguientes técnicas de investigación:

- **Documental:** Recopilación de información como investigaciones, libros, publicaciones, estudios similares realizados, los cuales son necesarios para poder definir parámetros, elegir los dispositivos que se deben adaptar al sistema para la implementación, los mismos que deben ser adecuados para el monitoreo de la presión arterial.
- **De campo:** Una vez seleccionado los elementos que forman parte del dispositivo, es necesario verificar que la supervisión, coordenadas y pulso cardíaco pertenece a la persona que lleva el prototipo, controlando que el almacenamiento de datos sea el correcto.

3.2. Desarrollo de la metodología

El proyecto comprende el desarrollo de un sistema de monitoreo local y remoto de la presión arterial de pacientes utilizando la tecnología de IoT (Internet of Things), para ello se diseña y desarrolla una aplicación para Smartphone, la cual permitirá el control y la monitorización de la información clínica de pacientes, además, se desarrolla el dispositivo electrónico biomédico a partir de la tarjeta de desarrollo y el sensor biométrico fotoeléctrico existente en el mercado,

finalmente se realizará la integración y programación de los componentes electrónicos y de comunicación para que, a través de la red se envíen los datos obtenidos como resultado del proceso de toma de información.

Para ello y por ser una metodología de investigación flexible y adaptable al escenario de este trabajo se optó por desarrollar las diferentes fases del proyecto, bajo la metodología de Gestión de proyectos AGILE, la cual es una metodología de gestión de proyectos ampliamente usada en el sector IT (Information Technology) y proyectos de organización empresariales (MANUEL DE JESÚS GÁMEZ LÓPEZ, 2020,p 9), que tiene como principal virtud la flexibilidad y capacidad de modificar el producto a lo largo del proyecto, ya que estos se van usando al mismo tiempo que se desarrollan.

Esta metodología se basa en dividir el proyecto en fases: Inicio, Desarrollo y Cierre, donde el resultado es un producto con una serie de funcionalidades permitiendo su uso.

3.2.1. Matriz operacional de la metodología

Tabla 1-3: Matriz operacional de la metodología

OBJETIVOS	ACTIVIDADES
Realizar la planificación de la investigación.	<ul style="list-style-type: none"> •Investigación y revisión bibliográfica. •Investigación de materiales y componentes necesarios para el desarrollo del proyecto investigativo.
Desarrollar el prototipo del dispositivo electrónico biomédico.	<ul style="list-style-type: none"> •Investigación acerca de posibles proveedores de los componentes electrónicos. • Ensamblado e implementación del prototipo. •Instalación, configuración y uso del IDE para programación de los componentes electrónicos del sistema. •Verificación de funcionalidad.
Desarrollar la aplicación móvil para dispositivos android.	<ul style="list-style-type: none"> •Instalación y configuración del IDE de desarrollo de aplicaciones móviles (App). •Uso del Framework material Design para el diseño de la aplicación •Diseño de un Dashboard para visualización de la información Comunicación y monitorización.
Desarrollar la comunicación para la interconexión e interoperabilidad de las diferentes partes que conforman el sistema IoT	<ul style="list-style-type: none"> • Creación de un web service o API. • Realizar pruebas de funcionalidad en el servidor web. • Configuraciones requeridas para la interoperabilidad de la aplicación, web service y la base de datos.
Diseñar la base de datos del sistema.	<ul style="list-style-type: none"> • Instalación, creación y diseño de la base de datos.
Realizar pruebas de funcionamiento del dispositivo electrónico biomédico, para	<ul style="list-style-type: none"> • Puesta en marcha: pruebas del dispositivo electrónico biomédico en pacientes y supervisado por el especialista médico.

determinar precisión y confiabilidad de las mediciones.	
Resultados esperados.	<ul style="list-style-type: none"> •Aplicación móvil para dispositivos Smartphone Android. •Diseño de la base de datos del sistema. •Web Service o API. •Prototipo electrónico biomédico •Tabla de parámetros médicos estandarizados. •Determinación de los resultados finales.

Realizado por: Mora, Marcela, 2022

A continuación, se detallan las características de los diferentes dispositivos que forman parte del sistema IoT y una descripción de la elección de cada uno.

3.2.2. *Dispositivo electrónico biomédico*

A continuación, se detallan los principales elementos del dispositivo electrónico.

- a. Sensor biomédico de fotopleletismografía
- b. Tarjeta de desarrollo
- c. Módulo de comunicación

3.2.2.1. *Sensor Biomédico*

Se propone el diseño del dispositivo electrónico biomédico con un sensor de fotopleletismografía (FPG) capaz de realizar la toma de muestras para las variables de la presión arterial del paciente. Como se muestra en la Tabla 2-3, se detallan diferentes tipos de sensores biomédicos existentes en el mercado para la medicina tienen las siguientes características (López, 2020, p. 9)

Tabla 2-3: Tipos de Sensores Biomédicos

CARACTERÍSTICAS	MAX32664	MKB0908
Sensor óptico	MAX30102	HR6816 + HR6707
Consumo de energía	< 10Ma	<7MA
Alimentación	3.1V to 5.5V	3.7v-5v
Compatibilidad	Tarjeta Microcontroladora, MAX32630, Arduino.	Arduino, Raspberry, Tarjeta Microcontroladora.
Resolution (bits) (ADC)	19	16
SNR (dB)	102	No disponible
Sectores de uso	Aplicaciones médicas, salud, ejercicio.	Aplicaciones médicas, salud, ejercicio.
Ubicación en el paciente	Muñeca, dedo, oído y otras partes del cuerpo como el brazo.	Muñeca, dedo, otras partes del cuerpo.

Canales de lectura óptica	2 canales que operan simultáneamente	1 canal
Estándar de comunicación	SPI	UART
Interface	I ² C/SPI0	I ² C
Certificación	FDA Certification	No disponible
Calibración	Sí	No disponible

Fuente: (López, 2020,p. 9)

Realizado por: Mora, Marcela, 2022

Tabla 3-3: Alternativas escala de Likert

Alternativas	Valor
Excelente	4
Muy Bueno	3
Bueno	2
Regular	1

Realizado por: Mora, Marcela, 2022

Tabla 4-3: Comparación Likert del Sensor

Pregunta/Sensor	MAX32664	MKB0908.
Sensor óptico	4	3
Consumo de energía	3	2
Alimentación	3	2
Compatibilidad	4	4
Resolution (bits) (ADC)	4	4
SNR (dB)	4	1
Sectores de uso	4	4
Ubicación en el paciente	4	4
Canales de lectura óptica	3	1
Estándar de comunicación	4	3
Interface	4	3
Certificación	4	1
Calibración	4	1
Total	49	33

Realizado por: Mora, Marcela, 2022

Debido a la Tabla 2-3 muestra los Tipos de Sensores Biomédicos, podemos observar que para el trabajo de Investigación se escogió el sensor de MAX32664 ya que cuenta con las características más adecuadas para el desarrollo del prototipo del dispositivo electrónico; cuenta con certificación y calibración de equipos médicos por lo que la toma de muestra será igual a los equipos utilizados en centros de salud.

3.2.3. Análisis de la Tarjeta de Desarrollo

La Tabla 5-3 en la cual se podrá analizar y a su vez realizar una comparativa de las Tarjetas de Desarrollo que ayudarán para el trabajo de titulación: (Aranda, 2014, p. 64), (Bruno, 2019,p. 2), (Maxim, 2019,p. 4)

Tabla 5-3: Análisis comparativo de las tarjetas de desarrollo

CARACTERISTICAS	NodeMCU	Arduino	Microntrolador
Modelo	ESP32	Nano	MAX32630
Memoria RAM	520KB	2KB	512KB SRAM
Memoria Flash	32MB	32KB	2MB
Entorno de Desarrollo	IDE Arduino	IDE Arduino	MAC, Windows, Eclipse, ARMCortexToolchain, IDLE.
Procesador	Tensilica Xtensa LX6 160 MHz (240 MHz overclock.)	ATMega	ARM Cortex-M4F
Puerto U	1	1	1
Voltaje de Operación	2.55 / 3.6 V	5V	1.8V/3.3V
Voltaje de Entrada	3.3V	7-12V	3.3V/5V
E/S Digitales	10	14	4
E/S Analógicos	19	6	4
Dimensión	51 x 23 x 8 mm	4.5 x 1.8cm	4,37 mm × 4,37 mm

Fuente: (Aranda, 2014, p. 64), (Bruno, 2019,p. 2), (Maxim, 2019,p. 4)

Realizado por: Mora, Marcela, 2022

Tabla 6-3: Comparación Likert de la Tarjeta de Desarrollo

Pregunta/Tarjeta de Desarrollo	ESP32	Arduino	MAX32630 MCU
Modelo	4	4	3
Memoria RAM	4	3	4
Memoria Flash	4	4	3
Entorno de Desarrollo	4	4	3
Procesador	4	4	4
Puerto U	4	4	2
Voltaje de Operación	3	4	2
Voltaje de Entrada	4	4	2
E/S Digitales	4	3	2
E/S Analógicos	3	2	3
Dimensión	2	2	2
Total	40	38	30

Realizado por: Mora, Marcela, 2022

Mediante la Tabla 5-3 de la Comparación de Likert la Tarjeta de Desarrollo en su análisis comparativo de las tarjetas de desarrollo, podemos observar que las tarjetas Arduino Nano y

ESP32 son las más adecuadas para el proyecto de investigación. Entre ellas la que mejores prestaciones técnicas a nivel de hardware es la ESP32 pues presenta ventajas en procesamiento, memoria además de ser compatible con el sensor biomédico tiene incorporado el módulo de comunicación Wifi/Bluetooth siendo una característica importante al momento de la elaboración del prototipo aprovechando de mejor manera el espacio físico y evitando adquirir módulos externos.

3.2.4. *Análisis de las Tecnologías Inalámbricas*

En la Tabla 7-3 se puede analizar y a su vez realizar una comparativa de las tecnologías inalámbricas que servirán para el desarrollo del trabajo de titulación (Aranda, 2014, p. 64), (Bruno, 2019,p. 2), (Maxim, 2019,p. 4).

Tabla 7-3: Análisis comparativo de las tecnologías inalámbricas

Características	Wifi	Bluetooth
Aplicaciones	Web, Video, e-mail	Compatibilidad, Transferencia
Alcance	30-100m	10-100m
Ventajas	Capacidad Flexibilidad	Bajo costo, Menor Gasto de Energía Años
Vida de la Batería	Horas	
Arquitectura de Red	Estrella	
Tasa de Transmisión	11-54Mbps	721Kbps
Frecuencia	2.4 y 5 GHz	2.4Ghz (Banda ISM)
Consumo de Energía	Medio	Bajo

Fuente: (Aranda, 2014, p. 64), (Bruno, 2019,p. 2), (Maxim, 2019,p. 4).

Realizado por: Mora, Marcela, 2022

Tabla 8-3: Comparación Likert Tecnologías Inalámbricas

Preguntas/Inalámbrica	Wifi	Bluetooth
Aplicaciones	4	4
Alcance	3	4
Ventajas	3	4
Vida de la Batería	3	3
Arquitectura de Red	4	4
Tasa de Transmisión	3	4
Frecuencia	4	3
Consumo de Energía	3	3

Total	31	31
--------------	----	----

Realizado por: Mora, Marcela, 2022

El resultado de la Tabla 7-3 arrojó que ambas tecnologías podrían ser usadas, por lo cual se vio necesario considerar una tecnología que sea compatible con la tarjeta de desarrollo y que pueda transmitir la información a la plataforma en la nube para su procesamiento y visualización de resultados; en este caso la ESP32 cuenta con la comunicación WIFI/Bluetooth siendo WIFI la opción más adecuada para la comunicación entre el cliente servidor.

3.3. Diseño lógico del sistema

El diseño lógico conlleva toda la parte en la que el sensor logrará enviar la información la cual será almacenada y procesada posterior a ser visualizada.

3.3.1. Sistema operativo

En primera instancia es necesario seleccionar un Sistema operativo para la instalación de los programas que permitirán el correcto funcionamiento del sistema.

Tabla 9-3: Características de los Sistemas Operativos

Características	Windows	Linux
Precio	Según sus versiones depende el valor de su licencia	Gratis
Código Fuente	Copyright	Código Abierto
Estabilidad	Poco Estable	Muy Estable
Seguridad	Muy poco Seguro	Extremadamente Seguro
Difusión	En todo el mercado, salvo el de servidores	Poco extendido en hogares y oficinas, muy extendido en servidores
Precios de los Programas	La mayor parte de los programas son con pago	Existen programas de pago, pero lo más habitual que sean libres
Seguridad	Es necesario adquirir un software adicional de pago	Se configuran políticas de seguridad

Fuente: (Higuerey, 2019, p. 3)

Realizado por: Mora, Marcela, 2022

Tabla 10-3: Comparación Likert Sistema Operativo

Preguntas/Sistema Operativo	Windows	Linux
Precio	4	3
Código Fuente	4	3
Estabilidad	2	3
Seguridad	3	3

Precios de los Programas	2	4
Seguridad	4	4
Total	19	20

Realizado por: Mora, Marcela, 2022

Por medio de la Tabla comparativa 9-3 se seleccionó el Sistema Operativo Linux ya que permitirá la instalación de todos los programas necesarios.

3.3.2. Servidor de alojamiento

Para el diseño lógico del sistema es necesario alojar e instalar los programas necesarios para lograr la recepción, almacenamiento y procesamiento de información. Por tal razón los medios disponibles son: (Medina, 2016, p. 61), (Higuerey, 2019, p. 3)

Tabla 11-3: Características del Servidor de Alojamiento

Características	AWS	Google Cloud	VPS
Costo	Pago por hora o fracción. Descuentos en contrataciones	Pago por minutos. Descuentos proporcionales a las horas de consumo	El costo de acceso remoto es económico
Seguridad	20 certificaciones	6 certificaciones	Proporciona encriptación, autenticación e integridad para el tráfico
Estabilidad	99,98% de disponibilidad mensual.	99,95% de disponibilidad mensual.	Si con mayor rango para la elección
Alternativas de hipervisores	XEN y LXC (Linux Containers)	IDE Arduino	Linux, Windows
Protocolos	MQTT, HTTP	MQTT, HTTP	SSH

Fuente: (Medina, 2016, p. 61), (Higuerey, 2019, p. 3)

Realizado por: Mora, Marcela, 2022

Tabla 12-3: Comparación del Servidor del Servidor de Alojamiento

Preguntas/Servidor de Alojamiento	AWS	Google Cloud	VPS
Costo	4	3	4
Seguridad	4	3	3
Estabilidad	2	3	4
Alternativas de hipervisores	3	3	4
Protocolos	2	2	3
Total	15	14	18

Realizado por: Mora, Marcela, 2022

Por medio de la Tabla comparativa 11-3 para el Servidor de alojamiento se seleccionó, VPS. Servidores Virtuales (Vps) debido que AWS, Google Cloud su instalación simple o ya vienen instaladas las aplicaciones. Son servidores Pagados, Disponible en Linux y Windows, se puede escoger cualquier versión. Acceder remotamente.

Se escogió VPS Pagada por arrendamiento del servidor lo cual requiere de la instalación y configuración de los programas convirtiéndose así en la ganadora por lo general el Sistema operativo Linux ofrece (mejores prestaciones de seguridad) permite la instalación. Cuenta con una IP pública.

Linux: UBUNTU 18.04

- ✓ Hosting (con aplicaciones). Se descarta por que ya cuenta con los programas, fácil de usar.
- ✓ Servidor físico con IP Pública. No fue posible por el tipo de contratación del IP pública.

3.3.3. Servidor Web

Un servidor web se mantiene a la espera de peticiones HTTP por parte de un cliente HTTP que solemos conocer como navegador. El cliente realiza una petición al servidor y éste le responde con el contenido que el cliente solicita. A modo de ejemplo, al teclear www.wikipedia.org en nuestro navegador, éste realiza una petición HTTP al servidor de dicha dirección. (Toledo, 2008, p. 77)

3.3.3.1. Tipos de servidores web

Se puede analizar en la Tabla 13-3 y a su vez realizar una comparativa de los Tipos de Servidores Web que servirán para el desarrollo del trabajo de titulación (Flores, p. 2)

Tabla 13-3: Características de Servidor Web

CARACTERISTICAS	Nginx	Apache 2.0
Tipo de Licencia	Open Source	Open Source
Sistemas Operativos Soportados	Multiplataforma	Windows/Linux
Soportes de Seguridad	SSL y TLS	SSL y TLS
Manejo de Contenido	Dinámico	Estático

Fuente: (Flores, p. 2)

Realizado por: Mora, Marcela, 2022

Tabla 14-3: Comparación de Servidor Web

Preguntas/Servidor WEB	Nginx	Apache 2.0
Tipo de Licencia	4	4

Sistemas Operativos Soportados	3	4
Soportes de Seguridad	2	3
Manejo de Contenido	3	4
Total	12	15

Realizado por: Mora, Marcela, 2022

Se seleccionó en la Tabla 13-3 en la comparación de los Servidores Web, Apache 2.0 ya que es el más usado, además, es un software libre es fácil de usar información en la nube

3.3.4. Base de Datos

En la Tabla 15-3 se puede analizar y a su vez realizar una comparativa de la Base de Datos que servirán para el desarrollo del trabajo de titulación (Garcia, 2015,p 2)

Tabla 15-3: Características de la Base de Datos

CARCTERISTICAS	PostgreSQL	MySQL
Descripción	No es manejado por una empresa y/o persona, sino que es dirigido por una comunidad de desarrolladores que trabajan de forma desinteresada.	Es un sistema de gestión de bases de datos relacional, multihilo y multiusuario, desarrolla MySQL como software libre
Plataforma	Microsoft Windows, MacOS, Linux, Unix	Microsoft Windows, Linux, Unix.
Lenguaje de Programación	PL/PgSQL, C, C++, Java PL/Java Web	C, C++, Pascal, PHP
Herramienta de Case	PowerDesigner	MySQLWorkbench

Fuente: (Garcia, 2015,p 2)

Realizado por: Mora, Marcela, 2022

Tabla 16-3: Comparación de Base de Datos

Preguntas/Base de Datos	PostgreSQL	MySQL
Descripción	4	4
Plataforma	3	4
Lenguaje de Programación	2	3
Herramienta de Case	3	4
Total	12	15

Realizado por: Mora, Marcela, 2022

La Tabla comparativa 15-3 se seleccionó la Base de Datos MySQL siendo Open Source y además es fácil de Implementar.

3.3.5. *Protocolos de Comunicación cliente/servidor para IoT*

En la Tabla 17-3 se puede realizar y analizar una tabla comparativa de los Protocolos de Comunicación cliente/servidor que servirán para el desarrollo del trabajo de titulación (Aron, 2016, p. 33), (Mora et al., 2018,pp70,71)

Tabla 17-3: Características de los Protocolos de Comunicación cliente/servidor para IoT

Características	HTTP	CoAP	OPC UA
Tecnología	Protocolo muy accesible por ser de código abierto	Implementación sencilla	Estándar de comunicaciones libre y abierto
Protocolo	TCP/IP	UDP/multicast	PLC
	Intercambio de información entre los clientes Web (navegadores)	Las peticiones y respuestas se reciben de manera asincrónica	Plataforma neutral que funciona en cualquier sistema operativo

Fuente: (Aron, 2016, p. 33), (Mora et al., 2018,pp70,71)

Realizado por: Mora, Marcela, 2022

Tabla 18-3: Comparación Likert de los Protocolos de Comunicación cliente/servidor para IoT

Pregunta/Protocolo	HTTP	COAP	OPC UA
Tecnología	4	3	2
Protocolo	4	4	3
Seguridad	4	4	4
Total	12	11	9

Realizado por: Mora, Marcela, 2022

En la Tabla 17-3 mediante las características de los Protocolos de Comunicación cliente/servidor para IoT, el protocolo HTTP es efectivo para enviar la información de lecturas del sensor en un intervalo de tiempo en el orden de los kilobytes y pueda iniciar las conexiones con el servidor web mientras conserva la seguridad, el control y la autenticación permitiendo determinar el contenido al que se puede acceder.

