



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CARRERA DE INGENIERÍA EN ELECTRÓNICA Y REDES DE COMUNICACIÓN

**“SISTEMA EMBEBIDO INTELIGENTE DE REPRESENTACIÓN DE DATOS
BIOMÉDICOS PARA EL ANÁLISIS DE LA APNEA”**

**TRABAJO DE GRADO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERIA
EN ELECTRÓNICA Y REDES DE COMUNICACIÓN**

AUTOR: JHONNY GERARDO IPIALES MATANGO

DIRECTOR: ING. JAIME ROBERTO MICHILENA CALDERÓN MSc.

ASESOR: ING. CARLOS ALBERTO VÁSQUEZ AYALA MSc.

ASESOR: ING. EDGAR DANIEL JARAMILLO VINUEZA

Ibarra-Ecuador

2021



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE.

IDENTIFICACIÓN DE LA OBRA.

En cumplimiento del Art. 144 de la ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información.

DATOS DEL CONTACTO	
CÉDULA DE IDENTIDAD	1003557699
APELLIDOS Y NOMBRES	Ipiales Matango Jhonny Gerardo
DIRECCIÓN	Barrio Santa Rosa del Tejar; Av. Santa Rosa del Tejar
E-MAIL	jgipialesm@utn.edu.ec
TELÉFONO MÓVIL	0986855850
DATOS DE LA OBRA	
TEMA	“Sistema embebido inteligente de representación de datos biomédicos para el análisis de la Apnea”
AUTOR	Jhonny Gerardo Ipiales Matango
FECHA	11 de noviembre del 2021
PROGRAMA	PREGRADO <u>X</u> POSTGRADO <u> </u>
TÍTULO	Ingeniero en Electrónica y Redes de Comunicación
DIRECTOR	Ing. Jaime Michilena MSc.



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

CONSTANCIA

El autor manifiesta que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto, la obra es original y que es el titular de los derechos patrimoniales, por lo que asume la responsabilidad sobre el contenido de la misma y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 12 días del mes de Noviembre del 2021

Jhonny Gerardo Ipiates Matango

C.I.: 100355769-9



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CERTIFICACIÓN

ING. JAIME MICHILENA MSc, DIRECTOR DEL PRESENTE TRABAJO DE TITULACIÓN

CERTIFICA

Que, el presente Trabajo de Titulación “SISTEMA EMBEBIDO INTELIGENTE DE REPRESENTACIÓN DE DATOS BIOMÉDICOS PARA EL ANÁLISIS DE LA APNEA” ha sido desarrollado por el señor Jhonny Gerardo Ipiales Matango bajo mi supervisión.

Es todo en cuanto puedo certificar en honor a la verdad.

Ing. Jaime Roberto Michilena Calderón MSc.

DIRECTOR



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

AGRADECIMIENTO

Agradezco a Dios por la vida y por la gran familia que me ha otorgado, gracias por guiarme y protegerme siempre, por esta fortaleza para superar los obstáculos, por estas ganas para luchar por lo que quiero y sobre todo por esos innumerables momentos de felicidad juntos a mis seres queridos.

A mi familia, quienes siempre estuvieron en los buenos y malos momentos, gracias a mis padres Luis Gerardo Ipiales y María Ubaldina Matango por su gran apoyo, esfuerzo, comprensión, dedicación y sobre todo por ese gran amor que durante toda mi vida he recibido. Gracias por que sin ustedes nada de esto tendría sentido ni hubiese sido posible.

De todo corazón aquella mujer especial, a quien adoro y por más de 6 años ha sido una persona incondicional en mi vida, ha sido mi soporte, mi mejor amiga, mi consejera, mi apoyo, mi luz, mi guía, mi todo para seguir adelante y no bajar los brazos en los momentos difíciles, por ser la mujer que Dios me presentó en la vida y sobre todo por su gran amor y paciencia.

A mis directores del presente proyecto, Ing. Jaime Michilena, Ing. Carlos Vásquez, Ing. Paul Rosero y el Ing. Daniel Jaramillo por los consejos y guías durante el desarrollo del tema de tesis.

Finalmente agradezco a mis amigos de carrera por su compañía, esfuerzo y apoyo.

Jhonny Gerardo Ipiales Matango



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

DEDICATORIA

Han pasado muchos años desde que nací, y desde ese momento e incluso antes que eso, ya estaban buscando maneras de ofrecerme lo mejor. Han trabajado duro, y sin importar si llegasen cansados de su trabajo siempre tenían una sonrisa que ofrecer a su familia. La ayuda que me han brindado han construido bases de gran importancia y ahora más que nunca soy consciente de eso.

Por ello Dedico mi tesis a Dios a quién amo y admiro; a mis extraordinarios Padres, Luis Gerardo Ipiales y María Ubaldina Matango por su noble dedicación y amor, por ser mis amigos, mis consejeros, y por siempre guiarme y ser la voz y bendición de Dios como prioridad en mi vida.

Madre, Tus esfuerzos son impresionantes y tu amor es para mí invaluable. Junto con mi padre me has educado, me has proporcionado todo y cada cosa que he necesitado. Tus enseñanzas las aplico cada día y por ello tengo mucho por agradecerte.

Padre, por siempre desear y anhelar siempre lo mejor para mi vida, gracias por cada consejo y por cada una de sus palabras que me guiaron durante mi vida.

Jhonny Gerardo Ipiales Matango

ÍNDICE

IDENTIFICACIÓN DE LA OBRA.....	I
CERTIFICACIÓN	III
AGRADECIMIENTO	IV
DEDICATORIA	V
INDICE DE ECUACIONES	XVI
RESUMEN	XVII
ABSTRACT.....	XVIII
CAPITULO I. Antecedentes.....	1
1.1. Tema.....	1
1.2. Problema.....	1
1.3. Objetivos	3
1.3.1. General.....	3
1.3.2. Específicos	3
1.4. Alcance.....	3
1.5. Justificación.....	5
CAPITULO II. MARCO TEÓRICO	7
2.1. Problemas del sueño.....	7
2.1.1. Insomnio	7
2.1.2. Trastornos respiratorios	8
2.1.3. Hipersomnias	9

2.1.4. Alteraciones del ritmo circadiano.....	9
2.1.5. Parasomnias.....	10
2.1.6. Movimientos anormales relacionados con el sueño	11
2.1.7. Síntomas aislados	11
2.1.8. Apnea.....	12
2.2. Bioseñales.....	12
2.2.1. Tipos de bioseñales.....	13
2.3. Adquisición de datos	16
2.3.1. Adquisición Mediante ADC	18
2.3.2. Técnicas de Filtrado.....	20
2.3.3. Amplificadores	21
2.4. Sensores.....	22
2.4.1. Según el principio de funcionamiento	22
2.4.2. Según el tipo de señal eléctrica que generan	22
2.4.3. Según el rango de valores que proporcionan.....	23
2.4.4. Según el nivel de integración.....	23
2.4.5. Según el tipo de variable física medida.....	23
2.5. Sistemas inteligentes	25
2.6. Aprendizaje de Maquina	26
2.6.1. Análisis de Datos.....	27
2.6.2. Técnicas de Análisis de Datos.....	28

2.6.3. Algoritmos de Clasificación	29
2.7. Análisis de sensado de Apnea	31
2.7.1. Sensor SPO2	34
2.7.2. Sensor de Detección de Posición del Cuerpo	35
2.7.3. Sensor de Detección de Ronquido.....	35
2.8. Almacenamiento de datos	36
2.8.1. Almacenamiento definido por software	36
2.8.2. Almacenamiento en la nube	36
2.8.3. Almacenamiento adjunto a la red	37
2.8.4. Almacenamiento de objetos.....	37
2.8.5. Almacenamiento en bloques.....	37
2.9. Modelos de desarrollo	37
2.9.1. Modelo lineal.....	38
2.9.2. Modelo en cascada.....	38
2.9.3. Modelo en V	39
CAPÍTULO III. DESARROLLO EXPERIMENTAL.....	40
3.1. Análisis general	40
3.2. Metodología	40
3.2.1. Situación actual.....	42
3.2.2. Encuesta de requerimientos	44
3.2.3. Stakeholders.....	46

3.3. Requerimientos del sistema.....	48
3.3.1. Requerimientos iniciales del sistema.....	48
3.3.2. Requerimiento de arquitectura.....	51
3.3.3. Selección de hardware y software	54
3.4. Diseño global del sistema.....	59
3.4.1. Descripción general del sistema	59
3.4.2. Arquitectura.....	60
3.5. Diseño.....	63
3.5.1. Bloque de acoplamiento y acondicionamiento de la señal. (Bloque 1).....	63
3.5.2. Bloque de sensores (Bloque 2).	74
3.5.2.1. Hardware.	74
3.5.3. Bloque de procesamiento (Bloque 3)	83
3.5.4. Bloque de Alimentación. (Bloque 4).....	85
3.5.5. Bloque transmisión de datos. (Bloque 5).....	88
3.5.6. Bloque representación de datos. (Bloque 6)	89
3.5.7. Diseño general del sistema.	93
3.5.8. Diagrama de flujo general del sistema.	95
3.6. Implementación.....	98
3.6.1. Implementación de armazón protector del sistema.	98
3.6.2. Bloque de acoplamiento de la señal.	99
3.6.3. Bloque de sensores.	100

3.6.4. Bloque de alimentación.	102
3.6.5. Bloque de transmisión.	102
3.6.6. Bloque de procesamiento de datos.	103
3.6.7. Bloque de Representación de datos.	106
3.7. Implementación final de sistema.....	107
3.8. Ubicación del prototipo.....	108
CAPÍTULO IV. PRUEBAS DE FUNCIONAMIENTO Y RESULTADOS	110
4.1. Resultados	112
4.1.1. Pruebas de funcionamiento del sistema.....	112
4.1.2. Pruebas generales en usuarios	118
4.1.2.1. Prueba con características biomédicas de usuario 1.....	120
4.1.2.2. Prueba con características biomédicas de usuario 2.....	121
4.1.2.3. Prueba con características biomédicas de usuario 3.....	122
4.1.2.4. Prueba con características biomédicas de usuario 4.....	123
4.1.2.5. Prueba con características biomédicas de usuario 5.....	124
4.1.2.6. Prueba con características biomédicas de usuario 6.....	125
4.2 Prueba validación funcional de sensor Oxímetro.....	126
4.3. Matriz de confusión.....	127
4.5. Valores máximo y mínimos obtenidos de la toma de datos.....	129
4.6 Test de pruebas basadas en el modelo en V	129
4.6.1. Test Eléctrico.....	130

4.6.2. Test Subsistema pasivo.....	131
4.6.3. Test Subsistema activo	132
4.6.4. Test Subsistema Hardware	133
4.6.5. Test Subsistema Software.....	134
CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES	135
5.1. Conclusiones	135
5.2. Recomendaciones.....	136
BIBLIOGRAFIA	137
ANEXOS	142
ANEXO 1. Tabla de nivel de confianza de encuesta (distribución normal).....	142
ANEXO 2. Formato de encuesta para requerimientos de usuarios.....	143
ANEXO 3. Programación script ide-arduino para la lectura de sensor de oximetría .	145
ANEXO 4. Programación script ide-arduino para la lectura de sensor acelerómetro.	149
ANEXO 3. Programación script ide-arduino para la lectura de sensor respiración....	153
ANEXO 4. Programación script ide-arduino para la lectura de sensor ronquido.	156
ANEXO 5. Señal obtenida por serial plotter de arduino del prototipo de oximetría. .	159
ANEXO 6. Manual de usuario funcionamiento del sistema inteligente.	160
ANEXO 7. Especificaciones técnicas de Node MCU.....	165
ANEXO 8. Características de sensor SPO2	168
ANEXO 9. Características de sensor de posición del cuerpo	172
ANEXO 10. Características del sensor de medición de ronquido	174

INDICE DE FIGURAS

Figura 1. Proceso de adquisición de la señal	17
Figura 2. Clasificación de los sensores	25
Figura 3. Proceso para el análisis de datos	27
Figura 4. Diagrama del proceso KDD	28
Figura 5. Etapas del modelo lineal.....	38
Figura 6. Etapas del modelo en cascada	39
Figura 7. Modelo en V	39
Figura 8. Estructura modelo en V	41
Figura 9. Diagrama de bloques general	61
Figura 10. Señal sin amplificador	64
Figura 11. Esquema de amplificador de señal	66
Figura 12. Diagrama de amplificador de audio	69
Figura 13: Distribución de pines de Mux analógico.....	70
Figura 14: Filtro lógico (Average Moving)	71
Figura 15: señal captada sin filtro.....	71
Figura 16: Código multiplexor analógico	73
Figura 17. Diagrama de flujo del bloque 1	74
Figura 18. Diagrama de conexión de posición del cuerpo.....	75