3.3.6. *Aplicación Móvil*

En la Tabla 19-3 se puede realizar y analizar una tabla comparativa de Aplicaciones Móviles que servirán para el desarrollo del trabajo de titulación (Aron, 2016, p. 33), (Mora et al., 2018,pp70,71)

Tabla 19-3: Características de Aplicaciones Móviles

Tecnología	React Native	App Inventor
Lenguaje de Programación	Java Script	Java
Licencia de Software	Libre	Libre
Soporte en la Nube	Bloc	Componentes Adaptivos

Precio	Open Source	Open Source
--------	-------------	-------------

Fuente: (Aron, 2016, p. 33), (Mora et al., 2018, pp70,71)

Realizado por: Mora, Marcela, 2022

Tabla 20-3: Comparación Likert de las Aplicaciones Móviles

Pregunta/Protocolo	React Native	App Inventor
Lenguaje de Programación	4	3
Licencia de Software	4	4
Soporte en la Nube	4	4
Precio	2	4
Total	18	19

Realizado por: Mora, Marcela, 2022

En la Tabla 19-3 Mediante las características de las Aplicaciones Móviles, se escogió el App Inventor ya que su entorno de programación visual e intuitiva el cual permite crear aplicaciones totalmente funcionales para dispositivos móviles que trabajen con el sistema operativo Android. Su entorno trabajo bajo la lógica de bloques de códigos (Blockly) lo cual facilita la creación de aplicaciones en pocos minutos a diferencia de los tradicionales lenguajes de programación.

3.4. Diseño de la arquitectura del prototipo IoT

El diseño del sistema IoT contiene los elementos necesarios para envío y almacenamiento de información. El paciente será quién de inicio con el funcionamiento del sistema al momento que activa el Prototipo Biomédico, luego de ello los datos son enviados al servidor privado virtual y almacenados en la base de datos para a continuación mostrar los resultados del monitoreo en la aplicación móvil. En la Figura 1-2 se muestra el esquema del sistema IoT necesario para la medición y monitoreo de presión arterial a pacientes.

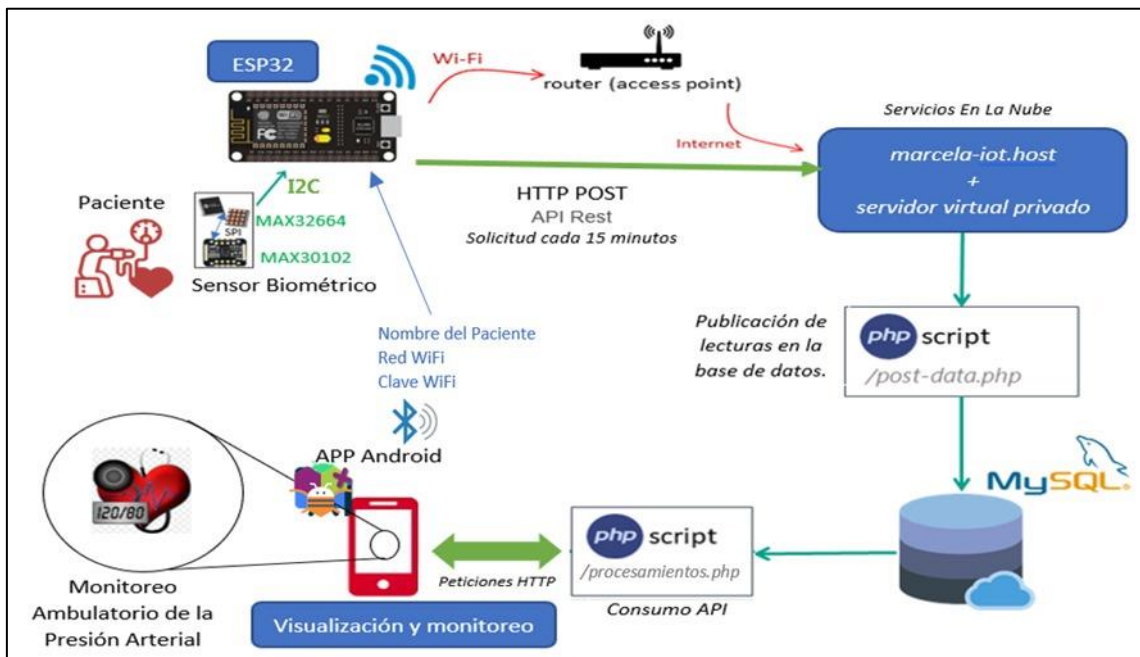


Figura 1-3: Esquema del prototipo del sistema IoT para el monitoreo de la presión arterial
 Realizado por: Mora, Marcela, 2022

3.4.1. Sensor Biomédico

Como primer elemento del sistema, se describe al dispositivo encargado de medir la presión arterial del paciente.

El dispositivo cuenta con la forma y el tamaño de un dedo, es ideal para medir los signos vitales con un orificio de correa de velcro para posicionar el dedo en la posición correcta. El dispositivo integra un sensor biométrico MAX32664D específicamente diseñado para la medición digital de la presión arterial presión Arterial (Presión Sistólica, Presión Diastólica) de potencia ultrabaja y un sensor óptico MAX30102 con elementos de alta sensibilidad.

El MAX35664D interactúa con el sensor óptico MAX30102 a través de un bus de bus de comunicación I²C, tal como se observa en la Figura 2-3.

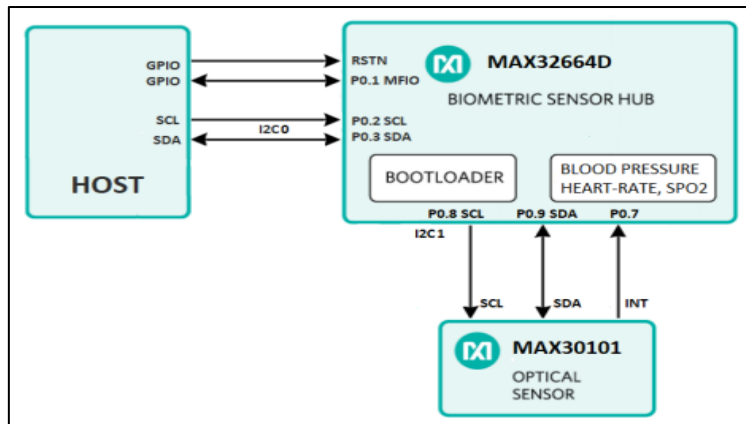


Figura 2-3: Arquitectura del sensor biométrico

Fuente: (Integrated, pp 4)

Normalmente la medición está disponible en una salida del sensor óptico MAX30102 y el concentrador del sensor MAX32664D hace los cálculos de la presión arterial mediante el algoritmo integrado BPT de Maxim (Integrated, pp 4)

Ambos sensores se encuentran instalados en la placa ProtoCentral Pulse Express; ésta placa fue usada para el prototipo electrónico, como se puede observar en la Figura 3-3.



Figura 3-3: Sensor de presión arterial

Fuente: (Protocentral, p. 2)

El dispositivo proporciona datos calculados de presión arterial sistólica y diastólica a un dispositivo host o placa programable a través de su interfaz de comunicación I²C.

Calibración del sensor biométrico.

Se requiere un proceso de calibración para que cada usuario ajuste el algoritmo de estimación al momento de hacer uso del dispositivo electrónico. Este paso generalmente se realiza midiendo en primera instancia la presión arterial sistólica y diastólica del paciente tres veces. El sensor MAX32664D toma los tres valores como referencia y realiza un cálculo de calibración, este

proceso dura aproximadamente 1 minuto. A continuación, se muestra en la Figura 4-3 el algoritmo implementado para la calibración del sensor.

```
58 void loadAlgomodeParameters() {  
59  
60     algomodeInitialiser algoParameters;  
61  
62     algoParameters.calibValSys[0] = 120;  
63     algoParameters.calibValSys[1] = 122;  
64     algoParameters.calibValSys[2] = 125;  
65  
66     algoParameters.calibValDia[0] = 80;  
67     algoParameters.calibValDia[1] = 81;  
68     algoParameters.calibValDia[2] = 82;  
69  
70     MAX32664.loadAlgorithmParameters(&algoParameters);  
71 }  
72
```

Figura 4-3: Líneas de código del algoritmo de calibración del sensor programado en el IDE de Arduino

Realizado por: Mora, Marcela, 2022

Una vez completada la calibración el progreso está al 100% y se genera un vector de calibración que se almacena internamente en la memoria estática de acceso aleatorio (SRAM) para ese paciente, con ello se podrá mantener la exactitud y la precisión de los valores de presión arterial que serán enviados a la placa de desarrollo.

3.4.2. Tarjeta de Desarrollo

Al contar con los elementos electrónicos, se procedió a conectarlos e instalarlos conjuntamente con la placa de desarrollo, para la alimentación del dispositivo se usó una batería de 9 voltios ya que la tarjeta de desarrollo posee un regulador de voltaje en la entrada de voltaje de los pines de 5voltios. El diagrama electrónico diseñado se muestra en la Figura 5-3

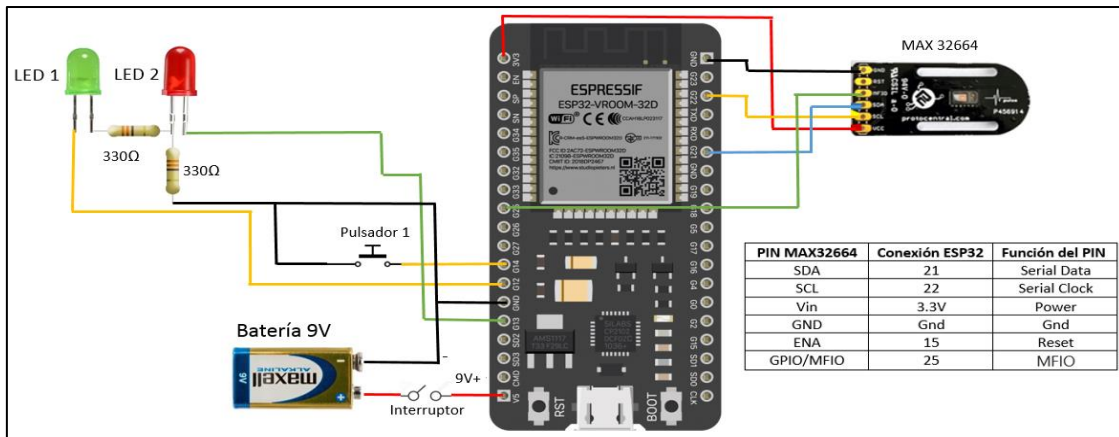


Figura 5-3: Diagrama de conexiones de los componentes electrónicos del prototipo biomédico
Fuente: (options, pp 3)

Como punto importante, el profesional de la salud recomienda al paciente mantenerse en reposo las 24 horas del monitoreo por lo que la manera obligatoria debe mantener en casa; para lo cual la placa programable ESP32 cuenta con la opción de configuración de acceso a la red inalámbrica 802.11 del lugar de residencia del paciente o de algún punto de red con acceso a internet. La tarjeta de desarrollo ESP32 utilizada en el prototipo biomédico fue programada bajo el lenguaje de programación de código abierto IDE de Arduino como se muestra en la Figura 6-3. Con la ayuda de este software se instalaron las librerías y configuraciones necesarias para la toma de información desde el dispositivo electrónico además del envío de información de forma continua a la plataforma IoT

```

PRESION_ARTERIAL_WIFI_APP_BLUETOOTH | Arduino 1.8.10
File Edit Sketch Tools Help
PRESION_ARTERIAL_WIFI_APP_BLUETOOTH $
1 //////////////////////////////////////////////////////////////////////////////////////////////////////////////////
2 // MARCELA MORA- IOT
3 //////////////////////////////////////////////////////////////////////////////////////////////////////////////////
4
5 #ifndef ESP32
6 #include "WiFi.h"
7 #include <HTTPClient.h>
8 #else
9 #include <ESP8266WiFi.h>
10 #include <ESP8266HTTPClient.h>
11 #include <WiFiClient.h>
12 #endif
13
14 #include <Wire.h>
15 #include "max32664.h"
16 #include "BluetoothSerial.h"
17 #define RESET_PIN 04
18 #define MFIO_PIN 02
19 #define RAWDATA_BUFFLEN 250
20
21 BluetoothSerial SerialBT;
22
23 WiFiClient client;
24 unsigned long counterChannelNumber = 1226807; // Canal ID
25 const char * myCounterReadAPIKey = "E1C8JZROJEADDQ01"; // API Key
26 const int FieldNumber1 = 1; // Campo a leer
27
28 const int LED1 = 12;
29 const int LED_BT = 13;
30 const int SW = 14;
31
32 const char* ssid = "FLIAMORA"; //SSID

```

Figura 6-3: Captura de pantalla de una parte del código configurado en la placa ESP32 en el IDE de Arduino.

Realizado por: Mora, Marcela, 2022

Una vez instalados todos sus componentes el prototipo biomédico se encuentra listo para recibir el nombre y datos de la red Wifi del paciente mediante la aplicación móvil por Bluetooth, en la Figura 7-3 se observan las líneas de código utilizadas para recibir la información de los campos en la tarjeta de desarrollo

```
212     while(SerialBT.available())
213     {
214         ssid_pass_paciente = SerialBT.readString();
215         //Serial.write(SerialBT.read());
216         Serial.println(ssid_pass_paciente);
217
218         for (int i = 0; i < ssid_pass_paciente.length(); i++)
219         {
220             if (ssid_pass_paciente.substring(i, i+1) == ",")
221             {
222                 ssid = ssid_pass_paciente.substring(0, i);
223                 int a = ssid.length();
224                 palabra = ssid_pass_paciente.substring(i+1);
225                 for (int j = 0; j < palabra.length(); j++)
226                 {
227                     if (palabra.substring(j, j+1) == ",")
228                     {
229                         pass = palabra.substring(0, j);
230                         paciente = palabra.substring(j+1);
231                         Serial.print("SSID = ");Serial.println(ssid);
232                         Serial.print("Password = ");Serial.println(pass);
233                         Serial.print("Paciente= ");Serial.println(paciente);
234                         delay(2000);
235                         flag = false;
236
237
238
239                     int n1 = ssid.length();
240                     char char_array1[n1 + 1];
241                     strcpy(char_array1, ssid.c_str());
242
243                     int n2 = pass.length();
244                     char char_array2[n2 + 1];
245                     strcpy(char_array2, pass.c_str());
246
247                     int n3 = paciente.length();
248                     char char_array3[n3 + 1];
249                     strcpy(char_array3, paciente.c_str());
```

Figura 7-3: Líneas de código para envío de información a la tarjeta ESP32 por Bluetooth programado en el IDE de Arduino

Realizado por: Mora, Marcela, 2022

Para facilidad de uso y que el usuario pueda operar y configurar la red Wifi, en el prototipo biomédico se instaló un botón pulsador el cual permite configurar la red inalámbrica de 2.4Ghz en la placa de desarrollo al momento que el paciente haya ingresado los datos de su red en la aplicación.

A continuación, al estar enlazado en la red Wifi empezará a calibrarse el sensor y se procede a tomar los datos de presión arterial. En la Figura 8-3 se muestra el código utilizado para la medición.

```

143 //MAX32664.enableInterruptPin();
144 Serial.println("Getting the device ready..");
145 delay(1000);
146
147 //CICLO DE TOMA DE DATOS
148 for(;;){
149 uint8_t num_samples = MAX32664.readSamples();
150
151 if(num_samples){
152
153     Serial.print("sys = ");
154     Serial.print(MAX32664.max32664Output.sys); //Presión sistólica
155     Serial.print(", dia = ");
156     Serial.print(MAX32664.max32664Output.dia); //Presión diastólica
157     Serial.print(", hr = ");
158 }
159

```

Figura 8-3: Líneas de código para toma de presión arterial programado en el IDE de Arduino
Realizado por: Mora, Marcela, 2022

Finalmente realizará el envío de información mediante el Protocolo de Transferencia de HiperTexto (HTTP) el cual articula el intercambio de información de presión arterial del paciente al servidor IoT utilizando el método HTTP POST haciendo uso de la interfaz de programación (API) de transferencia de estado representacional (REST) la solicitud se hará cada quince minutos. En la Figura 9-3 se muestra la configuración realizada para que la tarjeta de desarrollo realice el envío de la información recolectada por el sensor y compruebe la conexión con el servidor

```

//Check WiFi connection status
if(WiFi.status() == WL_CONNECTED){
    WiFiClient client;
    HTTPClient http;
    // Your Domain name with URL path or IP address with path
    http.begin(client, serverName);
    // Specify content-type header
    http.addHeader("Content-Type", "application/x-www-form-urlencoded");

    // HTTP POST respuesta
    String httpRequestData = "api_key=" + apiKeyValue + "&fecha=" + String(paciente) + "&PDIAS=" + String(MAX32664.max32664Output.dia) + "&PSIS=" + String(MAX32664.max32664Output.sys);
    Serial.print("httpRequestData: ");
    Serial.println(httpRequestData);
    // Send HTTP POST request
    int httpResponseCode = http.POST(httpRequestData);
    if (httpResponseCode > 0) {
        Serial.print("HTTP Response code: ");
        Serial.println(httpResponseCode);
    }
    else {
        Serial.print("Error code: ");
        Serial.println(httpResponseCode);
    }
    // Free resources
    http.end();
}
else {
    Serial.println("WiFi Disconnected");
}
delay(10000);
}
}

```

Figura 9-3: Líneas de código para envío de información al servidor por HTTP programado en el IDE de Arduino

Realizado por: Mora, Marcela, 2022

3.4.3. *Plataforma IoT*

La información recolectada debe ser almacenada y procesada además de encontrarse disponible para acceder a ella desde cualquier parte mediante la aplicación móvil, para ello se listan las opciones para un entorno IoT que es posible usar; los métodos más accesibles para construir la plataforma requerida son:

- La utilización de una IP Pública fija con la traducción de dirección de red (NAT) a la IP de la red local donde se encuentra el servidor físico, para ello necesario contratar una IP Pública directamente con el proveedor del acceso a Internet (ISP) y la habilitación de puertos para el servidor web: el puerto 80, usado comúnmente para el protocolo HTTP y si el tráfico web es seguro, se utiliza el puerto 443 para el protocolo HTTPS. Además de contar con los recursos de hardware disponibles para un funcionamiento 24/7. Algo que resulta en cuanto a costos algo caro y tramitación algo demoroso
- Otra opción, es alquilar un servidor en la nube por un tiempo determinado con opción a extender el plan adquirido, contando de esta manera el servicio disponible en cualquier momento de manera más oportuna.