Figura 19. Diagrama de conexión sensor de OXIMETRIA	77
Figura 20. Diagrama de conexión interno SPO2	78
Figura 21. Diagrama de conexión de sensor de ronquido.....	79
Figura 22. Diagrama de conexión de sensor de flujo de aire.....	80
Figura 23. Lectura de datos a través de sensores	81
Figura 24. Diagrama de flujo de bloque 2	82
Figura 25: Código de bloque de procesamiento	84
Figura 26. Diagrama de conexión de alimentación	87
Figura 27. Diagrama de flujo del bloque 3	88
Figura 28. Programa de funcionamiento de módulo WiFi	89
Figura 29: Código de envío desde Arduino	92
Figura 30. Representación de datos recibidos por Cayenne en tiempo real	93
Figura 31. Diagrama circuital del sistema	94
Figura 32. Diagrama de flujos del sistema.....	97
Figura 33. Diagrama de chasis de prototipo	98
Figura 34: Diseño e impresión de amplificador en placa	99
Figura 35. Señal de sensor acelerómetro	100
Figura 36. Diseño e impresión de placa.....	101
Figura 37. Implementación y acoplamiento de sensores a placa	101
Figura 38. Implementación de Fuente de alimentación	102
Figura 39. Implementación de código para módulo WiFi	103

Figura 40. BDD obtenida por el sistema.....	104
Figura 41. Porcentaje de reducción.....	104
Figura 42. Interfaz de visualización de datos.....	106
Figura 43. Diseño de Prototipo	107
Figura 44. Diseño y presentación final	108
Figura 45: Ubicación de sensores en el cuerpo.....	109
Figura 46. Etapas y ciclos del sueño.....	111
Figura 47: Medición de voltaje de funcionamiento de sistema	112
Figura 48: Indicadores y/o Luces de encendido	113
Figura 49: Funcionamiento de filtro lógico	114
Figura 50: Evaluación de monitoreo de variables.....	115
Figura 51: Confirmación de envío a plataforma	116
Figura 52: Datos recibidos desde el sistema de monitoreo.....	116
Figura 53. Datos obtenidos de usuario 1.....	120
Figura 54: Datos obtenidos del usuario 2	121
Figura 55. Datos obtenidos del usuario 3.....	122
Figura 56. Datos obtenidos del usuario 4.....	123
Figura 57. Datos obtenidos de usuario 5.....	124
Figura 58: Datos biomédicos del usuario 6.....	125
Figura 59. Valores de saturación de oxígeno y frecuencia cardiaca	126
Figura 60: Señal final obtenida por el sistema.....	159

INDICE DE TABLAS

Tabla 1. Bioseñales relevantes del cuerpo humano	13
Tabla 2. Porcentajes de niveles de oxígeno en la sangre.	32
Tabla 3. Frecuencia respiratoria.....	33
Tabla 4. Abreviaturas de requerimientos	43
Tabla 5. Prioridad de los Requerimientos.....	44
Tabla 6. Lista de stakeholders.....	46
Tabla 7. Requerimientos de stakeholders	47
Tabla 8. Requerimientos iniciales del sistema.	49
Tabla 9. Requerimientos de arquitectura	52
Tabla 10. Elección del sistema embebido.....	54
Tabla 11. Selección de sensor SPO2.....	55
Tabla 12. Selección de sensor de posición del cuerpo.....	56
Tabla 13. Selección de sensor de ronquido.....	57
Tabla 14. Elección del software.....	58
Tabla 15. Asignación de valores a cada posición.	75
Tabla 16. Comparación de plataformas de recepción de datos.....	90
Tabla 17. Algoritmos de aprendizaje	105
Tabla 18: Niveles de referencia de bioseñales de monitoreo.....	110
Tabla 19: Chech list de Pruebas de funcionamiento	117

Tabla 20. Simbología de resultados	118
Tabla 21. Características de usuarios sometidos a monitoreo	119
Tabla 22. Comparativa entre diseño comercial y prototipo	127
Tabla 23. Interpretación de matriz de confusión.	127
Tabla 24. Matriz de confusión	128
Tabla 25. Valores máximos y mínimos del sistema	129
Tabla 26. Especificaciones técnicas de Node MCU	165
Tabla 27. Características de sensor SPO2.....	168
Tabla 28. Características de sensor de posición del cuerpo	172
Tabla 29. Características del sensor de medición de ronquido.	174

INDICE DE ECUACIONES

Ecuación 1. Calculo de tamaño de muestra	45
Ecuación 2. Ecuación de tensión de salida	64
Ecuación 3. Ecuación para frecuencia de corte	65
Ecuación 4. Ecuación ley de ohm	67
Ecuación 5. Ecuación ganancia del circuito	67
Ecuación 6. Ecuación filtro pasa alto.....	68
Ecuación 7. Ley de OHM	68
Ecuación 8. Formula Moving Average	72
Ecuación 9: Suma total de requerimientos de sensores	86

RESUMEN

El presente proyecto se enfoca en el desarrollo de un sistema embebido inteligente de representación de datos biomédicos para el análisis de la apnea, cuya función es monitorear las bioseñales de las personas con el fin de detectar si el paciente posee una tendencia a padecer apnea. Además, el sistema envía los datos recolectados a la plataforma “Cayenne” para finalmente representarlos.

Los requerimientos del proyecto son muy importantes para realizar la elección de los elementos utilizados en el sistema y tomando en cuenta las ventajas del uso de hardware y software libre, se selecciona sensores dedicados al monitoreo de las bioseñales del cuerpo. Además, consta de un método de visualización, a través del cual se muestra información sobre las bioseñales y el resultado obtenido por el análisis del prototipo.

El sistema será adherido al cuerpo de manera adecuada para evitar errores de lectura y en consecuencia errores de resultado. El prototipo al estar expuesto a diferentes usuarios, debe soportar ciertas manipulaciones externas. Por este motivo, el proyecto se desarrolla basado en la metodología del modelo en V. De esta manera, garantizando un adecuado proceso en el diseño e implementación del sistema.

ABSTRACT

This project focuses on the development of an intelligent embedded system for the representation of biometric data for the analysis of apnea, whose function is to monitor the biosignals of people in order to detect if the patient has a tendency to have apnea. In addition, the system sends the collected data to the “Cayenne” platform to finally represent them.

The requirements of the project are very important to make the choice of the elements used in the system and taking into account the advantages of using free Hardware and Software, sensors dedicated to monitoring the body's biosignals are selected. In addition, it consists of a visualization method, through which information about the biosignals and the result obtained by the analysis of the prototype are displayed.

The system will be properly adhered to the body to avoid reading errors and consequently result errors. The prototype, being exposed to different users, must withstand certain external manipulations. For this reason, the project is developed based on the methodology of the model in v. in this way, guaranteeing an adequate process in the design and implementation of the system.

CAPITULO I. Antecedentes

En el presente capítulo se definirá el problema identificado, que darán paso a la formulación de los objetivos y el alcance de esta tesis. Finalmente, se presenta la justificación para la realización del presente proyecto.

1.1.Tema

SISTEMA EMBEBIDO INTELIGENTE DE REPRESENTACIÓN DE DATOS BIOMÉDICOS PARA EL ANÁLISIS DE LA APNEA

1.2.Problema

Las personas con problemas de salud comúnmente sufren de trastornos del sueño, los cuales afectan directamente a las acciones y actitudes de cada individuo. En algunos casos particulares, estos trastornos pueden ocasionar en la muerte súbita en neonatos. Según investigación de Seung Lee Y., Kye S., Moon J., Lee T., Lee S. and Lee K. (2017) “Los individuos con trastornos del sueño, como la apnea y el insomnio experimentan conductas anormales con más frecuencia que las personas sin dichos trastornos”. A pesar de que este problema fue detectado hace mucho tiempo atrás, no es sino hasta la última década en donde se lo estudia debido a su incremento y su gran impacto en la salud y desarrollo en los niños.

Por otro lado argumenta Olivi H. (2013) que “el desarrollo de los síntomas de este trastorno es lento y progresivo”, por ende dichos síntomas pueden generar graves consecuencias de no ser detectadas y tratadas a tiempo. Para ello es necesario realizar un análisis exhaustivo del

trastorno del sueño denominado apnea para de esta manera detectar y prevenir posibles complicaciones.

Actualmente, el único método clínicamente aprobado para su diagnóstico es la polisomnografía (PSG por su abreviación) que examina el sueño de las personas en un laboratorio con varios sensores fisiológicos unidos al cuerpo. Este método únicamente se realiza en clínicas y hospitales, por lo que se requiere que los pacientes duerman en un laboratorio con varios sensores unidos a su cuerpo durante varias noches. Ya que las personas muestran una variabilidad nocturna en sus síntomas. En consecuencia, dormir en un entorno diferente puede perturbar el comportamiento natural de sueño del paciente y por ende el estudio se vería afectado. Según estudios realizados por Olivi H. (2013) “La apnea constituye la tercera enfermedad respiratoria más frecuente después del asma bronquial y la enfermedad pulmonar obstructiva crónica. Este síndrome se asocia a trastornos cognitivos y al deterioro de la calidad de vida y en niños afecta el aprendizaje y la conducta”. Se ha comprobado que las personas con apnea pueden presentar hipertensión arterial sistémica, hipertensión pulmonar, enfermedad vascular cerebral y generar un aumento en la tasa de accidentes, por lo cual es imperativo su diagnóstico y tratamiento.

En Ecuador existe una clínica especialista en los trastornos del sueño la cual usa los métodos antes mencionados y por ende sus limitantes en cuanto a la exactitud de los parámetros influyentes en un paciente. Por este motivo se proyecta a futuro implementar un diagnóstico con un sistema embebido inteligente de análisis de los parámetros del apnea que más influyen en el desarrollo y desempeño de una persona, el mismo que será estudiado en su entorno natural, mediante el cual se pueda obtener los parámetros lo más cercanos a la realidad.

1.3. Objetivos

1.3.1. General

Desarrollar un sistema embebido inteligente de representación de datos biomédicos para el análisis de la apnea.

1.3.2. Específicos

- Identificar las variables intrínsecas más influyentes del trastorno del sueño para la determinación de los parámetros del sistema.
- Proponer un esquema de adquisición y almacenamiento de datos biomédicos que represente el conjunto de variables previamente establecidas.
- Implementar un algoritmo de aprendizaje de máquina que represente a la apnea en forma de señales biomédicas para su correcta estimación.

1.4. Alcance

Para realizar el diseño del sistema embebido inteligente, es necesario el análisis de los fundamentos teóricos respecto a los parámetros más importantes del trastorno denominado apnea. Por ende, entender el propósito con el cual será creado el presente sistema y sus ventajas enfocados a la mejora de la calidad de vida, debido a que en la actualidad se han encontrado posibles casos de trastornos del sueño y no han sido detectados previamente. Estos no se encuentran en ningún tipo de tratamiento *ad hoc*, ya que cada caso depende de diferentes parámetros tales como: flujo respiratorio, posición del cuerpo, ronquido, presión arterial o problemas fisiológicos; por tal motivo el trastorno debe ser diagnosticado ya que cada uno de los casos desencadena un problema diferente.

Posteriormente de tener conocimiento de la teoría del trastorno se realizara la recopilación y almacenamiento de los datos previamente ya estudiados e identificados como más influyentes en la apnea, esta adquisición de datos se realizará por medio de sensores especializados en salud tales como SPO2, Sensor posición del cuerpo, Sensor flujo de aire, Sensor ronquido, Sensor presión arterial, entre otros; según se exija la necesidad y el requerimiento establecido por los parámetros antes identificados. Estos sensores tomaran varios datos de los cuales no todos serán considerados óptimos, para ello se definirá la forma de adquirir los datos por medio de lenguaje de programación y cuales específicamente se almacenaran en el sistema.

A continuación se verificará e implementará un algoritmo de aprendizaje de máquina que permita comparar los datos antes adquiridos por los sensores, se pueda detectar si el paciente tiene una tendencia a tener apnea. Este proceso se realiza mediante una representación de las señales biomédicas adquiridas por los sensores que tengan la posibilidad de generar alertas de estados de la persona.

Una vez diseñado e implementado el algoritmo de aprendizaje de maquina se procederá a realizar las pruebas en las cuales uno o más personas que padecen del trastorno y otras que no lo poseen, serán sometidas a la evaluación y de esta manera permitirán predecir las posibles complicaciones y obtener un resultado satisfactorio.

1.5. Justificación

Los problemas de desarrollo y desempeño son muy comunes en niños que presentan trastornos del sueño debido a diferentes situaciones y condiciones en cada persona en particular, a pesar de las consecuencias que estos generan, no existen estrategias para identificar que parámetros son los que más afectan al individuo, por tal motivo se ve la necesidad de realizar un análisis, de esta manera tratar y prevenir posibles complicaciones que se presenten en cada caso.