En ambos casos, es necesario realizar las instalaciones y configuraciones respectivas de todo el software tal como: sistema operativo, base de datos y servidor web; de esta manera será posible dar acceso al prototipo electrónico para que aloje la información de presión arterial del paciente, siendo un tipo de nodo de red de comunicación o interfaz expuesta por un comunicante o un canal de comunicación para el equipo cliente.

Por lo cual, la opción más adecuada que se utilizó fue el adquirir un servicio de servidor privado virtual que permita la instalación del sistema operativo ganador y un dominio, el cual fue adquirido mediante el proveedor Hostinger que proporciona ambos servicios por planes de contratación mensual, en la Figura 10-3 se muestra al servidor con la IP 31.170.165.164 ejecutándose en la página del proveedor.

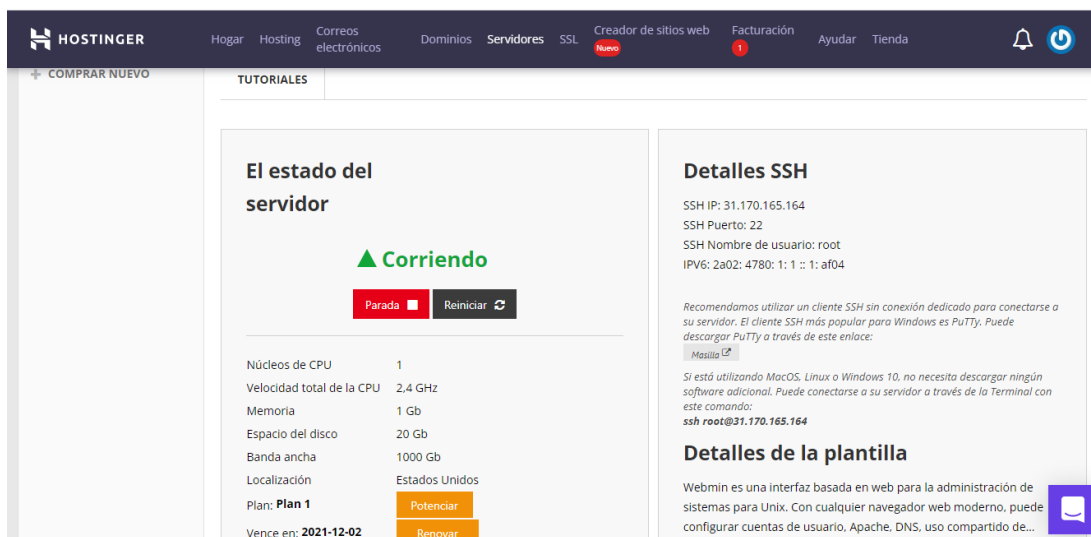


Figura 10-3: Captura de pantalla del servidor privado virtual
Realizado por: Mora, Marcela, 2022

3.4.4. *Instalación del sistema operativo*

Como primera instancia, al ya contar con el servidor es necesario instalar el sistema operativo Ubuntu 18.04 de 64bits en el servidor, para ello accedemos a la sección de opciones y seleccionamos el nombre del sistema operativo; una vez realizado aquello procedemos a colocar una clave de administrador y guardamos los cambios, por consiguiente, levantamos el servidor.

Existen diversas formas para poder almacenar la información y poder hacerla pública, de tal forma sea posible acceder a ella desde cualquier parte y dispositivo, entre las cuales los métodos más accesibles son:

La utilización de una IP Pública fija con la traducción de dirección de red (NAT) a la IP de la red local donde se encuentra el servidor, para ello el contrato de una IP Pública se lo debe gestionar directamente con el proveedor del acceso a Internet (ISP) y la habilitación de puertos para el servidor web: el puerto 80, usado comúnmente para el protocolo HTTP y si el tráfico web es seguro, se utiliza el puerto 443 para el protocolo HTTPS. Además de contar con los recursos de hardware disponibles para un funcionamiento 24/7. Algo que resulta en cuanto a costos algo caro y tramitación algo demoroso: El costo que tiene un Hosting es de 10 dólares mensuales

Otra opción, es alquilar un servidor de alojamiento y un nombre de dominio por un tiempo determinado con opción a extender el plan adquirido, contando de esta manera el servicio disponible en cualquier momento.

En ambos casos, es necesario realizar las configuraciones respectivas de base de datos, programación del servidor web, dar acceso al prototipo electrónico para envío de datos y disponer de un punto final de comunicación, siendo un tipo de nodo de red de comunicación o interfaz expuesta por un comunicante o un canal de comunicación para el equipo cliente donde se monitoreará la presión arterial del paciente. Por lo cual, la opción más adecuada que se utilizó fue el adquirir un servicio de Hosting y dominio el adquirido fue del proveedor Hostinger el cual entrega ambos servicios por planes mensuales, como se muestra en la Figura 11-3

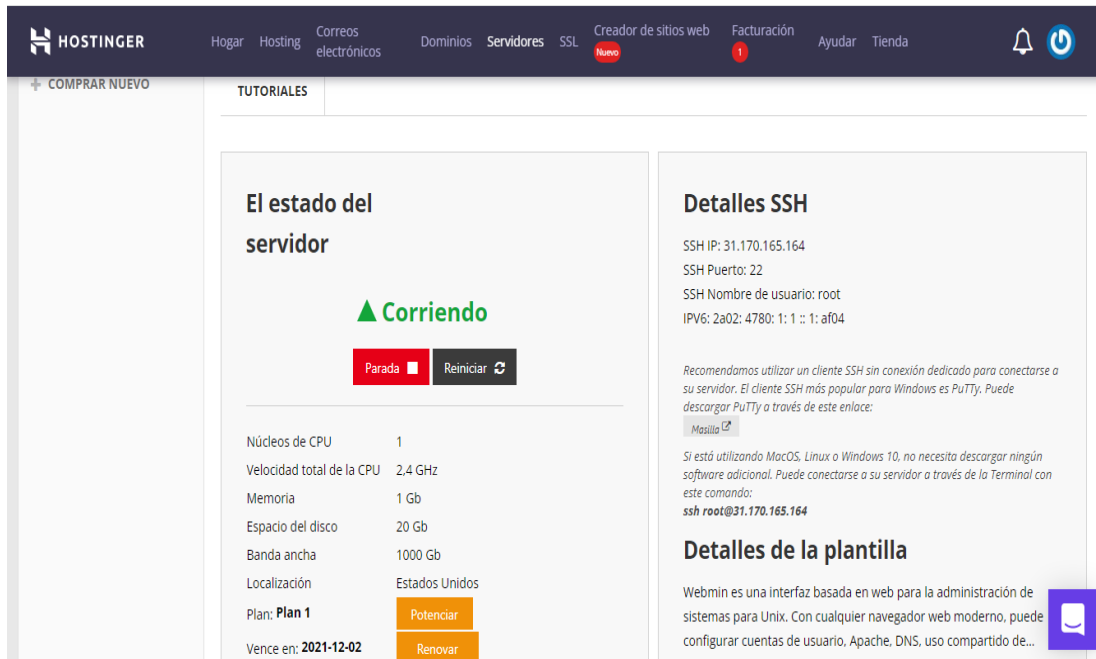


Figura 11-3: Captura de pantalla de Hostinger
Realizado por: Mora, Marcela, 2022

3.4.5. Instalación del sistema operativo

Como primera instancia, al ya contar con el servidor es necesario instalar el sistema operativo Ubuntu 18.04 de 64bits en el servidor, para ello accedemos a la sección de opciones y seleccionamos el nombre del sistema operativo; una vez realizado aquello procedemos a colocar una clave de administrador y guardamos los cambios, por consiguiente, levantamos el servidor.

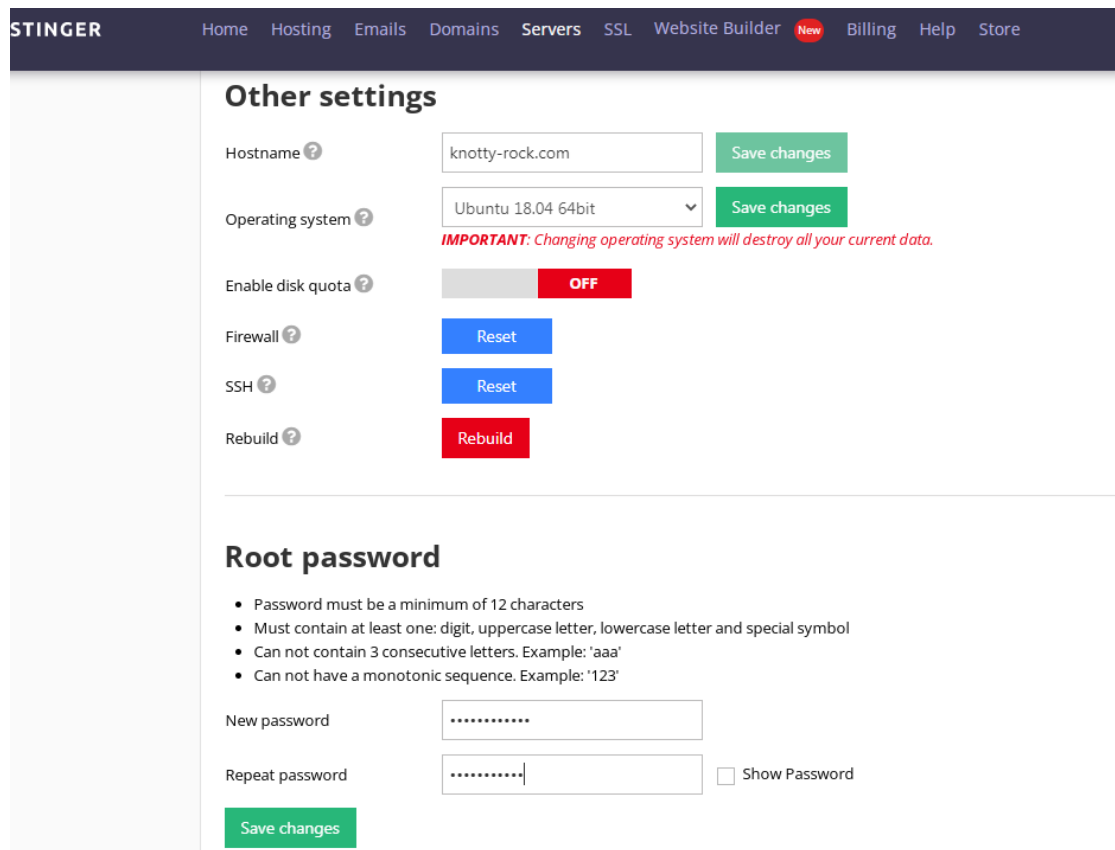


Figura 12-3: Captura de pantalla de las opciones configuradas en el servidor VPS
Realizado por: Mora, Marcela, 2022

Una vez ejecutado el sistema operativo es necesario contar con el acceso al servidor que permita la instalación de todo el software. Para ello es necesario utilizar un protocolo de administración remota, que permita controlar y modificar el servidor remotamente a través de Internet con un mecanismo de autenticación, esto es posible mediante un cliente SSH (Secure Shell); el usado fue el cliente Putty pues es recomendado por el proveedor, tiene licencia libre y es de fácil manejo.

Para poder contar con el acceso al servidor es necesario ingresar la SSH Key proporcionada por el proveedor, la cual se muestra en la Figura 13-3

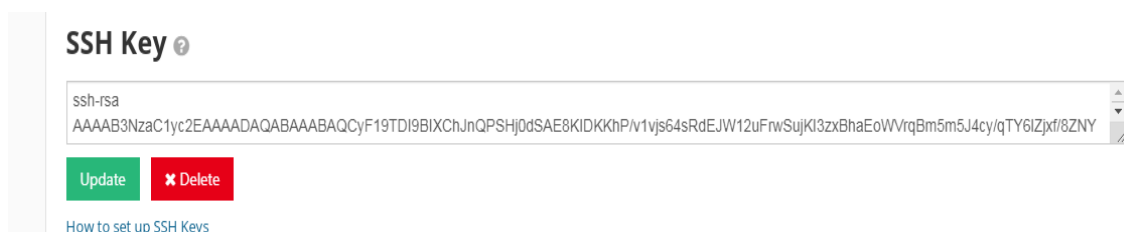


Figura 13-3: SSH usada para la conexión cliente -servidor
Realizado por: Mora, Marcela, 2022

En el cliente Putty procedemos a registrar la llave en el computador cliente para lograr una conexión éxitos al servidor, tal como se observa en la Figura 14-3

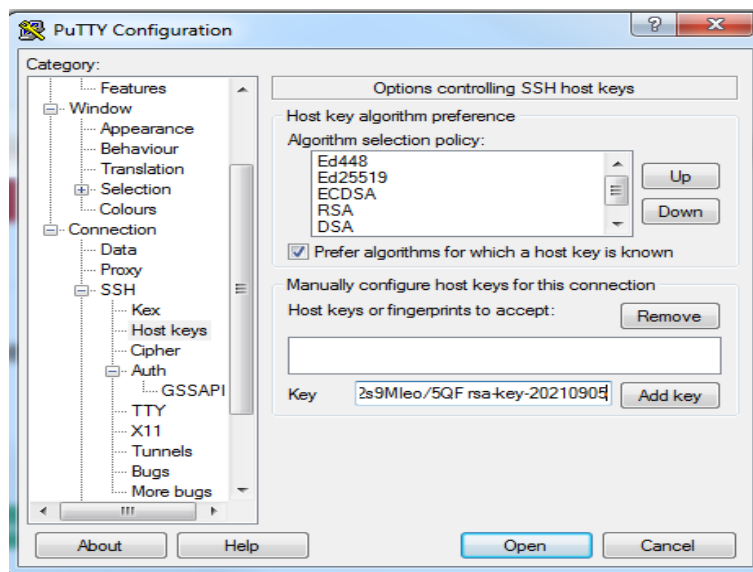


Figura 14-3: Registro de la llave SSH en el cliente
Realizado por: Mora, Marcela, 2022

A continuación, en la opción de sesión se procede a configurar la IP pública 31.170.165.164 en el puerto 22 por SSH. Al abrir la conexión es posible acceder al servidor; para ello se ingresa el usuario y contraseña configurado al momento de instalar el sistema operativo, tal como se observa en la Figura 15-3

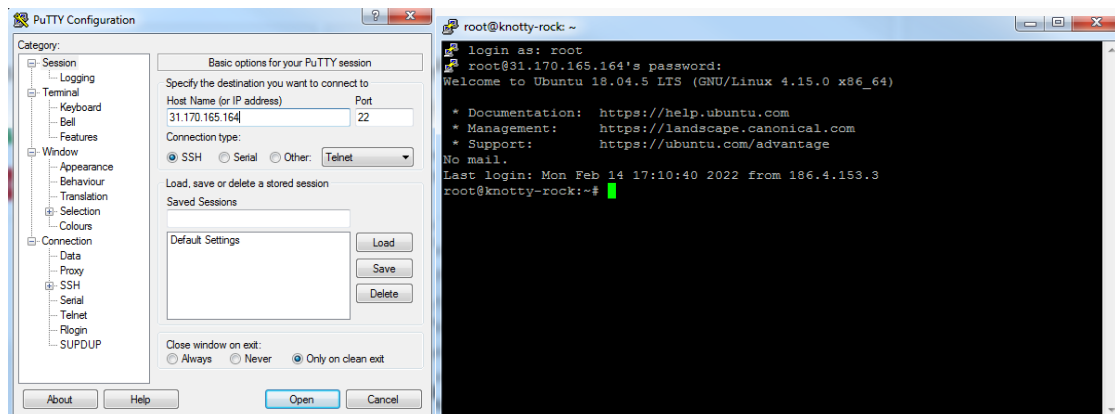


Figura 15-3: Acceso al servidor por SSH
Realizado por: Mora, Marcela, 2022

Una vez realizada la instalación del sistema operativo y contar con acceso mediante SSH se proceder a instalar la base de datos y el servidor web.

3.4.6. *Instalación del servicio web*

El servidor HTTP Apache es el servidor web más usado en el mundo y el usado el desarrollo del sistema IoT. Provee muchas características poderosas, incluyendo módulos de carga dinámica,

soporte robusto a medios, así como amplia integración a otros programas comúnmente utilizados. A continuación, se describen los pasos realizados en la consola de administración de Linux del servidor para su instalación y configuración.

- **Instalar Apache**

Se procede a actualizar el índice de los paquetes locales, para garantizar que se reflejen las cargas más recientes de las nuevas versiones de los paquetes

```
sudo apt-get update
```

Con el siguiente comando se instala el paquete apache2

```
sudo apt-get install apache2
```

- **Configurar el cortafuegos**

Es necesario modificar los ajustes del cortafuegos de tal manera que se garantice el acceso externo a los puertos web por defecto.

Con el siguiente comando se Instala el cortafuegos.

```
sudo apt-get install ufw
```

Con el siguiente comando se Permite el acceso de http y https

```
sudo ufw allow http  
sudo ufw allow https
```

Con el siguiente comando se comprueba el estado y verifica el servicio.

```
sudo systemctl status apache2
```

Al seguir todos estos pasos se procede a comprobar que Apache está operativo ingresando únicamente por la IP pública. Tal como se observa en la Figura 16-3

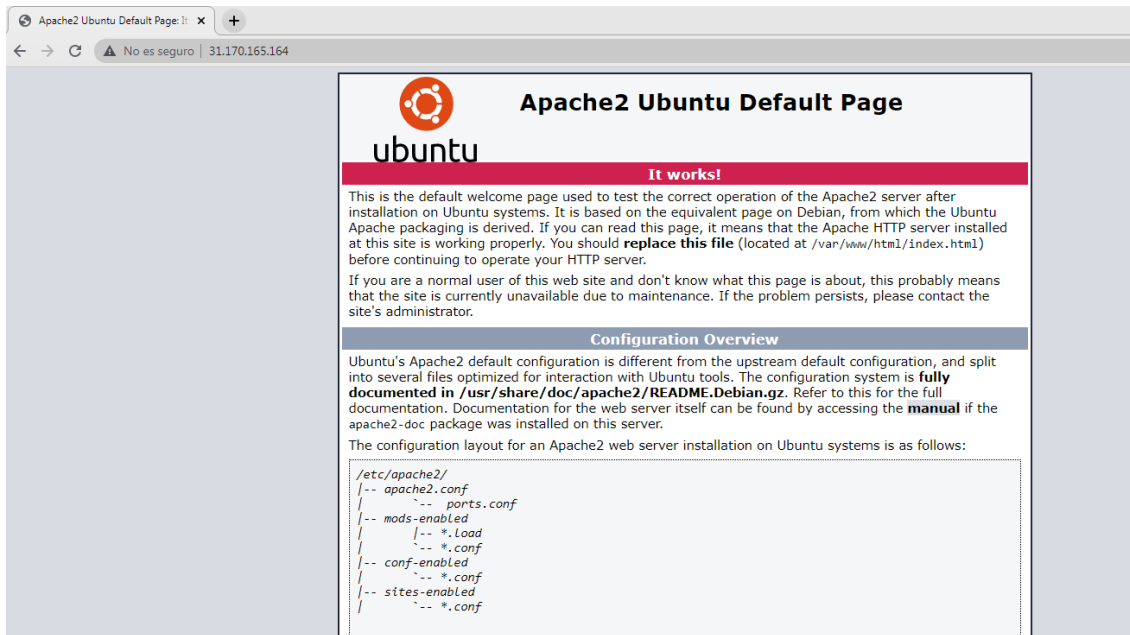


Figura 16-3: Comprobación del servidor web funcionando correctamente en la dirección <http://31.170.165.164/>

Realizado por: Mora, Marcela, 2022

Una vez finalizado se procede a instalar la base de datos.