En la actualidad es posible determinar qué tipo de trastorno posee una persona por medio de una PSG. Sin embargo, este proceso se deberá llevar a cabo en un hospital o clínica. De esta manera, se afectan ciertos parámetros al encontrarse la persona en un entorno diferente al habitual de sus horas de sueño. Dado que el PSG consume mucho tiempo y es costoso para los pacientes, no es práctico controlar el sueño de las personas en su hogar, por ello este sistema embebido pretende monitorear a la persona en su entorno ideal evitando que los parámetros sean lo mínimamente afectados. Por otro lado, estos sensores poseen un reducido tamaño por ende un bajo consumo energético, costo asequible y flexibilidad si se desea agregar nuevas funcionalidades de monitoreo posteriormente.

Según la Organización Mundial de la Salud (OMS), la apnea obstructiva del sueño es un riesgo potencial para la salud, aproximadamente el 40% de la población mundial padece algún tipo de alteración del sueño, y de los cuales un 80% de los afectados no son conscientes que lo padecen. Esto agrava aún más sus consecuencias, ya que estas personas no reciben ningún tipo de tratamiento y son más propensas a dormirse y sufrir accidentes de tránsito. De esta manera, atentando no solo con la vida de la persona que sufre dicho problema sino también indirectamente a las personas en su entorno. En Ecuador muchos expertos han diseñado equipos

tales como mascarillas y sensores los cuales ayudan al tratamiento de este síndrome debido al alarmante aumento de este problema en la población. Si bien durante los últimos años en el país han aumentado centros y clínicas del sueño. No obstante todavía no es suficiente para abordar la complejidad del problema debido a que cada persona padece algún tipo de trastorno diferente, y esto aumenta cada día debido a cómo está cambiando el estilo de vida de las personas.

Con el presente proyecto se pretende determinar los parámetros antes mencionados, los cuales basados en un algoritmo de aprendizaje de maquina ayudara a la detección del trastorno y un tratamiento adecuado en cada caso, de esta manera obteniéndose como resultado una disminución en cuanto al porcentaje de afectados que padecen este problema.

CAPITULO II. MARCO TEÓRICO

En el presente capítulo, se abordan los principios básicos del trastorno del sueño determinado apnea y sus características, los métodos utilizados para tratar trastornos similares, sus efectos en los usuarios y la importancia del trastorno según la Organización Mundial de la Salud (OMS).

2.1. Problemas del sueño

Los diferentes problemas del sueño ya sea por exceso o déficit, ha desencadenado algunos cambios en el comportamiento normal de las personas, como la irritabilidad, la depresión, la ansiedad y otros aspectos muy importantes a tomar en cuenta tales como los problemas de memoria, déficit de atención y bajo rendimiento físico (Marín Agudelo, H., Franco A., Vinaccia Alpi S., Tobón S., 2008).

Conforme avanza la tecnología son más abundantes las áreas de estudio con las que se establecen conexiones entre el sueño y la salud; tal es el caso de la calidad de vida relacionada con el sueño, lo cual podría desencadenar enfermedades crónicas y a su vez contraer consecuencias adversas en diferentes niveles. En muchos estudios se establecen como simples relaciones entre los patrones del sueño y la salud, mientras que otras investigaciones las definen como problemas de los trastornos del sueño, entre las cuales las más importantes a tomar en cuenta son el insomnio, narcolepsia y apnea del sueño.

2.1.1. Insomnio

El insomnio es uno de los síntomas más comunes en la personas, este síntoma es la dificultad persistente para conciliar el sueño o por mantenerse dormido. Pero en ocasiones, la

dificultad para dormir no es una característica tan precisa para determinar dicho problema. Para ello, es necesario tomar en cuenta ciertos aspectos y molestias diurnas tales como: la fatiga, problemas de concentración, alteraciones de ánimo, somnolencia, entre otros (Gállego Pérez-Larraya, Toledo, Urrestarazu, & Iriarte, 2007). Según la clasificación internacional de trastornos del sueño, el insomnio puede clasificarse como se detalla a continuación.

- Insomnio agudo
- Insomnio psicofisiológico
- Insomnio paradójico
- Insomnio idiopático
- Insomnio debido a trastornos mentales
- Insomnio debido a una inadecuada higiene del sueño
- Insomnio debido a fármacos o tóxicos
- Insomnio debido a problemas médicos

2.1.2. Trastornos respiratorios

En este grupo se incluyen los trastornos que tienen como característica una alteración respiratoria durante el sueño. Esta alteración pueden aparecer periodos en donde el movimiento respiratorio disminuye o desaparece de manera cíclica e intermitente. A continuación, se enlistan los principales trastornos relacionados con la respiración.

- Apnea central primaria
- Patrón respiratorio de Cheyne Stokes
- Síndrome de apnea obstructiva del sueño (SAOS)

- Hipoventilación alveolar central
- Hipoventilación alveolar central congénita

2.1.3. Hipersomnias

Se denomina hipersomnia a las enfermedades que posee problemas de somnolencia diurna, y no precisamente por falta de sueño en las noches o por alteraciones en el ritmo circadiano. La somnolencia diurna se caracteriza por la incapacidad de mantenerse alerta en diferentes situaciones. En la clasificación Internacional de los Trastornos del Sueño se incluyen en este grupo las siguientes:

- Narcolepsia con cataplejia o síndrome de Gelineau
- Hipersomnia recurrente
- Hipersomnia idiopática con sueño prolongado
- Hipersomnia idiopática sin sueño prolongado
- Sueño insuficiente inducido por el comportamiento
- Otros tipos de hipersomnia

2.1.4. Alteraciones del ritmo circadiano

En este grupo se toma en cuenta el tiempo que se debe dedicar para obtener un sueño óptimo y reparador, dicho tiempo puede variar dependiendo de la edad del paciente. A continuación se muestran los síndromes más importantes.

- Síndrome de la fase del sueño retrasada
- Síndrome de la fase del sueño adelantada
- Ritmo sueño-vigilia irregular

- Ritmo sueño-vigilia libre
- Jet Lag o alteración del sueño por viajes con cambio de huso horario
- Alteración del trabajador nocturno
- Alteración del ritmo circadiano debida a un proceso médico

2.1.5. Parasomnias

Las parasomnias se caracterizan por la conducta y el comportamiento anormal del individuo durante el sueño. Basado en algunos estudios, también se cree que la presencia de la parasomnia puede ser por influencia genética. En la clasificación internacional, dichas parasomnias se dividen en tres grupos que a continuación se detallan.

El primer grupo se lo denomina como parasomnias del despertar, entre las cuales se encuentran las siguientes:

- Despertar confusional.
- Sonambulismo.
- Terrores nocturnos.

En el segundo grupo constan las parasomnias asociadas al sueño REM, entre las principales:

- Trastorno de conducta del sueño REM
- Parálisis del sueño aislada
- Pesadillas

El tercer grupo está conformado por otras parasomnias, mismas que se han agrupado por diferentes características diferentes a las mostradas en los grupos anteriores.

2.1.6. Movimientos anormales relacionados con el sueño

En este caso se agrupan los trastornos relacionados con los movimientos anormales durante el sueño. A continuación se muestra la clasificación.

- Síndrome de piernas inquietas
- Movimientos periódicos de las piernas.
- Calambres nocturnos
- Bruxismo
- Movimientos rítmicos durante el sueño

2.1.7. Síntomas aislados

Este grupo refiere a los individuos en dos casos, el primer caso trata sobre pacientes que requieren mayor tiempo de descanso de lo normal, ya que de lo contrario presentan caso de somnolencia. El segundo caso por el contrario presenta personas que comúnmente duermen menos de 5 horas sin presentar problemas de somnolencia.

- Ronquido
- Somniloquios
- Mioclonías del sueño
- Mioclonías benignas de la infancia
- Otros trastornos

2.1.8. Apnea

El síndrome de apnea del sueño (*SAS por sus siglas en inglés*) es un trastorno importante del sueño que causa episodios recurrentes de bloqueo completo denominado apnea o parcial como hipopnea de las vías respiratorias superiores durante el sueño. Uno de los problemas más comunes con este síndrome es el desconocimiento por parte de los pacientes que lo sufren, ya que en la actualidad se ha encontrado una gran cantidad de pacientes con SAS sin diagnosticar. Las visitas clínicas de rutina y los análisis de sangre por lo general no detectan SAS, lo que hace más difícil su estudio. Según algunos expertos, la apnea genera consecuencias adversas, que incluyen hipersomnolencia diurna, disfunción neurocognitiva, enfermedad cardiovascular, disfunción metabólica e insuficiencia respiratoria (Selvaraj & Narasimhan, 2013).

Todos los síntomas y complicaciones ya mencionadas de la apnea desatan significativamente problemas en cuanto a la calidad de vida relacionada con la salud de cada paciente en particular, ya que depende también del grado de severidad del presente trastorno.

Debido a que la apnea es la consecuencia de sucesos anormales que por lo general aparecen durante la etapa de descanso o sueño, es muy importante analizar el grupo de alteraciones que producen dicho trastorno, ya que la gravedad puede variar desde asuntos muy indefensos hasta otros considerados muy graves.

2.2. Bioseñales

El cuerpo humano genera bioseñales las cuales activan el funcionamiento de las diferentes partes del organismo, mismas que son monitorizadas y evaluadas para determinar si existe algún tipo de problema en el paciente.

Las señales son considerados medios de transmisión de información, la adquisición de estas señales permiten descifrar información generada desde la fuente de transmisión. En el caso particular de las bioseñales, ayudan a controlar el funcionamiento de los sistemas fisiológicos del organismo y de esta manera permite al médico dar un diagnóstico. Los factores más importantes que caracterizan a las bioseñales son los rangos de amplitud y frecuencias. Dichos rangos pueden variar dependiendo del método de adquisición de la información (Guerrero Martínez, 2010).

Las bioseñales no son fáciles de entender por lo que se necesita de tecnologías y métodos que completen su análisis y comprensión. La naturaleza de señales fisiológicas son de carácter no lineal y eso hace más difícil su tratamiento, para ello se utilizan modelos de comprensión de señales, donde se pretenden analizarlas y buscar grupos característicos que ayuden a determinar patrones (Gómez Díaz & Lara Rendón, 2008).

2.2.1. Tipos de bioseñales

A continuación en la Tabla 1 se muestran algunas de las bioseñales consideradas las más importantes y sus características correspondientes.

Tabla 1. *Bioseñales relevantes del cuerpo humano*

Bioseñal	Definición	Rango de Amplitud	Rango de Frecuencia
Electrocardiograma (EEG)	Actividad eléctrica cardíaca	0.5 – 4 mV	0.01 – 250 Hz
Electroencefalograma(EEG)	Actividad eléctrica cerebral	5 – 300 μ V	DC – 150 Hz
Electrogastrograma(EGG)	Actividad eléctrica	10 μ V – 1 mV	DC – 1 Hz

gástrica			
Electromiograma (EMG)	Actividad eléctrica muscular	0.1 – 5 mV	DC – 10 kHz
Electroneurograma(ENG)	Actividad eléctrica nerviosa	0.01 – 3 mV	DC – 1 kHz
Electrooculograma (EOG)	Potencial retina – cornea	50 – 3500 μ V	DC – 50 Hz
Electrorretinograma (ERG)	Actividad eléctrica de la retina	0 – 900 μ V	DC – 50 Hz
Fonocardiograma (FCG)	Sonidos cardiacos	80 dB	5– 2 kHz
Flujo sanguíneo	Flujo sanguíneo	1 – 300 ml/s	DC – 20 Hz
Gasto cardíaco	Cantidad de sangre bombeada	4 – 25 litros/minuto	DC – 20 Hz
Presión arterial	Presión sanguínea arterial	10 – 400 mm Hg	DC – 50 Hz
Presión venosa	Presión sanguínea venosa	0 – 50 mm Hg	DC – 50 Hz

Fuente: Adaptado de (Guerrero Martínez, 2010)

Como se pudo observar en la Tabla 1, existen un gran número de bioseñales importantes en el cuerpo. Sin embargo, para la presente investigación, a continuación se describen las más importantes en el estudio de la apnea.

2.2.1.1. Electrocardiograma (ECG). Relacionada con la captación de biopotenciales o funcionamiento eléctrico generados por el corazón. El ECG específicamente realiza monitorizaciones de la actividad como la variación en el número de pulsos con electrodos superficiales.

2.2.1.2. Electroencefalograma (EEG). Registra una actividad eléctrica de las neuronas del encéfalo, obtenida como resultado de los campos eléctricos. Los registros del EEG, muestran señales irregulares que no presentan en general patrones definidos. Por ello, se pueden analizar las ondas en 4 bandas principales: ondas delta, ondas theta, ondas alfa y ondas beta.

2.2.1.3. Electrogastrograma (EGG). Captan la señal eléctrica en el estómago, la señal eléctrica atraviesa órganos internos y varias capas de tejidos para poder ser registrada en la pared abdominal, lo que implica que pertenece a rangos en microvoltios. Además, se encuentra combinada con otras señales provenientes de otros órganos, lo que hace aún más difícil su registro (Villamar, Rabadán, Martínez Pérez, & Garay, 2008).