3.4.7. *Instalación de la base de datos*

Para proceder a instalar la base de datos MySQL en la parte de administración de consola ingresamos el siguiente comando.

```
sudo apt-get install mysql-server
```

- **Con el siguiente comando se comprueba el estado y verifica el servicio.**

```
sudo systemctl status mysql
```

A continuación, instalamos las librerías necesarias para ejecutar el proyecto.

```
sudo apt-get install php libapache2-mod-php php-mcrypt php-mysql php-cgi php-curl php-json
```

Al haber realizado la instalación de la base de datos es recomendable habilitar el entorno gráfico de administración y acceso, para lo cual se instaló la extensión Phpmyadmin de MySQL mediante el siguiente comando.

```
sudo apt-get install phpmyadmin
```

De esta manera es posible acceder por la web a la parte de administración de la base de datos para crear la tabla y las columnas donde se guardará la información, se creó una base de datos denominada IoT en MySQL con una tabla con el nombre Sensor, en la tabla se crearon cuatro columnas: Nombre del paciente (Paciente), Presión Diastólica (PDIAS), Presión Sistólica (PSIS) y Fecha y Hora (Reading_time) con la configuración de colación de utf8_general_ci y de tipo varchar. Tal como se observa en la Figura 17-3

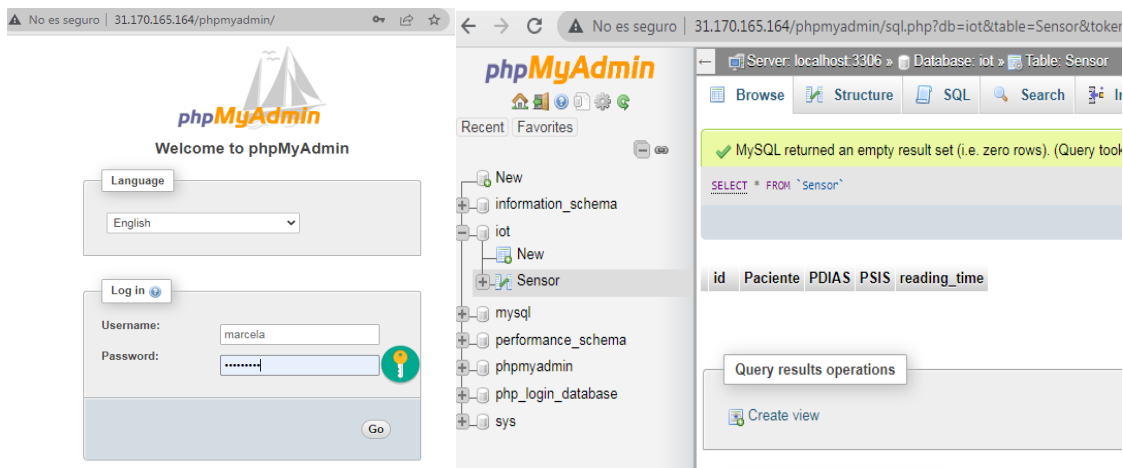


Figura 17-3: Ingreso y configuración de la base de datos, tabla y columnas para el almacenamiento de información.

Realizado por: Mora, Marcela, 2022

En el panel de administración en el servidor se ubicó el "Administrador de archivos" y luego, en el sitio public_html se crean los archivos en formato *.php*.

Al tener instalado todo lo necesario para poner en marcha el servidor, se procede a programar el script que recibirá la información del dispositivo biomédico y permitirá que se almacene en la base de datos. Para ello se procede a crear un archivo denominado *post-data.php* con todo el código necesario para su correcto funcionamiento.

Utilizando el siguiente comando, editamos y agregamos el código necesario.

```
sudo nano /var/www/html/post-data.php
```

El script cuenta con la programación en PHP para recibir las solicitudes entrantes de la tarjeta de desarrollo ESP32, de esta forma recibe la información del sensor y las credenciales de acceso a la base de datos por consiguiente almacena la información. En la Figura 18-3 se observa parte del código empleado para su correcto funcionamiento

```

root@knotty-rock: ~
GNU nano 2.9.3 /var/www/html/post
1 #!/usr/bin/php
2 MARCELA IOT
3 */
4 $servername = "localhost";
5 // Database
6 $dbname = "iot";
7 // Database user
8 $username = "marcela";
9 // Database user password
10 $password = "serqN3c.";
11 // API Key compatible con el ESP32
12 $api_key_value = "cPmATSab3j7F9";
13 $api_key = $fecha = $PDIAS = $PSIS = "";
14 if ($_SERVER["REQUEST_METHOD"] == "POST") {
15     $api_key = test_input($_POST["api_key"]);
16     if ($api_key == $api_key_value) {
17         $fecha = test_input($_POST["fecha"]);
18         $PDIAS = test_input($_POST["PDIAS"]);
19         $PSIS = test_input($_POST["PSIS"]);
20     }
21     // Conexión
22     $conn = new mysqli($servername, $username, $password, $dbname);
23     // Si error
24     if ($conn->connect_error) {
25         die("Falló la conexión" . $conn->connect_error);
26     }
27     $sql = "INSERT INTO Sensor (fecha, PDIAS, PSIS)
28     VALUES ('" . $fecha . "', '" . $PDIAS . "', '" . $PSIS . "')";
29     if ($conn->query($sql) === TRUE) {
30         echo "Creado correctamente";
31     }
32     else {
33         echo "Error: " . $sql . "<br>" . $conn->error;
34     }
35     $conn->close();
36     }
37     else {
38         echo "API Key erronea";
39     }
40 }

```

Figura 18-3: Código del scrip post-data.php
 Realizado por: Mora, Marcela, 2022

Una vez la información enviada es almacenada en la base de datos, es necesario procesarla para realizar los cálculos necesarios para mostrar los resultados. Para ello se creó el script *procesamientos.php* mediante el comando.

`sudo nano /var/www/html/procesamientos.php`

En este script se procede a programar el código para tomar los datos de la base datos y procesarlos. Ver Anexo 1: Se realizan los cálculos necesarios para mostrar los resultados al Doctor quién podrá determinar el diagnóstico del paciente. Los resultados se obtuvieron en base al software de monitoreo propio del equipo médico y los sugeridos por el profesional de la salud, los cuales son: A continuación, se muestra en la Tabla 21-3 los Parámetros Requeridos por el Profesional

Tabla 21-3: Parámetros Requeridos por el profesional de la salud para diagnosticas al paciente luego del mapa

#	NOMBRE DEL PARÁMETRO
1	Nombre del paciente
2	Hora de Inicio de Monitorización
3	Hora de finalización de la Monitorización
4	Muestras totales
5	Muestras Válidas
6	Valores máximos de la presión sistólica en la mañana y en la noche
7	Valores máximos de la presión diastólica en la mañana y en la noche

8	Valores mínimos de la presión sistólica en la mañana y en la noche
9	Valores mínimos de la presión diastólica en la mañana y en la noche
10	Promedios de la presión sistólica en la mañana y en la noche
11	Promedios de la presión diastólica en la mañana y en la noche
12	Carga de la presión sistólica
13	Carga de la presión diastólica
14	Grafica de la tendencia de presión sistólica
15	Grafica de la tendencia de presión diastólica

Realizado por: Mora, Marcela, 2022

Además, en el servidor web se programó para la parte administrativa una opción con restricción de acceso para descargar en formato .xlsx toda la información del paciente que se cuenta en la base de datos una vez finalizado el monitoreo, de esta manera comparar y validar la información con las tablas exportadas del software ABPMS, además de la opción de borrar la información de la base de datos una vez se haya respaldado. En la Figura 19-3 se puede observar parte del código del script para procesamiento de información.

```

root@knotty-rock: /var/www/html
GNU nano 2.9.3 procesamientos.php
---
MARCELA IOT
-->
<script language="javascript">
  var getIn = prompt("Ingrese la contraseña para acceder al sitio.")
  if (getIn=="MARCELA IOT")
    (location.href="http://31.170.165.164/esp-chart.php")
  else
    (alert("Contraseña incorrecta, acepta para ver la página de administrador"))
</script>
</php>
$servername = "localhost";
// REPLACE with your Database name
$dbname = "iot";
// REPLACE with Database user
$username = "marcela";
// REPLACE with Database user password
$password = "Mar@1902";

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

//Sql = "SELECT id, fecha, FDIAS, PSIS, reading_time FROM Sensor order by reading_time desc limit 40";
//cada 15 minutos en el día total 48 muestras y En la noche cada 30 minutos solo para calculos, se mantiene los 48: TOTAL 96
$Sql = "SELECT id, fecha, FDIAS, PSIS, reading_time FROM Sensor order by reading_time desc limit 96";

$result = $conn->query($Sql);

//MUESTRAS TOTALES
$Sql2 = "SELECT * FROM Sensor";

if ($result2=mysqli_query($conn,$Sql2)) {
    //echo "MODO DE MUESTRAS: ". $rowcount;
}

//SUMA DE UNA COLUMNA PRESION DIASTOLICA
$Sql_query1 = "SELECT SUM(FDIAS) AS count FROM Sensor";
$rowcount1 = $conn->query($Sql_query1);
while($rowcount1 = $rowcount1->fetch_array()){

```

Figura 19-3: Código del script procesamientos.php
Realizado por: Mora, Marcela, 2022

Al contar con todos los cálculos necesarios requeridos para poder entregar un diagnóstico al paciente, se procede a crear la aplicación Android.

3.4.8. Creación de la aplicación Android

Para la presentación de resultados se creó una aplicación móvil en la plataforma MIT App Inventor de Google Labs ideal para crear aplicaciones de software para Android. Inicialmente es necesario crearse una cuenta en la página oficial de App Inventor, a continuación, se inicia un nuevo proyecto donde es posible ir diseñando la pantalla de la aplicación agregando recursos tal como se observa en la Figura 20-3

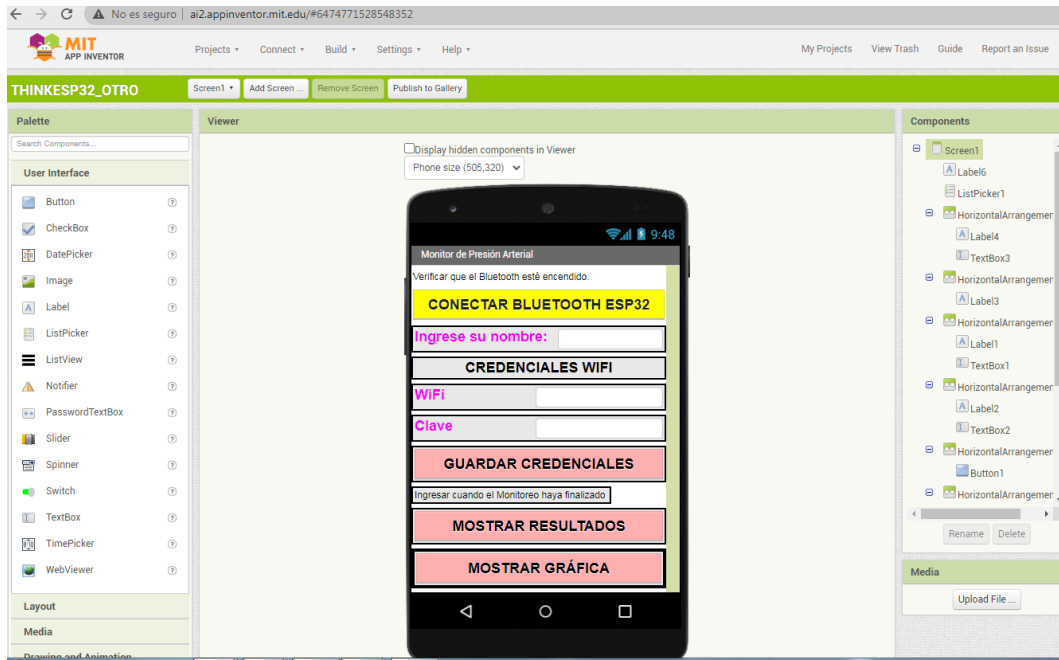


Figura 20-3: Diseño visual de la Aplicación en App Inventor
Realizado por: Mora, Marcela, 2022

Al contar con el diseño se procede a programar a partir de bloques. La aplicación deberá conectarse a la tarjeta de desarrollo ESP32 mediante Bluetooth para enviar el nombre del paciente, la red y la clave Wifi. Como se observa en la Figura 21-3 la programación permite la interacción de los dispositivos para el envío del texto ingresado en cada campo.

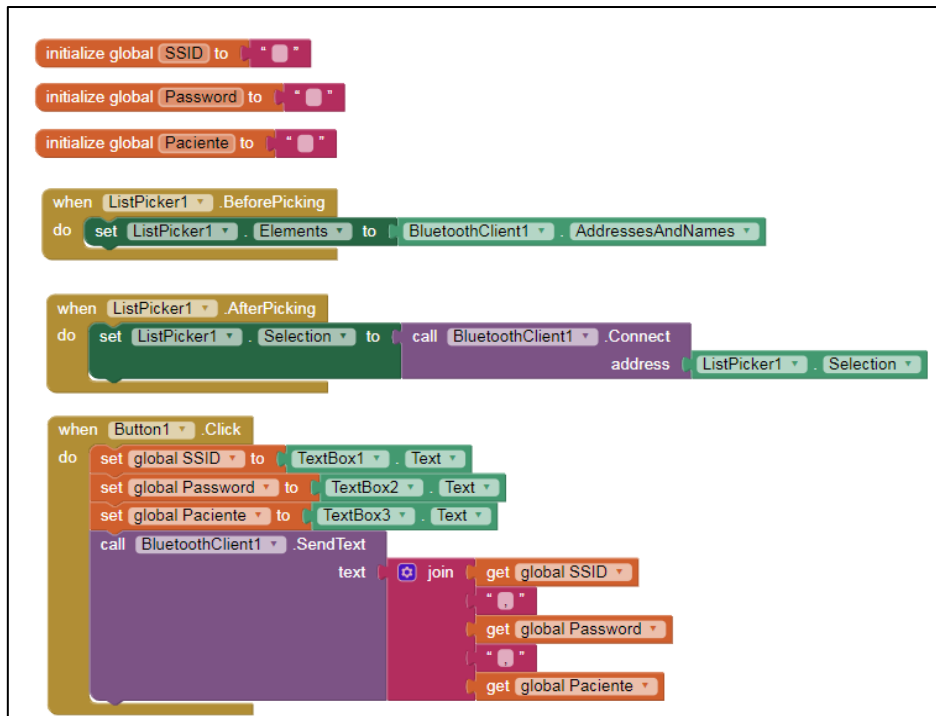


Figura 21-3: Programación de los bloques del registro del paciente y red Wifi
 Realizado por: Mora, Marcela, 2022

Para la visualización de los resultados en la aplicación Android existen botones programados que se enlazan con el servidor para la toma de la data procesada. En la Figura 22-2 se observan los bloques programados en la aplicación Android que permiten la visualización de los resultados.

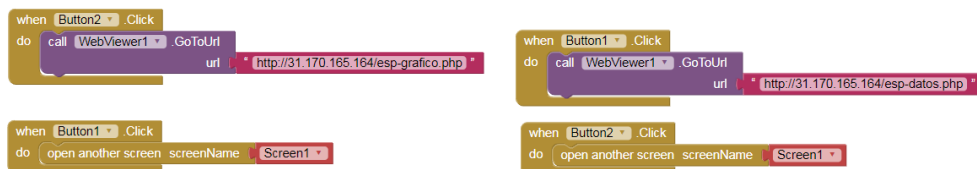


Figura 22-3: Programación de bloques para mostrar los resultados en la aplicación Android
 Realizado por: Mora, Marcela, 2022

Los botones acceden a los resultados de los scripts esp-grafico.php y esp-datos-php, mostrados en las Figura 23-3


```

root@knotty-rock: /var/www/html
GNU nano 2.9.3

$# inicio = $readings_time3[0];
$# fin = $readings_time3[$k-1];
$result->free();
$conn->close();
?>

<!DOCTYPE html>
<html>
<meta name="viewport" content="width=device-width, initial-scale=1">
<script src="https://code.highcharts.com/highcharts.js"></script>
<style>
  body {
    min-width: 310px;
    max-width: 1280px;
    height: 500px;
    margin: 0 auto;
  }
  h2 {
    font-family: Arial;
    font-size: 2.5rem;
    text-align: center;
  }
</style>
<body>
  <h2>DATOS DEL MONITOREO</h2>
  <h3>PACIENTE : <?php echo $nombre?> </h3>
  <h3>Hora De Inicio: <?php echo $h_inicio?></h3>
  <h3>Hora De Finalizaci n: <?php echo $h_fin?> </h3>
  <h3> MEDICIONES </h3>
  <h3> Muestras Totales: <?php echo $rowcount?></h3>
  <h3> Muestras V lidas : <?php echo $rowcount3?> </h3>
  <h3> VALORES M axIMOS </h3>
  <h3> Sist lica Ma ana : <?php echo $sismanana2?> </h3>
  <h3> Sist lica Noche : <?php echo $sisnoche2?></h3>
  <h3> Sist lica Total : <?php echo $maxsis?> </h3>
  <h3> Diast lica Ma ana : <?php echo $diasmanana2?></h3>
  <h3> Diast lica Noche : <?php echo $diasnoche2?> </h3>
  <h3> Diast lica Total : <?php echo $maxdias?> </h3>
  <h3> M mNIMOS </h3>
  <h3> Sit lica Ma ana : <?php echo $sismanana3?> </h3>
  <h3> Sist lica Noche : <?php echo $sisnoche3?></h3>
  <h3> Sit lica Total : <?php echo $minsie?> </h3>
  <h3> Diast lica Ma ana : <?php echo $diasmanana3?></h3>
  <h3> Diast lica Noche : <?php echo $diasnoche3?></h3>
  <h3> Diast lica Total : <?php echo $mindias?></h3>
  <h3> PROMEDIOS </h3>

  font-family: Arial;
  font-size: 2.5rem;
  text-align: center;
}
</style>
<body>
  <div id="chart-diastolica" class="container"></div>
  <div id="chart-sistolica" class="container"></div>
  <div id="chart-pressure" class="container"></div>
  <script>
    var fecha = <?php echo $fecha?>;
    var PDIAS = <?php echo $PDIAS?>;
    var PSIS = <?php echo $PSIS?>;
    var reading_time = <?php echo $reading_time?>;

    var chartT = new Highcharts.Chart({
      chart: { renderTo: 'chart-diastolica' },
      title: { text: 'VALORES DE PRESI ^sN DIAST ^sLICA' },
      series: [{
        showInLegend: false,
        data: PDIAS
      }],
      plotOptions: {
        line: { animation: false,
          dataLabels: { enabled: true }
        },
        series: { color: '#059e8a' }
      },
      xAxis: {
        type: 'datetime',
        categories: reading_time
      },
      yAxis: {
        title: { text: 'PRESION DIAST ^sLICA' }
        //title: { text: 'Temperature (Fahrenheit)' }
      },
      credits: { enabled: false }
    });

    var chartH = new Highcharts.Chart({
      chart: { renderTo: 'chart-sistolica' },
      title: { text: 'VALORES DE PRESI ^sN SIST ^sLICA' },
      series: [{
        showInLegend: false,
        data: PSIS
      }],
    });
  </script>