2.2.1.4. Electromiograma (EMG). Evalúa la actividad eléctrica muscular. La activación muscular se produce como respuesta a un potencial transmitido a través de las fibras nerviosas.

2.2.1.5. Electroneurograma (ENG). Tiene como fin detectar problemas o patologías en las fibras nerviosas, mediante registros de actividades eléctricas en los nervios periféricos, así calculando la velocidad de conducción nerviosa.

2.2.1.6. Electrooculograma (EOG). “Consiste en medir del potencial eléctrico creado por el dipolo ocular con los cambios de fijación de la mirada” (Merino Monge, 2015, p. 256).

2.2.1.7. Electrorretinograma (ERG). “Es el registro resultante de la suma de todas las actividades eléctricas de las células de la retina”(Pérez García, Hernández Echevarría, Rodríguez Martín, & Gómez González, 2016).

2.2.1.8. Fonocardiograma (FCG). Se obtiene sonidos cardíacos mediante diferentes tipos de frecuencia. En casos especiales como el estrechamiento de las válvulas arteriales, se producen sonidos de mayor frecuencia. Debido a esto, es importante un análisis espectral ya que pueden proporcionar un diagnóstico útil para detectar patologías (Alfaro, 2012).

2.2.1.9. Flujo sanguíneo. Determina el flujo sanguíneo ya que en algunos casos las obstrucciones pueden hacer variar la velocidad de la sangre, a pesar de tener la misma presión (Saínz Menéndez, 2006).

2.2.1.10. Presión arterial. Es una magnitud de la fuerza que se aplica sobre las paredes de las arterias y/o venas a medida que el corazón bombea sangre a través del cuerpo. Está determinada por la fuerza y el volumen de sangre bombeada, así como por el tamaño y la flexibilidad de las arterias.

2.3. Adquisición de datos

Los sistemas de adquisición han evolucionado en gran medida, siendo así los electromecánicos los primeros, mismos que poseían muchas limitantes. En la actualidad, existen

registradores completamente electrónicos, los cuales tienen la capacidad de medir cientos de variables al mismo tiempo que utilizan ordenadores para su almacenamiento.

Al principio, el uso de los computadores eran exclusivos, solo se utilizaban en grandes proyectos para recopilación de cientos de variables debido a su gran costo. Con la aparición de las minicomputadoras, se convirtieron en un medio común para recoger datos.

Sin embargo, el funcionamiento de las tarjetas insertadas en los computadores no fue lo más óptimo debido a ruidos producidos por sus componentes mecánicos y electromagnéticos, ya que estos interferían con la medida de las variables. Para ello, el aislamiento ha sido una solución para disminuir dicho problema. Sin embargo, en otros casos se han implementado sistemas de acondicionamiento y procesamiento de la señal. En la Figura 1, se muestra el procesamiento de la señal.

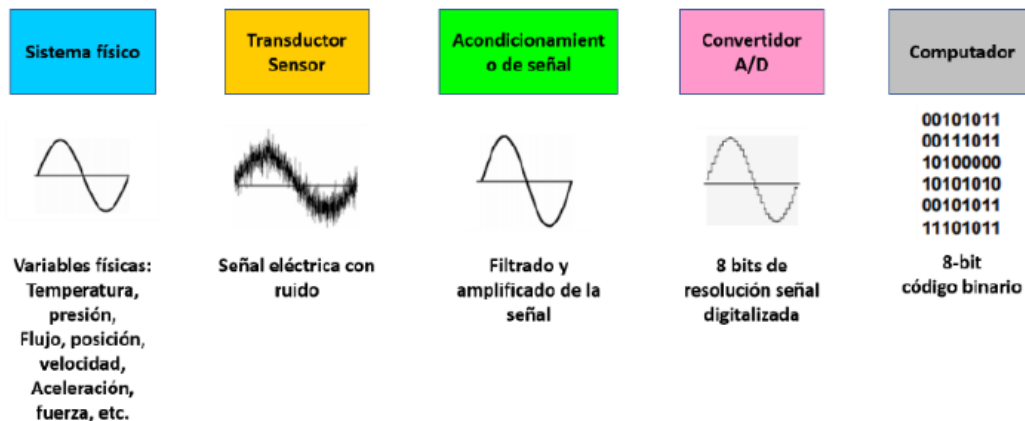


Figura 1. Proceso de adquisición de la señal

Fuente: Adaptado de (Gómez Díaz & Lara Rendón, 2008)

2.3.1. Adquisición Mediante ADC

Los conversores análogos digitales (ADC, por sus siglas en ingles), se encargan de convertir señales analógicas a números binarios y posteriormente en un número digital para su representación en monitores o por medio de gráficos. Sin embargo, los valores obtenidos tan solo son aproximaciones del voltaje analógico. Ya que la resolución de un ADC depende directamente del número de bits y la entrada de voltaje máxima. A continuación se muestran los ADCs más importantes.

2.3.1.1. ADCS de aproximación sucesiva. Los ADC de aproximación sucesiva son lentos debido a que necesitan comparar en cada bit y esperar que se restablezca su salida. Estos ADC están conformados de un DAC, registros de lógica y un comparador. Al determinar la presencia de tensión entrante la lógica de control del sistema pone a todos los bits en cero. Posteriormente el bit más significativo toma el valor de uno, dando como resultado una señal de salida a la mitad de su valor inicial, a continuación se compara los valores de entrada y salida, en donde si el valor de salida es menor al de entrada el bit más significativo permanece en 1 caso contrario tomara el valor de 0. Este proceso se repetirá para los demás bits hasta el bit menos significativo y comparando cada vez que exista una señal de salida (D.A,Handbook, 2012).

A pesar de su lento procesamiento estos pueden llegar a tasas de conversión de hasta 1 Mega Hertzio (Mhz). Además, de ser de bajo costo, razón por la cual son utilizados en la mayoría de sistemas de adquisición.

2.3.1.2. ADC de voltaje a frecuencia. Este ADC convierte el voltaje a un tren de pulsos con frecuencia proporcional a la amplitud del voltaje de entrada. Para determinar la frecuencia se realiza un conteo de impulsos en un determinado periodo de esta manera evitando la aparición de

ruido, ya que la señal se integra durante el proceso de conteo. Este tipo de conversión se utiliza para señales lentas y ruidosas (D.A,Handbook, 2012).