```

Figura 23-3: Scripts usados por la aplicación móvil para mostrar los resultados
 Realizado por: Mora, Marcela, 2022

3.4.9. Implementación del prototipo biomédico

El prototipo biomédico, se procedió a instalarlo en una placa con soldadura conjuntamente con todos los elementos electrónicos, tal como se observa en la Figura 24-3



Figura 24-3: Proceso de implementación del prototipo biomédico
 Realizado por: Mora, Marcela, 2022

A continuación, el sensor biomédico se lo instaló en un case para que el paciente pueda insertar el dedo y posicionarlo de manera óptima para evitar que se mueva. Luego de ello la placa y todos

los componentes electrónicos fueron implementados en un guante de poliéster con un bolsillo con velcro para utilizarlo de forma adecuada, como se observa en la Figura 25-3



Figura 25-3: Prototipo biomédico instalado en un guante de poliéster listo para su uso
Realizado por: Mora, Marcela, 2022

De esta manera el prototipo biomédico se encuentra listo para que los pacientes puedan usarlo sin inconvenientes. Adicional a ello a cada paciente que requirió el monitoreo de 24 horas se le entregó el manual de uso y una breve explicación de uso por parte del doctor tal como se muestra en la Figura 26-3



Figura 26-3: Explicación de uso e implementación del prototipo biomédico conjuntamente con el equipo médico a una paciente
Realizado por: Mora, Marcela, 2022

CAPÍTULO IV

4. RESULTADOS

En este capítulo se presentan los resultados de las pruebas realizadas del prototipo de sistema IoT para la monitorización de datos de presión arterial a través de un sensor foto electrónico en los pacientes que se realizaron el MAPA. Se comprobó el correcto funcionamiento del dispositivo biomédico implementado, la toma de datos de presión arterial, el envío y recepción de la información en el servidor VPS y la visualización de los cálculos realizados en la aplicación móvil. A continuación, la comparación de los resultados entre el equipo médico y el prototipo realizado en esta investigación. Finalmente se presenta el costo de la implementación del prototipo. Llega el paciente con síntomas de hipertensión para comprobar el medico coloca el Equipo Médico CONTEC PM50, el cual el equipo monitoreara las 24 horas al paciente los valores de Presión Arterial al momento de entregar al paciente el médico revisa las baterías del Equipo que tenga un correcto funcionamiento al segundo día el paciente regresa donde el Médico le retira el Equipo médico mediante el software ABPM en donde revisa los datos del paciente que fueron tomados durante las 24 horas y da el diagnóstico

4.1. Lugar de estudio

En la ciudad de Riobamba los lugares especializados para realizar el Mapeo pueden ser en Hospitales públicos y clínicas privadas en el sector público el paciente con síntomas de Presión Alta el diagnóstico es muy demoroso por la demanda de pacientes y limitados recursos médicos, por lo cual se optó probar el prototipo biomédico realizado en esta investigación en el consultorio del Doctor Fiallos cardiólogo ubicado en la clínica Metropolitana, debido a que en el Consultorio tratan y diagnostican casos mediante el Mapeo siendo el médico especializado en este tipo de enfermedades y óptimo para realizar las pruebas necesarias y la comparación de resultados en la tabla 1-4 se ubica el lugar por medio de Google Earth

Tabla 1-4: Localización Hospital Metropolitano

Provincia	Chimborazo
Ciudad	Riobamba
Sector	Riobamba Centro Oriente
Coordenadas	1° 40' 7.3" sur 78° 38' 55.4" oeste

Realizado por: Mora, Marcela, 2022

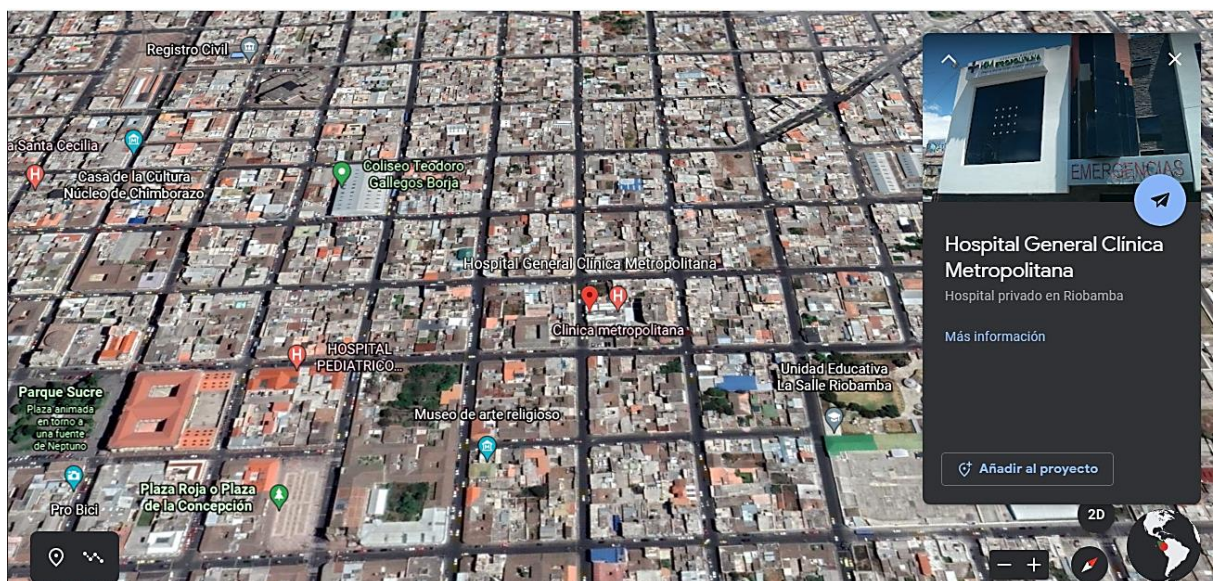


Figura 1-4: Localización Hospital Metropolitano
 Realizado por: Mora, Marcela, 2022

4.1.1. *Tamaño de la muestra*

En el consultorio del doctor Fiallos se lleva un historial mensual de pacientes que han realizado el mapa y diagnósticos con alguna enfermedad de presión alta por año, para esta investigación el mes de estudio fue diciembre para realizar el mapeo a pacientes que lo requieran tanto con el equipo médico y el prototipo de esta investigación. En la tabla 2-4 se listan la cantidad de pacientes mayores a 65 años que se han realizado el monitoreo ambulatorio por mes

Tabla 2-4: Cantidad Hombres, Mujeres mayores de 65 años

	Tamaño	Hombres	Mujeres
Enero	28	10	18
Febrero	26	9	17
Marzo	22	8	14
Abril	18	9	9
Mayo	20	8	12
Junio	22	11	11
Julio	25	11	14
Agosto	16	7	9
Septiembre	14	7	7
Octubre	19	9	10
Noviembre	27	7	20

Realizado por: Mora, Marcela, 2022

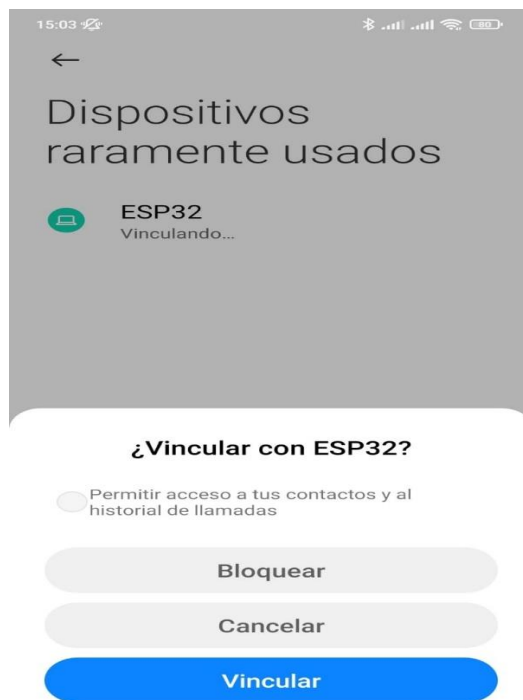
Como se puede observar en la tabla 2-4 las mujeres mayores de 65 años son las más propensas a sufrir Hipertensión Arterial, para este estudio se considera este grupo de personas donde al identificarse alguna paciente en el mes de diciembre se procederá con el monitoreo

4.1.2. Pruebas de funcionamiento

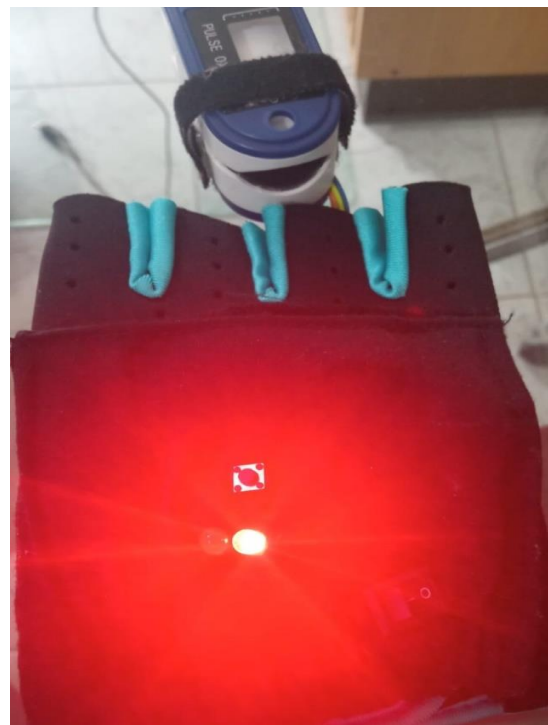
4.1.2.1. Funcionamiento del Prototipo Biomédico

- **Tarjeta ESP 32**

La tarjeta ESP 32 funciona correctamente con una alimentación de batería de 9v, luego que se enciende el interruptor se activa el bluetooth y está listo para enlazarse con el dispositivo móvil, Como se muestra en la Figura 2-4 a, Emparejamiento y en la Figura 2-4 b, el Funcionamiento de Bluetooth



a) Prueba de funcionamiento y emparejamiento de Bluetooth



b) Comprobación de funcionamiento de Bluetooth

Figura 2-4: Tarjeta ESP 32

Realizado por: Mora, Marcela, 2022

- **Wifi de Tarjeta de Desarrollo ESP 32**

El funcionamiento del wifi lo podemos comprobar al mediante el monitor del IDE de Arduino al momento de enviar las credenciales por la aplicación la tarjeta almacena esa información de red y se enlaza al internet el indicador en el prototipo medico es el encendido del led verde y el apagado del led rojo: Se observa en la Figura 3-4 a, se aprecia la Conexión a Wifi y en la Figura 3-4 b, Comprobación visual de conexión a wifi exitosa



Figura 3-4: Wifi de Tarjeta de Desarrollo ESP 32

Realizado por: Mora, Marcela, 2022

- **Sensor Biomédico**

Se verifica que el funcionamiento del sensor es óptimo ya que realiza su calibración gradualmente se va a dar hasta llegar al 100 % una vez realizada la calibración el sensor mide la presión arterial del paciente en cuanto se comprueba mediante el ide de Arduino que está tomando datos el sensor de presión arterial: En la Figura 4-4 a, se puede observar la Calibración del Sensor, en la Figura 4-4 b, se observa la Toma de datos de Presión Arterial.

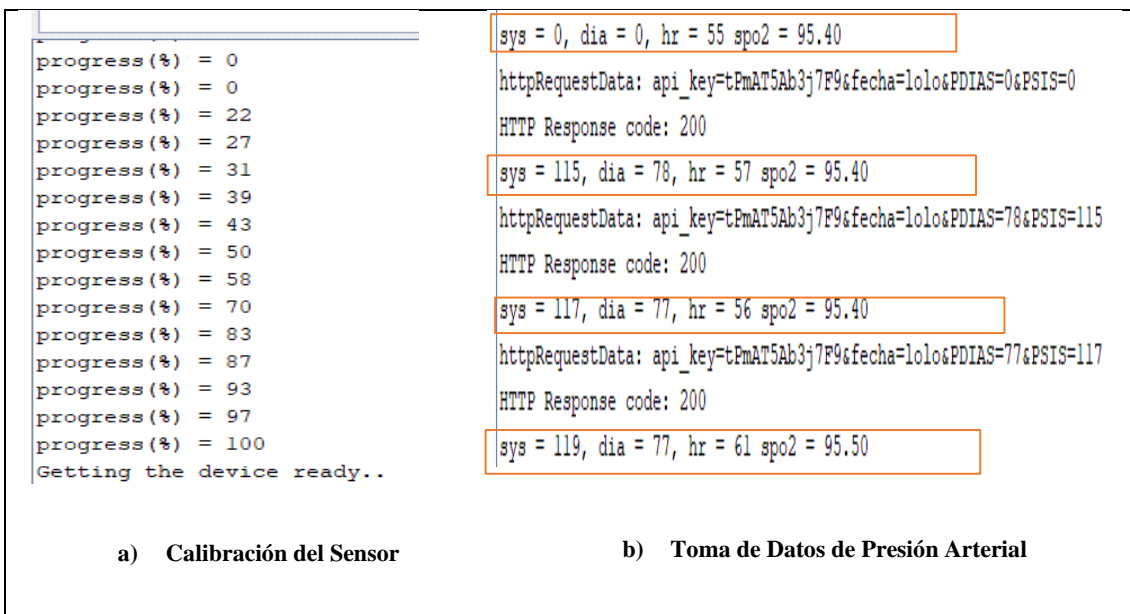


Figura 4-4: Sensor Biomédico

Realizado por: Mora, Marcela, 2022

4.1.3. Funcionamiento del Servidor VPS

Se comprobó que la placa se conecta al servidor VPS y obtiene la respuesta que el enlace esta creado mediante la estructura cliente servidor: Como se observa en la Figura 5-4 a, el Envió de datos al Servidor, en la Figura 5-4 b, se puede observar la Respuesta del Servidor



Figura 5-4: Funcionamiento del Servidor VPS

Realizado por: Mora, Marcela, 2022

Al iniciar el monitoreo el prototipo enviara a la base de datos del servidor VPS los datos de Presión Arterial cada 15 minutos en el caso del horario de la noche existe un control para los cálculos en un intervalo de 30 minutos, en total se obtiene 75 valores de cada presión arterial sistólica y diastólica

4.1.4. Funcionamiento de la Aplicación Android

Para comprobar el funcionamiento de la aplicación Android es posible descargarlo directamente de la plataforma mediante el escaneo de un código QR por el cual se descarga en el móvil y así la aplicación se abre correctamente: Se observa en la Figura 6-4 Validación del funcionamiento de la Aplicación

PANTALLA PRINCIPAL



PANTALLA DE CÁLCULOS



PANTALLA DE GRÁFICAS



Figura 6-4: Validación del funcionamiento de los botones y pantallas de la aplicación

Realizado por: Mora, Marcela, 2022

4.1.5. Comprobación de funcionamiento de los Cálculos

- **Resultados de los cálculos**

Se comprobó que los cálculos realizados por el script de procedimientos son correctos, adicionalmente a los resultados mostrados en la aplicación móvil es posible acceder a toda la información ingresando a <http://31.170.165.164/procesamientos.php> página en la cual se reflejan los resultados, gráficas y la posibilidad de descargar toda la información monitoreada en un archivo *.xlsx*. Al haber validado que la información mostrada en el servidor es correcta, la información se mostrará en la aplicación móvil.

A continuación, se muestra la página web para la administración del sitio la cual se puede observar en la Figura 7-4.

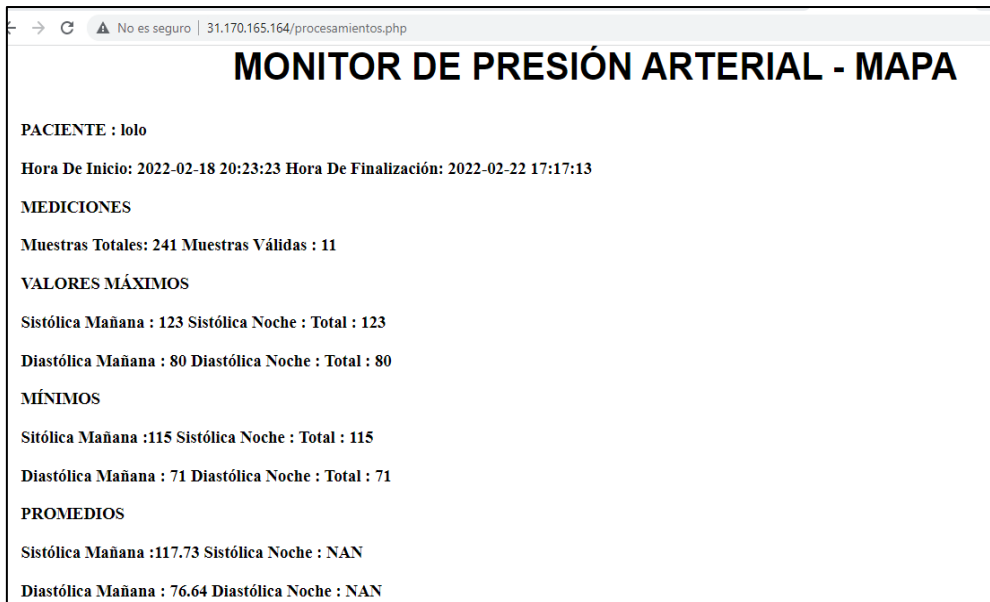


Figura 7-4: Comprobación de los cálculos realizados en la parte de Administración
Realizado por: Mora, Marcela, 2022

4.2. Resultados

Para comprobar la confiabilidad del cálculo de presión arterial del sensor implementado en el sistema se realizó la comparación de los datos entregados por el sensor con los datos que entrega un equipo comercial. Se hizo uso de un monitor de paciente multifuncional de la marca CONTEC PM50, que mide la Presión Arterial. Es de tamaño grande y es necesario el uso de un brazalete que ejerce presión en el brazo para que realice los cálculos de medición. Equipo PM50 usado actualmente por el Doctor Fiallos. En la Figura 8-4 el Equipo Médico CONTEC PM 50



Figura 8-4: Equipo Médico CONTEC PM50
Realizado por: Mora, Marcela, 2022

Para proceder con las pruebas de funcionamiento el doctor entrega el equipo médico conjuntamente con el guante que contiene el prototipo biomédico al paciente, por lo cual al iniciar el monitoreo el paciente simplemente deberá configurar la red inalámbrica de la red wifi mediante la aplicación móvil, una vez el prototipo biomédico se encuentre enlazado a la red wifi existe un led indicador que mostrará que el dispositivo está funcionando correctamente por las siguientes 24 horas

Al haber terminado el mapeo el paciente apaga el Prototipo Biomédico del interruptor, se retira el guante y se dirige al consultorio médico, el médico procede a retirar el equipo médico del paciente para extraer los datos almacenados en la memoria mediante el programa ABPMA, software utilizado para la presentación de resultados del MAPA del paciente este software permite descargar un informe necesario para el diagnóstico este informe nos serviría para realizar la comparación de resultados con el Prototipo Biomédico para ello es posible acceder a la página web del servidor para descargar la tabla completa de valores de la presión arterial del paciente actual: En la Figura 9-4, podemos apreciar las opciones disponibles para la parte de la administración.



Figura 9-4: Opciones de la parte de administración
Realizado por: Mora, Marcela, 2022

En la Figura 10-4 se aprecian los valores de la Presión Arterial del Software ABPM exportado por el doctor

Resultado examinación					
Todas las medias PA:	114/67mmHg	Umbral PA:	130/80mmHg		
Medias PA día:	115/67mmHg	Umbral PA:	135/85mmHg		
Medias PA noche:	104/62mmHg	Umbral PA:	120/70mmHg		
Valor carga PA día:Normal<40%		Valor carga PA noche:Normal<50%			
SYS(>135mmHg) 9.4%		SYS(>120mmHg) 0.0%			
DIA(>85mmHg) 5.3%		DIA(>70mmHg) 12.5%			
SYS máxima	170mmHg	Tiempo	2008/7/25 18:05	SYS mínima	81mmHg
DIA máxima	95mmHg	Tiempo	2008/7/25 18:05	DIA mínima	41mmHg
Ritmo circadiano de PA:DES SyS noche.	9.5%	Des DIA noche	7.3%	Normal:10%-20%	
PA CV:	Todos: SYS	12.9%	DIA	14.6%	
	Día: SYS	13.0%	DIA	14.6%	
	Noche:SYS	6.7%	DIA	12.1%	

Figura 10-4: Tabla de valores de Presión Arterial del software ABPM utilizado por el doctor
Realizado por: Mora, Marcela, 2022

4.2.1. Análisis del Prototipo Biomédico

En el mes de diciembre la cantidad de pacientes para este estudio fueron de once tal como se muestra en la Tabla 3-4, siendo las muestras utilizadas para los cálculos y la comparación de resultados entre el equipo médico y el prototipo biomédico. Además, es necesario obtener el error de cada muestra para validar la obtención de información del Prototipo Biomédico, mediante siguiente formula

$$SE = \frac{DS}{\sqrt{\# Muestras}}$$

Dónde:

SE=Error Estándar

DS=Desviación Estándar

M=Muestras

- **Carga**

Se utilizaron las siguientes fórmulas matemáticas para determinar los parámetros necesarios para el diagnóstico del paciente:

Para calcular la carga para Hipertensión Tipo 1 se utiliza la siguiente ecuación:

$$HT1 = \frac{(\sum \#Muestras \geq 140 + \#Muestras \leq 159) * 100}{\#Muestras Totales Validas}$$

La condición es que el resultado $HT_1 \geq 40$ sea mayor a cuarenta lo que sobre entiende que el paciente tiene hipertensión tipo 1

Para calcular la carga para Hipertensión Tipo 2 se utiliza la siguiente ecuación:

$$HT = (\Sigma \#Muestras \geq 160 + \#Muestras \leq 179) * 100 / \#Muestras \text{ Totales Validas}$$

La condición es que el resultado $HT_2 \geq 40$ sea mayor a cuarenta lo que sobre entiende que el paciente tiene hipertensión tipo 2

Para calcular la carga para Hipertensión Tipo 3 se utiliza la siguiente ecuación:

$$HT = (\Sigma \#Muestras > 180) * 100 / \#Muestras \text{ Totales Validas}$$

La condición es que el resultado $HT_3 \geq 40$ sea mayor a cuarenta lo que sobre entiende que el paciente tiene hipertensión tipo 3

Dónde: HT_1 es Hipertensión Tipo 1, HT_2 es Hipertensión Tipo 2, HT_3 es Hipertensión Tipo 3

Tabla 3-4: Análisis de Presión Arterial Equipo Médico, Prototipo Biomédico

Nombre	Cédula	Carga sistólica Prototipo	Carga diastólica prototipo	Diagnostico prototipo	Diagnostico Equipo
Carguacundo Hidalgo Rosa Elvira	0600555809	64%	8%	Hipertensión Tipo 1,2	Hipertensión Tipo 1,2
Oyate Ramos Fanny Yolanda	1800307355	62.67%	9.33%	Hipertensión Tipo 1,2	Hipertensión Tipo 1,2
Palmay Arévalo Mariana de Jesús	0601915663	64%	13.33%	Hipertensión Tipo 1,2	Hipertensión Tipo 1,2

(Tenelema Paguay María Rebeca	0600784441	88%	4.67%	Hipertensión Tipo 2	Hipertensión Tipo 2
Miyarcaja Miyacaja María Gregoria	0602245953	61.33%	13.33%	Hipertensión Tipo 1	Hipertensión Tipo 1
Calero León Dolores Patricia	0601109440	62.67%	17.33%	Hipertensión Tipo 1	Hipertensión Tipo 1
Paguay Mocha María Rosa	0600855894	57.33%	16%	Hipertensión Tipo 1	Hipertensión Tipo 1
Ortega Mosquera Yesenia Elizabeth	0601553951	58.67%	12%	Hipertensión Tipo 1	Hipertensión Tipo 1
Moncayo Cuello Ercilia Nelly	0602569428	60%	9.33%	Hipertensión Tipo 1	Hipertensión Tipo 1
Barreno Rea Jimena Verenice	0600948301	61.33%	12%	Hipertensión Tipo 1	Hipertensión Tipo 1
Alvarez duche clara Luz	0602330921	57.33%	33.33%	Hipertensión Tipo 1	Hipertensión Tipo 1

Realizado por: Mora, Marcela, 2022

Se muestra los valores necesarios para el diagnóstico del paciente del Equipo Médico y del Prototipo Biomédico Para proporcionar el diagnóstico, al obtener toda la información el doctor es capaz de diagnosticar a cada paciente mediante la tabla de resultados por lo tanto se considera anormal la presencia de 40% o más de registros sobre los valores de presión arterial, ya que sobre esta cifra hay alta correlación de presencia de enfermedad (Dr Germinal Alvarez, y otros, 2006,pp 5).De esta forma es posible diagnostica el tipo de Hipertensión que tiene el Paciente .

Tabla 4-4: Análisis de Presión Arterial

Número de Muestra Pacientes	Error Presión Sistólica	Error Presión Diastólica
1	1.92	1.25
2	2.26	1.27
3	2.09	1.43
4	2.34	1.87
5	2.17	1.56

6	2.14	1.52
7	2.15	1.32
8	3.09	1.77
9	2.24	1.33
10	2.06	1.23
11	2.65	1.58
PROMEDIO	25.11	16.13
MEDIA DEL ERROR	2.28	1.5

Realizado por: Mora, Marcela, 2022

Una vez realizada la medición y con los datos obtenidos en la Tabla 4-4, indica que el sensor implementado en el prototipo es confiable con una tasa de error de 2.8 % en Presión Sistólica y en Presión Diastólica 1.5 % estos valores se encuentran dentro del margen de error permitido que es de ± 3 , con un rango de variación de 0 a 3 puntos en la toma de presión arterial tomando de igual manera referencia del Equipo Médico CONTEC PM50 que tiene un rango de variación de 3 puntos. Según información de expertos en el área médica se acepta un margen de error de hasta el 3%, el resultado de los valores obtenidos por el sensor del prototipo es confiable y los datos pueden ser tomados con referencia. (Juliana Acevedo López, y otros, 2015, pp147)

Para validar los cálculos se usó el Software Estadístico R.

4.3. Análisis Económico del Prototipo

En el Análisis Económico se muestra el estudio de costos invertido en el prototipo implementado. Se detalla el presupuesto en la Tabla 5-4 el Análisis Económico del Prototipo

Tabla 5-4: Análisis Económico del Prototipo

Dispositivos	Cantidad	Costo Unitario \$	Costo Total \$
Proto Central Sensor	1	30	30
Tarjeta de Desarrollo ESP 32	1	8	8
Diodo Led	2	0.30	0.60
Pulsador	1	0.15	0.15
Resistencia de 300 Ohmios	2	0.20	0.40
Batería CR2 3V	1	3	3
Guante	1	4	4
Cable de Arduino	3	0.50	1.50
CABLE DE ALIMENTACION	1	1.50	1.50
Switch	1	0.25	0.25
Case del Sensor	1	1.50	1.50
HOSTINGER	Valor Mensual	10	10
		TOTAL	60.90

Realizado por: Mora, Marcela, 2022

Un paciente que requiere el monitoreo de 24 horas, el medico decide alquilar el Equipo Médico CONTEC PM50 le cuesta 70 dólares

VENTAJAS

- Notablemente se puede evidenciar que el costo del Equipo Médico CONTEC PM50 es elevado
- El uso del Prototipo Biomédico es de fácil uso y más ligero además de poder consultar en tiempo real los resultados e indicadores necesarios para el diagnostico
- En relación al equipo médico CONTEC PM50 debido a su funcionamiento el prototipo Biomédico no es sensible al movimiento de paciente por lo que puede monitorear en relación al Equipo Médico CONTEC PM50

DESVENTAJAS

- Pese a que los datos del Prototipo Biomédico son muy similares al Equipo Médico CONTEC PM50 es necesario que el Medico valide la información

CONCLUSIONES

- La implantación de un Prototipo Biomédico se realizó con acceso a la red inalámbrica de bajo costo cuya operatividad está basada en tres nodos llamados: lector, recolector y almacenamiento que en conjunto permiten acceder a los datos almacenados en una base de datos.
- Se optó por el uso de tarjeta ESP 32, sensor ProtoCentral Pulse Express, por ser elementos electrónicos asequibles y de tamaño reducido que los hace ideales para la implementación y portabilidad del prototipo.
- Esta investigación se monitorea en el mes de diciembre 2021 la cantidad de 11 pacientes siendo las muestras utilizadas para los cálculos y la comparación de resultados entre el equipo médico y el prototipo biomédico
- Se hizo la comparación del Equipo Biomédico con el Equipo Médico CONTEC PM50 del Consultorio del Dr. Fiallos donde hubo un coeficiente de error 2.28 % Presión Sistólica y 1.5% para Presión Diastólica de esta manera otorgan el diagnóstico al paciente mediante los valores de máximos y mínimos y carga siendo el prototipo óptimo para realizar el monitoreo
- De las pruebas se determinó que el Prototipo Biomédico presenta un error menor a 3 en la fiabilidad de los datos monitoreados en relación al equipo médico CONAPEC PM 50 utilizado en el consultorio del Doctor Fiallos
- Para la elaboración del Prototipo Biomédico se usa la Placa ESP 32 la cual se programa en IDE de Arduino conjuntamente con el sensor Biomédico MAX 32664 y los elementos electrónicos en un Guante Poliéster además se usó un servidor VPS con un Ubuntu 18.04 donde se instaló el servidor Apache, la Base de datos MySQL se programaron los scripts para recepción y almacenamiento de información y muestra cálculos necesarios para mostrar la información en el programa App Inventor

- El equipo médico presenta un costo de \$380 mayor al del prototipo implementado siendo un costo de \$60.90 lo que el prototipo biomédico es un ahorro significativo y fiable para la medición de presión arterial

RECOMENDACIONES

- Se recomienda el uso de una Batería de 9V como alimentación en todos los componentes electrónicos del Equipo Biomédico, no es recomendable usar la alimentación de 3v porque el dispositivo no satisface la alimentación a todos los componentes
- Diseño del Guante de un material más ergonómico, reducción de materiales e implementación de otro tipo de tecnologías inalámbricas y las posibilidades de que la placa soporte y cuente con una materia interna para conectarse al servidor en la nube
- Se sugiere que para mejorar las funcionalidades del prototipo permitir el uso de sensores para la medición de otras variables fisiológicas
- Se recomienda para futuras investigaciones el uso de más sensores para precisar la medición de la Presión Arterial
- Recomienda que el diseño sea más compacto para el uso de cualquier persona indistinto que se requiera un diagnóstico del médico.

BIBLIOGRAFÍA

AKYILDIZ, I. & JORNET, J. The Internet of nano-things. *IEEE Wireless Communications* [en línea], 2010. 17(6), pp. 58-63. ISSN 1536-1284. DOI 10.1109/MWC.2010.5675779. Disponible en: <http://ieeexplore.ieee.org/document/5675779/>.

AL-FUQAHA, A.; et al. *Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications*. 2015. S.l.: s.n.

ALTAMIRANO ANDRADE, D.C. *Aplicación móvil con realidad aumentada como estrategia de marketing 2.0 para el menú del restaurante chimichurri y moros y menestras en la ciudad de ambato* [en línea]. Ambato: Universidad técnica de ambato. 2017. [Consulta: 16 marzo 2022]. Disponible en: https://repositorio.uta.edu.ec/bitstream/123456789/26425/1/Tesis_t1311si.pdf.

ALWAN; ALA. Informe sobre la situación mundial. [en línea], 2010. [Consulta: 15 marzo 2022]. Disponible en: www.who.int.

ARANDA, D. *Electronica. Plataformas Arduino y Raspberry Pi | PDF | Informática | Informática y tecnología de la información.* [en línea]. 2014. [Consulta: 17 marzo 2022]. Disponible en: <https://es.scribd.com/doc/283137870/Electronica-plataformas-Arduino-y-Raspberry-Pi-pdf>.

ATENCIO FLORES, D.B.; & MAMANI MACHACA, D. *Interconectividad Basado En Api Rest En Aplicaciones De La Municipalidad Provincial De LampA* [en línea]. Puno - Perú: 2017.[Consulta: 16 marzo 2022]. Disponible en: http://repositorio.unap.edu.pe/bitstream/handle/UNAP/6163/Atencio_Flores_Dilmerd_Mamani_Machaca_David.pdf?sequence=1&isAllowed=y.

AUGUSTIN, A., YI, J., CLAUSEN, T. & TOWNSLEY, W.M. A Study of LoRa: Long Range & Low Power Networks for the Internet of Things. *Sensors 2016, Vol. 16, Page 1466* [en línea], 2016, 16(9), pp. 1466. [Consulta: 17 marzo 2022]. ISSN 1424-8220. DOI 10.3390/S16091466. Disponible en: <https://www.mdpi.com/1424-8220/16/9/1466/htm>.

BACCELLI, E.; et al. RIOT OS: Towards an OS for the Internet of Things. *2013 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)* [en línea]. 2013. IEEE, pp. 79-80. ISBN 978-1-4799-0056-5. DOI 10.1109/INFCOMW.2013.6970748. Disponible en: <https://ieeexplore.ieee.org/document/6970748>.

BANDYOPADHYAY, D. y SEN, J., 2011. Internet of things: Applications and challenges in technology and standardization. *Wireless Personal Communications*, 2011. 58(1), pp. 49-69. ISSN 09296212. DOI 10.1007/s11277-011-0288-5.

BELTRAME, F. Worldwide emergency telemedicine services: the R&D EU projects perspective. *Proceedings of the IEEE/EMBS Region 8 International Conference on Information Technology Applications in Biomedicine, ITAB* [en línea], 1997, pp. 3-6. [Consulta: 16 marzo 2022]. DOI 10.1109/ITAB.1997.649385. Disponible en: <https://ieeexplore.ieee.org/document/649385>.

BELTRÁN, A.B. Arquitectura De Referencia En El Internet Of Things. por Adam Barberà Beltrán Specialist Consultor IoT en SUNQU. WhitePaper - PDF Descargar libre. [en línea]. 2016. [Consulta: 16 marzo 2022]. Disponible en: <https://docplayer.es/62802007-Arquitectura-de-referencia-en-el-internet-of-things-por-adam-barbera-beltran-specialist-consultor-iot-en-sunqu-whitepaper.html>.

BENAVIDEZ GÓMEZ, J.A.; et al. *Arquitectura Rest Para La Plataforma Uao* [En Línea]. S.L.: Universidad Autónoma De Occidente. 2019. [Consulta: 16 marzo 2022]. Disponible en: <https://red.uao.edu.co/bitstream/handle/10614/11597/T08762.pdf?sequence=5&isAllowed=y>.

BODUCH, A. React and React Native : use React and React Native to build applications for desktop browsers, mobile browsers, and even as native mobile apps. 2017.

BONILLA-FABELA, I.; et al. Iot, El Internet de las Cosas y la Innovación de sus Aplicaciones. *VinculaTégica EFAN* [en línea], 2016. 2(1), pp. 2313-2340. ISSN 2448-5101. Disponible en: <http://www.web.facpya.uanl.mx/Vinculategica/Revistas/R2/2313-2340 - Iot, El Internet De Las Cosas Y La Innovacion De Sus Aplicaciones.pdf>.

CAO, Q.; et al. The LiteOS Operating System: Towards Unix-Like Abstractions for Wireless Sensor Networks. *2008 International Conference on Information Processing in Sensor Networks (ipsn 2008)* [en línea]. 2008. IEEE, pp. 233-244. ISBN 978-0-7695-3157-1. DOI 10.1109/IPSN.2008.54. Disponible en: <http://ieeexplore.ieee.org/document/4505477/>.

CARDIER, M.; et al. Telemedicina. Estado actual y perspectivas futuras en audiología y otología. *Revista Médica Clínica Las Condes* [en línea], 2016. 27(6), pp. 840-847. [Consulta: 16 marzo 2022]. ISSN 0716-8640. DOI 10.1016/J.RMCLC.2016.11.016. Disponible en: <https://www.sciencedirect.com/science/article/pii/S0716864016301195>.

CAROLINA, M., et al. Monografía Internet De Las Cosas : Modelos De Comunicación , Desafíos Y Aplicaciones Monografía Internet De Las Cosas : Modelos De Comunicación , Desafíos Y Aplicaciones. , 2008. pp. 1-173.

CASTRO CUASAPAZ, S.E. *Tecnologías Web y Web Semántica en el Internet de las Cosas.* Quito - Ecuador: Universidad Internacional de la Rioja (UNIR). 2018.

CERUTTI, S.; et al. Guest EditorialSpecial Section on Smart Wearable Devices for Human Health and Protection. *IEEE Transactions on Information Technology in Biomedicine* [en línea], 2010, 14(3), pp. 691-693. ISSN 1089-7771. DOI 10.1109/TITB.2010.2048937. Disponible en: <http://ieeexplore.ieee.org/document/5475411/>.

CHAMOSO, P. et al. Tendencies of Technologies and Platforms in Smart Cities: A State-of-the-Art Review. *Wireless Communications and Mobile Computing*, 2018. ISSN 15308677. DOI 10.1155/2018/3086854.

CREATIVE COMMONS ATRIBUCION. Guía De Iniciación A App Inventor. [en línea]. 2015. [Consulta: 16 marzo 2022]. Disponible en: <https://codeweek.eu/docs/spain/guia-iniciacion-app-inventor.pdf>.

CROSBY, G. IEEE Xplore Full-Text PDF: *IEEE* [en línea].2013.[Consulta: 15 marzo 2022]. Disponible en: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6761322>.

DOMÍNGUEZ COBOS, A. Diseño e implementación de una arquitectura basada en tecnologías Open Source. [en línea], 2016. [Consulta: 16 marzo 2022]. Disponible en: <https://biblus.us.es/bibing/proyectos/abreproy/70884/fichero/TFM-Antonio+Cobos+Dominguez.pdf>.

DUNKELS, A. et al. Contiki - a lightweight and flexible operating system for tiny networked sensors. *29th Annual IEEE International Conference on Local Computer Networks* [en línea]. S.l.: IEEE (Comput. Soc.), 2004, pp. 455-462. ISBN 0-7695-2260-2. DOI 10.1109/LCN.2004.38. Disponible en: <http://ieeexplore.ieee.org/document/1367266/>.

EDGE COMPUTING CONSORTIUM Edge Computing Reference Architecture 2.0. [en línea]. [Consulta: 16 marzo 2022]. 2017. Disponible en: <http://en.eccconsortium.net/Uploads/file/20180328/1522232376480704.pdf>.

ELIZALDE, I. Los wearables como transformadores del sector sanitario - DirectorTIC. [en línea]. [Consulta: 16 marzo 2022]. 2020. Disponible en: <https://directortic.es/noticias/los-wearables-como-transformadores-del-sector-sanitario-2020040624268.htm>.

EVANS, D. The Internet of Everything How More Relevant and Valuable Connections Will Change the World. *Cisco*, 2012.

FANG, S., et al. An integrated system for regional environmental monitoring and management based on internet of things. *IEEE Transactions on Industrial Informatics*, 2014. 10(2), pp. 1596-1605. ISSN 15513203. DOI 10.1109/TII.2014.2302638.

FERNÁNDEZ, A.J. *Implementación de un cliente REST para un servicio de almacenamiento de ficheros basado en metadatos* [en línea]. 2015. [Consulta: 16 marzo 2022]. Disponible en: <https://repositorio.upct.es/bitstream/handle/10317/4561/pfc6085.pdf?sequence=1&isAllowed=y>

FERNÁNDEZ, I. React Native: de JavaScript al desarrollo de aplicaciones móviles - Izertis. [en línea]. 2020. [Consulta: 16 marzo 2022]. Disponible en: <https://www.izertis.com/es/-/blog/react-native-de-javascript-al-desarrollo-de-aplicaciones-moviles>.

FIELDING, R.T. *Architectural Styles and the Design of Network-based Software Architectures* [en línea]. Universidad De California, Irvine. 2000. [Consulta: 16 marzo 2022]. Disponible en: <https://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm>.

FORTINO, G.; et al. Agent-oriented smart objects development. *Proceedings of the 2012 IEEE 16th International Conference on Computer Supported Cooperative Work in Design (CSCWD)* [en línea]. 2012. IEEE, pp. 907-912. ISBN 978-1-4673-1212-7. DOI 10.1109/CSCWD.2012.6221929. Disponible en: <http://ieeexplore.ieee.org/document/6221929/>.

GARCÍA SÁNCHEZ, A.J. et al. Plataformas Middleware comerciales para la integración de flujos de vídeo bruto. [en línea], 2000. [Consulta: 16 marzo 2022]. Disponible en: <http://www.cs.wustl.edu/~schmidt/TAO.html>.

GARNOCK-JONES, T. AMQP and Beyond Messaging by Extending RabbitMQ. [en línea], 2010. [Consulta: 16 marzo 2022]. Disponible en: <https://www.erlang-factory.com/upload/presentations/229/ErlangFactorySFBay2010-TonyGarnock-Jones.pdf>.

GEMALTO. Seguridad en Internet de las Cosas. [en línea]. 2016. [Consulta: 16 marzo 2022]. Disponible en: <https://www.thalesgroup.com/es/countries/americas/latin-america/dis/iot/seguridad-en-iot>.

GIGLI, M. & KOO, S. Internet of Things: Services and Applications Categorization. *Advances in Internet of Things* [en línea], 2011. 1(02), pp. 27-31. ISSN 2161-6817. DOI 10.4236/ait.2011.12004. Disponible en: <http://www.scirp.org/journal/doi.aspx?DOI=10.4236/ait.2011.12004>.

GUNTURI, M.; et al. An overview of internet of things. *ITU*, 2012, (10)9, pp. 659-665. ISSN 1943023X.

HERNÁNDEZ PAZ, E.F. *Utilidad Clínica Del Monitoreo Ambulatorio De La Presion Arterial* [en línea]. Guatemala: Universidad de San Carlos de Guatemala. 1998. [Consulta: 17 marzo 2022]. Disponible en: http://biblioteca.usac.edu.gt/tesis/05/05_7812.pdf.

HWANG, K.O.; et al. Social support in an Internet weight loss community. *International Journal of Medical Informatics* [en línea], 2018, 79(1), pp. 5-13. ISSN 13865056. DOI 10.1016/j.ijmedinf.2009.10.003. Disponible en: <https://linkinghub.elsevier.com/retrieve/pii/S1386505609001725>.

ISLAM, S.K.; et al. Hassle-free vitals. *IEEE Microwave Magazine* [en línea], 2014, 15(7), pp. 525-533. [Consulta: 16 marzo 2022]. ISSN 15273342. DOI 10.1109/MMM.2014.2356148. Disponible en: <https://ieeexplore.ieee.org/document/6954654?arnumber=6954654>.

LEVIS, P.; et al. TinyOS: An Operating System for Sensor Networks. *Ambient Intelligence* [en línea]. 2005. Berlin/Heidelberg: Springer-Verlag, pp. 115-148. Disponible en: http://link.springer.com/10.1007/3-540-27139-2_7.

LI, S., XU, L. Da & ZHAO, S. The internet of things: a survey. *Information Systems Frontiers* [en línea], 2015, 17(2), pp. 243-259. ISSN 1387-3326. DOI 10.1007/s10796-014-9492-7. Disponible en: <http://link.springer.com/10.1007/s10796-014-9492-7>.

LORA ALLIANCE LoRa Alliance®. [en línea]. 2015. [Consulta: 17 marzo 2022]. Disponible en: <https://lora-alliance.org/>.

LUGMAÑA, G. et al. Registro Estadístico de Defunciones Generales. [en línea], 2015.

[Consulta: 15 marzo 2022]. Disponible en: www.ecuadorencifras.gob.ec.

MARINEL·LO I ROURA, J. & JUAN SAMSÓ, J. Diagnóstico hemodinámico en angiología y cirugía vascular. [en línea], 2003[Consulta: 17 marzo 2022]. Disponible en: <https://dokumen.pub/diagnostico-hemodinamico-en-angiologia-y-cirugia-vascular.html>.

MCDERMOTT-WELLS, P. What is Bluetooth? *IEEE Potentials* [en línea], 2005, 23(5), pp. 33-35. ISSN 0278-6648. DOI 10.1109/MP.2005.1368913. Disponible en: <http://ieeexplore.ieee.org/document/1368913/>.

MEDIO, A. Desarrollo de aplicaciones móviles con NativeScript. . [en línea].2017. [Consulta: 16 marzo 2022]. Disponible en: <https://alexmarket.medium.com/desarrollo-de-aplicaciones-moviles-con-nativescript-6940d89daa84>.

MOHAMMED, A. Advertisement in the Muslim World: A Critical Analysis from the Islamic Perspective. [en línea]. 2016. [Consulta: 16 marzo 2022]. Disponible en: <https://ideas.repec.org/a/aes/icmbdj/v2y2016i1p100-108.html>.

MORALES, D. Internet of Things (IoT) applications in Warehouse management-A review of new proposals. , 2019.

NAIK, N. Choice of effective messaging protocols for IoT systems: MQTT, CoAP, AMQP and HTTP. *2017 IEEE International Systems Engineering Symposium (ISSE)* [en línea]. 2017. IEEE, pp. 1-7. ISBN 978-1-5386-3403-5. DOI 10.1109/SysEng.2017.8088251. Disponible en: <http://ieeexplore.ieee.org/document/8088251/>.

NIETO, N. & VEGA, M.L. *Diseño de un prototipo de medición de señales fisiológicas utilizadas en Biofeedback* [en línea]. 2017. Córdoba: Universidad Nacional de Córdoba. [Consulta: 17 marzo 2022]. Disponible en: https://rdu.unc.edu.ar/bitstream/handle/11086/4808/Informe_PI_Nieto-Vega.pdf?sequence=1&isAllowed=y.

NOGUERA ARNALDOS, J. Sistema de diálogo basado en mensajería instantánea para el control de dispositivos en el internet de las cosas. *All rights reserved. IJES* [en línea], 2016. 281(4), pp. 1-30. Disponible en: <http://nadir.uc3m.es/alejandro/phd/thesisFinal.pdf%5Cnhttp://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Universidad+de+murcia#0>.

OMS Hipertensión. [en línea]. 2020. [Consulta: 15 marzo 2022]. Disponible en: <https://www.who.int/es/news-room/fact-sheets/detail/hypertension>.

OPS. Indicadores Básicos.2016

ORTIZ MONET, M. Implementación Y Evaluación De Plataformas En La Nube Para Servicios De Iot. [en línea], 2019. [Consulta: 16 marzo 2022]. Disponible en: www.etsit.upv.es.

PANG, Z. et al. Intelligent packaging and intelligent medicine box for medication management towards the Internet-of-Things. *International Conference on Advanced Communication Technology, ICACT* [en línea], 2014. pp. 352-360. [Consulta: 16 marzo 2022]. ISSN 17389445. DOI 10.1109/ICACT.2014.6779193. Disponible en: <https://ieeexplore.ieee.org/document/6779193>.

PÉREZ LEONES, K. *Estudio y análisis del protocolo de mensajería avanzado en el internet de las cosas para aplicación en el campo de la domótica*. [en línea]. 2019. Guayaquil: UNIVERSIDAD CATÓLICA DE SANTIAGO DE GUAYAQUIL. [Consulta: 16 marzo 2022]. Disponible en: <http://repositorio.ucsg.edu.ec/bitstream/3317/13368/1/T-UCSG-PRE-TEC-ITEL-354.pdf>.

PRABHU, N., *Design And Construction Of An Rfid-Enabled Infrastructure : the next avatar of the internet*. [en línea].2013. [Consulta: 16 marzo 2022]. ISBN 9780367379186. Disponible en: <https://www.routledge.com/Design-and-Construction-of-an-RFID-enabled-Infrastructure-The-Next-Avatar/Prabhu/p/book/9780367379186>.

QUIÑONES LOPEZ, S. Desarrollo de aplicación mólo de aplicación móvil par vil para la pr a la promoción de hábit omoción de hábitos saludables fr saludables frente a la pr ente a la prevención y el contr ención y el control de enf ol de enfermedades ermedades infecciosas en niños de la localidad de Kennedy . [en línea]. 2017. [Consulta: 16 marzo 2022]. Disponible en: https://ciencia.lasalle.edu.co/cgi/viewcontent.cgi?article=1146&context=ing_automatizacion.

QUIROZ ESTRADA, A.; et al. Diseño de un sistema internet de las cosas (IoT) para el monitoreo de la presión arterial. *Revista EIA*, 2021.18(35), pp. 0-15. ISSN 1794-1237. DOI 10.24050/reia.v18i35.1474.

RED HAT ¿Qué es una API? [en línea]. 2020. [Consulta: 16 marzo 2022]. Disponible en:

<https://www.redhat.com/es/topics/api/what-are-application-programming-interfaces>.

RODÉS, J.; et al. 2007. Libro De La Salud Del Hospital Clinica De Barcelona Y La Fundacion Bbva. [en línea]. 2007. [Consulta: 17 marzo 2022]. Disponible en: https://www.fbbva.es/wp-content/uploads/2017/05/dat/DE_2007_salud_clinic_barcelona.pdf.

ROSE, K. et al. La Internet de las cosas - Una breve reseña. ,2015, pp. 5-6.

ROSERO CAIZA, G.E. *Grados De Hipertensión Arterial Y Factores De Riesgo Cardiovascular Asociados En Pacientes Hipertensos Que Acuden A La Consulta Externa Del Hospital Delfina Torres De Concha De Esmeraldas Periodo De Enero A Febrero Del 2018* [en línea].2018. Quito: PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR. [Consulta: 17 marzo 2022]. Disponible en: <http://repositorio.puce.edu.ec/bitstream/handle/22000/14983/TESIS DE GRADOS DE HTA Y FRC ASOCIADOS EN PACIENTES HIPERTENSOS QUE ACUDEN A LA CONSULTA EXTERNA .pdf?sequence=1&isAllowed=y>.

RUIZ SANCHEZ, W.R. *Redes De Sensores Inalámbricas Enfocadas A La Medicina I Pontificia Universidad Católica Del Ecuador Investigación Previa La Obtención Del Título De Maestría Tema* [En Línea]. 2016. Quito: Pontificia Universidad Católica Del Ecuador. [Consulta: 17 marzo 2022]. Disponible en: http://repositorio.puce.edu.ec/bitstream/handle/22000/11311/tesis_final_mastría.pdf?sequence=1&isAllowed=y.

SANMARTIN MENDOZA, P.; et al. Internet de las cosas y la salud centrada en el hogar. [en línea]. 2016. [Consulta: 16 marzo 2022]. Disponible en: http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S0120-55522016000200014.

SASITHARAN BALASUBRAMANIAM. Realización de la Internet de las nanocosas: desafíos, soluciones y aplicaciones — Universidad de Helsinki. [en línea]. 2013. [Consulta: 16 marzo 2022]. Disponible en: <https://researchportal.helsinki.fi/en/publications/realizing-the-internet-of-nano-things-challenges-solutions-and-ap>.

SEMTECH CORPORATION LoRa and LoRaWAN: A Technical Overview LoRa® and LoRaWAN®: A Technical Overview. [en línea], 2019. [Consulta: 17 marzo 2022]. Disponible en: https://lora-developers.semtech.com/uploads/documents/files/LoRa_and_LoRaWAN-A_Tech_Overview-Downloadable.pdf.

SETHI, P. & SARANGI, S.R. Internet of Things: Architectures, Protocols, and Applications.

Journal of Electrical and Computer Engineering, 2017. ISSN 20900155. DOI 10.1155/2017/9324035.

SHEN, W. et al. Proceedings of the 2013 IEEE 17th International Conference on Computer Supported Cooperative Work in Design (CSCWD). , 2013. pp. 1-12.

SOLA, M.C. *Internet de las Cosas. Privacidad y Seguridad* [en línea]. 2016 UNIVERSIDAD DE JAÉN. [Consulta: 16 marzo 2022]. Disponible en: https://sinbad2.ujaen.es/sites/default/files/publications/Memoria_0.pdf.

SOSA-REYNA et al. Enfoque para generar aplicaciones orientadas a servicios para IoT mediante el desarrollo dirigido por modelos. *CEUR Workshop Proceedings*, 2016.1807 December, pp. 1-10. ISSN 16130073.

TAHERIZADEH, S. et al. Monitoring self-adaptive applications within edge computing frameworks: A state-of-the-art review. *Journal of Systems and Software* [en línea],2018. 136, pp. 19-38. [Consulta: 16 marzo 2022]. ISSN 0164-1212. DOI 10.1016/J.JSS.2017.10.033. Disponible en: <https://www.sciencedirect.com/science/article/pii/S016412121730256X>.

URGILES, E. & COLCHA, D. “*Diseño E Implementación De Un Sistema De Comunicación Omron – Celular Para Medir La Presión Arterial*” [en línea]. 2015. [Consulta: 17 marzo 2022]. Disponible en: <http://dspace.esoch.edu.ec/bitstream/123456789/3794/1/98T000631.pdf>.

URUEÑA, A.; et al. Big data en salud digital. *Fundación Vodafone España* [en línea], 2016. pp. 86. [Consulta: 16 marzo 2022]. Disponible en: <https://es.slideshare.net/FtimaLopez/informe-big-data-en-salud-digital>.

VALENCIA ALTAMIRANO, D.G. “*Plan De Negocios De La Empresa Pública De Diseño Y Creación De Software Y Prestación De Servicios De Ntic (Nuevas Tecnologías De La Información Y Las Comunicaciones) En La Universidad Técnica Del Norte*” [en línea]. 2015 Universidad Técnica del Norte. [Consulta: 16 marzo 2022]. Disponible en: <https://docplayer.es/3684370-Universidad-tecnica-del-norte.html>.

VANGELISTA, L.; et al. Long-range IoT technologies: the dawn of LoRa TM. [en línea], 2015, [Consulta: 17 marzo 2022]. Disponible en: <http://www.patavinatech.com/en/>.

WIECK, L., KING, et al. Técnicas de enfermería manual ilustrado. [en línea],1989. [Consulta:

17 marzo 2022]. Disponible en: <https://dialnet.unirioja.es/servlet/libro?codigo=16653>.

YACCHIREMA VARGAS, C.D Universitat Politècnica De València Departamento De Comunicaciones Arquitectura De Interoperabilidad De Dispositivos Físicos Para El Internet De Las Cosas (Iot) Tesis Doctoral. 2019.

YOUNGBLOOD, G.M.; et al. Automation intelligence for the smart environment. *IJCAI International Joint Conference on Artificial Intelligence*, no. March 2014, 2005. pp. 1513-1514. ISSN 10450823.

ANEXOS

ANEXO A: Código usado para la programación del dispositivo biomédico en Arduino

```
#ifdef ESP32
  #include "WiFi.h"
  #include <HTTPClient.h>
#else
  #include <ESP8266WiFi.h>
  #include <ESP8266HTTPClient.h>
  #include <WiFiClient.h>
#endif

#include <Wire.h>
#include "max32664.h"
#include "BluetoothSerial.h"

#define RESET_PIN 04
#define MFIO_PIN 02
#define RAWDATA_BUFFLEN 250

BluetoothSerial SerialBT;

String ssid_pass_paciente;
char* SSID1 = "RED-WIFI";
char* PASS = "CLAVE-WIFI";

String ssid;
String pass;
String paciente;
String palabra;

const int LED1 = 12;
const int LED_BT = 13;
const int SW = 14;
char indice = 1;
int controlcalib = 0;
```

```

bool flag = false;
//SERVIDOR
const char* serverName = "http://31.170.165.164/post-data.php";
String apiKeyValue = "tPmAT5Ab3j7F9";

max32664 MAX32664(RESET_PIN, MFIO_PIN, RAWDATA_BUFFLEN);

void mfioInterruptHndlr(){
  //Serial.println("i");
}

void enableInterruptPin(){
  attachInterrupt(digitalPinToInterrupt(MAX32664.mfioPin), mfioInterruptHndlr, FALLING);
}

void loadAlgomodeParameters(){

  algomodeInitialiser algoParameters;
  /* Información de calibración del sensor: https://pdfserv.maximintegrated.com/en/an/an6921-measuring-blood-pressure-MAX32664D.pdf
  CALIBRACIÓN RECOMENDADO POR EL FABRICANTE
  */

  algoParameters.calibValSys[0] = 120;
  algoParameters.calibValSys[1] = 122;
  algoParameters.calibValSys[2] = 125;

  algoParameters.calibValDia[0] = 80;
  algoParameters.calibValDia[1] = 81;
  algoParameters.calibValDia[2] = 82;

  algoParameters.spo2CalibCoefA = 1.5958422;
  algoParameters.spo2CalibCoefB = -34.659664;
  algoParameters.spo2CalibCoefC = 112.68987;

```

```

MAX32664.loadAlgorithmParameters(&algoParameters);
}

unsigned long timing; // Variable para almacenar el punto de referencia void setup()
void setup()
{
  pinMode(LED1,OUTPUT);
  pinMode(LED_BT,OUTPUT);
  pinMode(SW,INPUT_PULLUP);

  //Serial.begin(115200);
  Serial.begin(57600);
  Serial.println();
  WiFi.begin(SSID1, PASS);
  Serial.print("Configure desde la aplicación su red WiFi");
  if (!SerialBT.begin("ESP32")) {
    Serial.println("Error activando Bluetooth - Reiniciar");
  }
  else {
    Serial.println("Bluetooth activado");
    digitalWrite(LED_BT, HIGH);
    digitalWrite(LED1, LOW);
  }
}

void loop()
{
  if(WiFi.status() == WL_CONNECTED){

    //CALIBRACIÓN
    Wire.begin();
    loadAlgomodeParameters();
    int result = MAX32664.hubBegin();
    if (result == CMD_SUCCESS){
      Serial.println("Sensorhub begin!");
    }else{

```

```

//stay here.
while(1){
  Serial.println("No es posible conectarse con el sensor");
  delay(5000);
}
}

bool ret = MAX32664.startBPTcalibration();
while(!ret){

  delay(10000);
  Serial.println("Falló la calibración, Reiniciar el dispositivo");
  int controlcalib = 1;
  digitalWrite(LED1, LOW);
  Serial.println("LEDS APAGADOS");
}

delay(1000);
ret = MAX32664.configAlgoInEstimationMode();
while(!ret){
  ret = MAX32664.configAlgoInEstimationMode();
  delay(10000);
}
Serial.println("Iniciando dispositivo..");
delay(1000);
//CICLO DE TOMA DE DATOS
for(;;){

  // CONFIGURACIÓN DE TIEMPO PARA ENVÍO DE DATOS AL SERVIDOR
  //if (millis() - timing > 900000){ // 900000 para 15 minutos
  //timing = millis();
uint8_t num_samples = MAX32664.readSamples();
if(num_samples){
  Serial.print("sys = ");
  Serial.print(MAX32664.max32664Output.sys);//Presión sistólica
  Serial.print(", dia = ");
  Serial.print(MAX32664.max32664Output.dia);//Presión diastólica

```



```

if(input == 0)
{
  delay(50);
  flag = true;
  digitalWrite(LED_BT, HIGH);
  digitalWrite(LED1, LOW);
  Serial.println("WiFi desconectado - Configurar bluetooth");
}
while(flag == true)
{
  while(SerialBT.available())
  {
    ssid_pass_paciente = SerialBT.readString();
    //Serial.write(SerialBT.read());
    Serial.println(ssid_pass_paciente);

    for (int i = 0; i < ssid_pass_paciente.length(); i++)
    {
      if (ssid_pass_paciente.substring(i, i+1) == ",")
      {
        ssid = ssid_pass_paciente.substring(0, i);
        int a = ssid.length();
        palabra = ssid_pass_paciente.substring(i+1);
        for (int j = 0; j < palabra.length(); j++)
        {
          if (palabra.substring(j, j+1) == ",")
          {
            pass = palabra.substring(0, j);
            paciente = palabra.substring(j+1);
            Serial.print("SSID = ");Serial.println(ssid);
            Serial.print("Clave = ");Serial.println(pass);
            Serial.print("Paciente= ");Serial.println(paciente);
            delay(2000);
            flag = false;

            int n1 = ssid.length();

```

```

char char_array1[n1 + 1];
strcpy(char_array1, ssid.c_str());

int n2 = pass.length();
char char_array2[n2 + 1];
strcpy(char_array2, pass.c_str());

int n3 = paciente.length();
char char_array3[n3 + 1];
strcpy(char_array3, paciente.c_str());

WiFi.begin(char_array1, char_array2);

Serial.print("Conectando..");
while (WiFi.status() != WL_CONNECTED)
{
    delay(500);
    Serial.print(".");
}
Serial.println();
Serial.print("Conectado, Dirección IP: ");
Serial.println(WiFi.localIP());
Serial.print(char_array3);
    break;
}
}
}
}
}
}
}
//digitalWrite(LED1,A);
if(WiFi.status() == WL_CONNECTED){
    digitalWrite(LED1, HIGH);
    digitalWrite(LED_BT, LOW);

}

```

```
else{
  digitalWrite(LED1, LOW);
  Serial.println("Las credenciales ingresadas del WiFi son incorrectas - Reiniciar");
}
}
}
```

ANEXO B: PHP del Servidor Web

```
<!--
MARCELA IOT
-->
<?php

$servername = "localhost";

// REPLACE with your Database name
$dbname = "iot";
// REPLACE with Database user
$username = "marcela";
// REPLACE with Database user password
$password = "UsfqN3c..";

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
```

```
//$sql = "SELECT id, fecha, PDIAS, PSIS, reading_time FROM Sensor order by reading_time desc limit 40";
```

```
//cada 15 minutos en el dia total 48 muestras y en la noche cada 30 minutos solo para calculos, se mantiene los 48: TOTAL 96
```

```
$sql = "SELECT id, fecha, PDIAS, PSIS, reading_time FROM Sensor order by reading_time desc limit 96";
```

```
$result = $conn->query($sql);
```

```
//MUESTRAS TOTALES
```

```
$sql2 = "SELECT * FROM Sensor";
```

```
if ($result2=mysqli_query($conn,$sql2)) {  
    $rowcount=mysqli_num_rows($result2);  
    // echo "NUMERO DE MUESTRAS: ".$rowcount;  
}
```

```
//SUMA DE UNA COLUMNA PRESION DIASTOLICA
```

```
$sql_qry1 = "SELECT SUM(PDIAS) AS count FROM Sensor";
```

```
$duration1 = $conn->query($sql_qry1);
```

```
while($record1 = $duration1->fetch_array()){
```

```
    $totaldias = $record1['count'];
```

```
}
```

```
$promediodias = $totaldias/$rowcount;
```

```
//echo "SUMA TOTAL DE PRESI ^sN DIAST ^sLICA: ".$totaldias;
```

```
//echo "PROMEDIO: ".$promediodias;
```

```
//SUMA DE UNA COLUMNA PRESION SISTOLICA
```

```
$sql_qry8 = "SELECT SUM(PSIS) AS count FROM Sensor";
```

```
$duration8 = $conn->query($sql_qry8);
```

```
while($record8 = $duration8->fetch_array()){
```

```
    $totalsis = $record8['count'];
```

```
}
```

```
//PROMEDIO SISTOLICA
```

```
$promediosis = $totalsis/$rowcount;
```

```
//echo "SUMA TOTAL DE PRESI ^sN SIST ^sLICA: ".$totalsis;
```

```
//echo "PROMEDIO: ".$promediosis;
```

```
//ESOOGER TODOS LOS VALORES DE LA TABLA PARA CALCULOS
```

```
while ($data2 = $result2->fetch_assoc()){
```

```
    $sensor_data2[] = $data2;
```

```
}
```

```
//FIN
```

```
while ($data = $result->fetch_assoc()){
```

```
    $sensor_data[] = $data;
```

```
}
```

```
$readings_time = array_column($sensor_data, 'reading_time');
```

```
$readings_time2 = array_column($sensor_data2, 'reading_time');
```

```
//SEPARAR FECHA
```

```
//ESOOGER TODOS LOS VALORES DE LA TABLA PARA CALCUL
```

```
//FIN
```

```
//SEPARAR fin
```

```
// ***** Uncomment to convert readings time array to your timezone *****
```

```
$i = 0;
```

```
foreach ($readings_time as $reading){
```

```
    // Uncomment to set timezone to - 1 hour (you can change 1 to any number)
```

```
    $readings_time[$i] = date("Y-m-d H:i:s", strtotime("$reading - 5 hours"));
```

```
// Uncomment to set timezone to + 4 hours (you can change 4 to any number)
```

```
    //$readings_time[$i] = date("Y-m-d H:i:s", strtotime("$reading + 4 hours"));
```

```
    $i += 1;
```

```
}
```

```
$PSISSEP = array_column($sensor_data2, 'PSIS');
```

```
$PDIASSEP = array_column($sensor_data2, 'PDIAS');
```

```
$nompaciente = array_reverse(array_column($sensor_data2, 'fecha'));
```

```
$k = 0;
```

```
$l = 0;
```

```
$m = 0;
```

```
$contadorsis1=0;
```

```
$contadorsis2=0;
```

```
$contadorsis3=0;
```

```
$contadordias1=0;
```

```
$contadordias2=0;
```

```

$contadordias3=0;

foreach ($readings_time2 as $reading){
    // HORA SEPARADA PARA EL DIA Y NOCHE
    $readings_time2[$k] = date("H", strtotime("$reading - 5 hours"));
    $readings_time3[$k] = date("Y-m-d H:i:s", strtotime("$reading - 5 hours"));
    if (($readings_time2[$k] >= "7") and ($readings_time2[$k] <= "22")) {
        $readings_manana[$l] = $readings_time2[$k];
        $sismanana[$l] = $PSISSEP[$k];
        $diasmanana[$l] = $PDIASSEP[$k];
        //OBTENCION DE LA CARGA - PORCENTAGE
        if (($sismanana[$l] >= "140") and ($sismanana[$l] <= "159")) {
            $contadorsis1=$contadorsis1+1;
        }
        else
        if (($sismanana[$l] >= "160") and ($sismanana[$l] <= "179")) {
            $contadorsis2=$contadorsis2+1;
        }else
        if ($sismanana[$l] >= "180") {
            $contadorsis3=$contadorsis3+1;
        }else
        if (($diasmanana[$l] >= "90") and ($diasmanana[$l] <= "99")) {
            $contadordias1=$contadordias1+1;
        }else
        if (($diasmanana[$l] >= "100") and ($diasmanana[$l] <= "109")) {
            $contadordias2=$contadordias2+1;
        }else
        if ($diasmanana[$l] >= "110") {
            $contadordias3=$contadordias3+1;
        }else
        //FIN PORCENTAGE
        $l =$l+1;
    }
    else{
        $readings_noche[$m] = $readings_time2[$k];
        $sisnoche[$m] = $PSISSEP[$k];
        $diasnoche[$m] = $PDIASSEP[$k];

        $m =$m+1;
    }
}

```

```

    $k = $k+1;
}
//VALORES CADA 30 MINS DE LA NOCHE

$cont_noche=count($sisnoche);
$tut=0;
for ($jv=0; $jv <= $cont_noche_sis ; $jv++) {
if ($sisnoche[$jv]%2==0) {
$sisnoche_ok[$tut]=$sisnoche[$jv];
$diasnoche_ok[$tut]=$diasnoche[$jv];

//OBTENCION DE LA CARGA - PORCENTAGE
if (($sisnoche_ok[$tut] >= "140") and ($sisnoche_ok[$tut] <= "159")) {
$contadorsis1=$contadorsis1+1;
}else
if (($sisnoche_ok[$tut] >= "160") and ($sisnoche_ok[$tut] <= "179")) {
$contadorsis2=$contadorsis2+1;
}else
if ($sisnoche_ok[$tut] >= "180") {
$contadorsis3=$contadorsis3+1;
}else
if (($diasnoche_ok[$tut] >= "90") and ($diasnoche_ok[$tut] <= "99")) {
$contadordias1=$contadordias1+1;
}else
if (($diasnoche_ok[$tut] >= "100") and ($diasnoche_ok[$tut] <= "109")) {
$contadordias2=$contadordias2+1;
}else
if ($diasnoche_ok[$tut] >= "110") {
$contadordias3=$contadordias3+1;
}
//FIN PORCENTAGE
$tut=$tut+1;
}
}

//DIAGNOSTICO - CARGA
$sis_c1= round((( $contadorsis1 * 100) / 72));
$sis_c2= round((( $contadorsis2 * 100) / 72));
$sis_c3= round((( $contadorsis3 * 100) / 72));

$dias_c1= round((( $contadordias1 * 100) / 72));

```



```

$dias_c2= round((( $contadordias2*100)/72));
$dias_c3= round((( $contadordias3*100)/72));

$uu = 0;
//echo "NOMBRE ".$fecha;
$fecha2 = array_reverse(array_column($sensor_data, 'fecha'));
$PSISMAX = array_reverse(array_column($sensor_data2, 'PSIS'));
$PDIASMAX = array_reverse(array_column($sensor_data2, 'PDIAS'));

$nombrp = end($fecha2);
// ESCOGER VALORES SOLO DEL ULTIMO USUARIO
$fecha = json_encode(array_reverse(array_column($sensor_data, 'fecha')), JSON_NUMERIC_CHECK);
$PDIAS = json_encode(array_reverse(array_column($sensor_data, 'PDIAS')),
JSON_NUMERIC_CHECK);
$PSIS = json_encode(array_reverse(array_column($sensor_data, 'PSIS')), JSON_NUMERIC_CHECK);
$reading_time = json_encode(array_reverse($readings_time), JSON_NUMERIC_CHECK);
$PSISMIN = array_filter($PSISMAX);
$PDIASMIN = array_filter($PDIASMAX);
$rowcount3 = count($PSISMIN);

$maxsis = max($PSISMAX);
$minsis = min($PSISMIN);
$maxdias = max($PDIASMAX);
$mindias = min($PDIASMIN);

$sismanana2 = max($sismanana);
$sisnoche2 = max($sisnoche);
$diasmanana2 = max($diasmanana);
$diasnoche2 = max($diasnoche);
//ESCOGER MINIMOS SIN CEROS
$sismananatem = array_filter($sismanana);
$sisnochetemp = array_filter($sisnoche);
$diasmananatem = array_filter($diasmanana);
$diasnochetemp = array_filter($diasnoche);
//FIN
$sismanana3 = min($sismananatem);
$sisnoche3 = min($sisnochetemp);
$diasmanana3 = min($diasmananatem);
$diasnoche3 = min($diasnochetemp);

```

```

$horasep = max($readings_manana);
$fechasep = max($readings_noche);

//PROMEDIOS

$scant_sismanana = count($sismananatemp);
$scant_sisnoche = count($sisnochetemp);
$scant_diasmanana = count($diasmananatemp);
$scant_diasnoche = count($diasnochetemp);

$suma_1 = array_sum($sismanana);
$suma_2 = array_sum($sisnoche);
$suma_3 = array_sum($diasmanana);
$suma_4 = array_sum($diasnoche);

$prosismanana = round(( $suma_1/$scant_sismanana), 2);
$prosisnoche = round(($suma_2/$scant_sisnoche), 2);
$prodiasmanana = round(($suma_3/$scant_diasmanana), 2);
$prodiasnoche = round(($suma_4/$scant_diasnoche), 2);
$h_inicio = $readings_time3[0];
$h_fin = $readings_time3[$k-1];
//echo $fecha;
/*echo $PDIAS;
echo $PSIS;
echo $reading_time;*/

$result->free();
$conn->close();
?>

<!DOCTYPE html>
<html>
<meta name="viewport" content="width=device-width, initial-scale=1">
<script src="https://code.highcharts.com/highcharts.js"></script>
<style>
body {
min-width: 310px;
max-width: 1280px;
height: 500px;

```

```

margin: 0 auto;
}
h2 {
font-family: Arial;
font-size: 2.5rem;
text-align: center;
}
</style>
<body>
<h2>MONITOR DE PRESI ^sN ARTERIAL - MAPA</h2>
<h3>PACIENTE : <?php echo $nombrep ?> </h3>
<h3>Hora De Inicio: <?php echo $h_inicio ?> Hora De Finalizaci n: <?php echo $h_fin ?> </h3>
<h3> MEDICIONES </h3>
<h3> Muestras Totales: <?php echo $rowcount ?> Muestras V lidas : <?php echo $rowcount3 ?>
</h3>
<h3> VALORES M ^aXIMOS </h3>
<h3> Sist lica Ma ana : <?php echo $sismanana2?> Sist lica Noche : <?php echo $sisnoche2?>
Total : <?php echo $maxsis?> </h3>
<h3> Diast lica Ma ana : <?php echo $diasmanana2?> Diast lica Noche : <?php echo $diasnoche2?>
Total : <?php echo $maxdias ?> </h3>
<h3> M ^mNIMOS </h3>
<h3> Sit lica Ma ana : <?php echo $sismanana3?> Sist lica Noche : <?php echo $sisnoche3?> Total :
<?php echo $minsisis?> </h3>
<h3> Diast lica Ma ana : <?php echo $diasmanana3?> Diast lica Noche : <?php echo $diasnoche3?>
Total : <?php echo $mindias ?> </h3>
<h3> PROMEDIOS </h3>
<h3> Sist lica Ma ana : <?php echo $prosismanana?> Sist lica Noche : <?php echo $prosisnoche?> </h3>
<h3> Diast lica Ma ana : <?php echo $prodiasmanana?> Diast lica Noche : <?php echo
$prodiasnoche?></h3>
<h3> DIAGN ^sSTICO </h3>
<h3> CARGA SIST ^sLICA : Tipo 1= <?php echo $sis_c1 ?>% Tipo 2= <?php echo $sis_c2 ?>% Tipo
3= <?php echo $sis_c3 ?>% </h3>
<h3> CARGA DIAST ^sLICA : Tipo 1= <?php echo $dias_c1 ?>% Tipo 2= <?php echo $dias_c2 ?>%
Tipo 3= <?php echo $dias_c3 ?>% </h3>
<div id="chart-temperature" class="container"></div>
<div id="chart-humidity" class="container"></div>
<div id="chart-pressure" class="container"></div>
<script>

var fecha = <?php echo $fecha; ?>;

```

```

var PDIAS = <?php echo $PDIAS; ?>;
var PSIS = <?php echo $PSIS; ?>;
var reading_time = <?php echo $reading_time; ?>;

var chartT = new Highcharts.Chart({
  chart:{ renderTo : 'chart-temperature' },
  title: { text: 'VALORES DE PRESI ^sN DIAST ^sLICA' },
  series: [{
    showInLegend: false,
    data: PDIAS
  }],
  plotOptions: {
    line: { animation: false,
      dataLabels: { enabled: true }
    },
    series: { color: '#059e8a' }
  },
  xAxis: {
    type: 'datetime',
    categories: reading_time
  },
  yAxis: {
    title: { text: 'PRESION DIAST ^sLICA' }
    //title: { text: 'Temperature (Fahrenheit)' }
  },
  credits: { enabled: false }
});

```

```

var chartH = new Highcharts.Chart({
  chart:{ renderTo:'chart-humidity' },
  title: { text: 'VALORES DE PRESI ^sN SIST ^sLICA' },
  series: [{
    showInLegend: false,
    data: PSIS
  }],
  plotOptions: {
    line: { animation: false,
      dataLabels: { enabled: true }
    }
  },
  },

```

```
xAxis: {
type: 'datetime',
  //dateTimeLabelFormats: { second: '%H:%M:%S' },
  categories: reading_time
},
yAxis: {
  title: { text: 'PRESION SIST ^sLICA' }
},
credits: { enabled: false }
});
</script>
```

```
<div class="row">
  <div class="col text-center">
    <a href="pdf.php">
      DESCARGAR TODA LA INFORMACION EN FORMATO XLS
    </a>
  </div>
</div>
</div>
</body>
</html>
```

ANEXO C: Información del Sensor

```
<?php
/*
MARCELA IOT
*/

$servername = "localhost";

// Database
$dbname = "iot";
// Database user
$username = "marcela";
// Database user password
$password = "UsfqN3c..";

// API Key compatible con el ESP32
$api_key_value = "tPmAT5Ab3j7F9";

$api_key = $fecha = $PDIAS = $PSIS = "";

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $api_key = test_input($_POST["api_key"]);
    if($api_key == $api_key_value) {
        $fecha = test_input($_POST["fecha"]);
        $PDIAS = test_input($_POST["PDIAS"]);
        $PSIS = test_input($_POST["PSIS"]);

        // Coneccion
        $conn = new mysqli($servername, $username, $password, $dbname);
        // Si error
        if ($conn->connect_error) {
            die("Fallo la conexion" . $conn->connect_error);
        }

        $sql = "INSERT INTO Sensor (fecha, PDIAS, PSIS)
VALUES ('" . $fecha . "', '" . $PDIAS . "', '" . $PSIS . "')";
```

```
if ($conn->query($sql) === TRUE) {
    echo "Creado correctamente";
}
else {
    echo "Error: " . $sql . "<br>" . $conn->error;
}

$conn->close();
}
else {
    echo "API Key erronea";
}

}
else {
    echo "Ningun valor posteadado con HTTP POST.";
}

function test_input($data) {
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}
```