2.3.1.3. Integración de ADCS doble pendiente. Este proceso carga un condensador con una corriente proporcional al voltaje de entrada y se descarga bajo una corriente constante, ayudando de esta manera a determinar el voltaje de entrada. Esta técnica es precisa y estable debido a que no depende de valores de componentes que pueden variar con la temperatura o el tiempo, sino de la relación de tiempos de carga y caída (D.A,Handbook, 2012).

2.3.1.4. Sigma-delta ADCS. Los ADC sigma delta son convertidores de muy bajo costo y útiles para la mayoría de aplicaciones de adquisición de datos debido a que posee un DAC de un solo bit, por lo general estos trabajan con señales de bajo ancho de banda pero son mucho más inmunes al ruido que otros. Estos ADC pueden aceptar señales de bajo nivel sin necesidad de circuitos que acondicionen a la señal (D.A,Handbook, 2012).

En todos los sistemas de adquisición de datos es necesario una etapa de muestreo y retención de datos antes de ADC, debido a que el ADC no podría digitalizar un valor variable en el tiempo, ya que para hacerlo el valor debería cambiar de manera muy lenta con respecto a la frecuencia de muestreo. Algunos ADC contienen circuitos que permiten muestrear y retener, de esta manera a su vez estabilizando la señal de entrada.

2.3.2. Técnicas de Filtrado

Existen tres tipos principales de filtros los cuales se utilizan en diferentes tipos de aplicaciones según la necesidad, ya que cada uno de estos posee diferentes características únicas para cada aplicación y por ende diferentes perfiles de respuesta.

2.3.2.1. Filtros de paso bajo. Los filtros pasa bajos están diseñados para atenuar frecuencias más altas. Un amplificador no necesita de un gran ancho de bando para una señal baja, ya que el filtro reduce el ancho de banda excesivo y por ende el ruido de la señal. Para señales bajas, la amplificación antes del filtrado permite que un filtro de paso bajo activo funcione de manera óptima en relaciones de señal a ruido.

2.3.2.2. Filtros de paso alto. Los filtros pasa bajos funcionan de manera inversa a los pasa bajos debido a que estos atenúan las señales bajas ya que estas podría ocultar señales de alta frecuencia que lleven información invalida. Existe la probabilidad de combinar los filtros los cuales se encargaran de atenuar una banda estrecha de frecuencias.

2.3.2.3. Filtros pasivos vs. Activos. Es normal que una señal se atenué a medida que esta se propague, al igual que el cambio de ciertas características del filtrado. Sin embargo, los filtros activos evitan estos problemas se susciten (elementos activos, tales como amplificadores y sus derivados).

2.3.2.4. Filtros de capacitancia conmutada. El filtro de capacitancia conmutada es una mejora reciente sobre el filtro activo tradicional. Los condensadores con valores parecidos pueden formar circuitos y usándolos en modo de conmutación remplazan las resistencias en los filtros.

2.3.3. Amplificadores

Tras realizar el respectivo filtrado, se obtiene una señal sin ruido pero demasiado pequeña para analizarla y estudiarla de manera óptima y adecuada, es por ello que se utiliza una etapa de amplificación. Debido a que, muchos sensores brindan una señal demasiado baja; es decir menos de los 50 milivoltios (mV), por lo que son demasiado pequeñas para enviarlas directamente al sistema de adquisición, debido a esto es necesario utilizar un circuito amplificador, mismo que pueden ser configurados para amplificar o almacenar señales (Márquez Flores, 2016). A continuación se detallan los amplificadores más importantes.

2.3.3.1. Seguidor. Se usa como un buffer, para eliminar efectos de carga o para adaptar impedancias (*conectar un dispositivo con gran impedancia a otro con baja impedancia y viceversa*).

2.3.3.2. Amplificador inversor. La señal de salida es igual a la señal de entrada (en forma) multiplicada por una constante y de signo contrario (*fase invertida 180 grados*).

2.3.3.3. Amplificador no inversor. Multiplica la señal de entrada por una constante (Márquez Flores, 2016, p. 53).

2.3.3.4. Comparador con histéresis. Este amplificador particular tiene realimentación positiva. Compara las señales de entrada, dando un valor de saturación de acuerdo a los tres casos posibles, aunque la igualdad puede no ser correctamente representada, dada pequeñas fluctuaciones.

2.4. Sensores

En la actualidad la mayoría de sistemas electrónicos necesitan de la presencia de dispositivos capaces de medir diferentes magnitudes para posteriormente analizarlas y mediante un proceso poder controlarlas de la mejor manera posible.

Estos dispositivos físicos denominados sensores tienen la capacidad de detectar cambios, fenómenos o propiedades físicas, químicas o biológicas de su entorno y las convierte en señales eléctricas para posteriormente transformar esas señales eléctricas en una acción y actuar en el entorno situado (Chio Cho, Tibaduiza Burgos, Aparicio Zafra, & Caro Ortiz, 2013).

2.4.1. Según el principio de funcionamiento

2.4.1.1. Activos. los sensores pasivos o generadores de señal tienen como característica especial, originar respuestas de censo sin necesidad de estar conectado a una fuente de voltaje o alimentación externa (Pallás Areny, 1993).

2.4.1.2. Pasivos. Se denominan pasivos aquellos que generan una señal de salida siempre y cuando el dispositivo se encuentre conectado a una fuente de alimentación, es decir no son dispositivos autónomos, ya que depende de un agente externo para ejecutar su función.

2.4.2. Según el tipo de señal eléctrica que generan

2.4.2.1. Digitales. Este tipo de sensores generan señales de salida codificadas ya sea en pulsos o sistemas de Binario decimal codificado (BCD por sus siglas en inglés).

2.4.2.2. Analógicos. Son aquellos sensores que se basan en valores de voltajes para la representación de sus señales de salida.

2.4.3. Según el rango de valores que proporcionan

2.4.3.1. Todo o nada (ON-OFF). son sensores que poseen únicamente dos estados, dichos estados están separados por un umbral dado por la o las variables monitoreadas (Alberto & Farfán, 2010).

2.4.3.2. De Medida. En estos sensores se obtiene una salida proporcional a la señal de entrada.

2.4.4. Según el nivel de integración

2.4.4.1. Discretos. Generalmente utilizado en el circuito de acondicionamiento, realizado mediante elementos electrónicos interconectados entre sí.

2.4.4.2. Integrados. Es la unión de un sensor y un circuito acondicionador, en un solo circuito integrado.

2.4.4.3. Inteligentes. Son aquellos que además de obtener valores, realizan cálculos, autocalibración o autodiagnóstico.

2.4.5. Según el tipo de variable física medida

2.4.5.1. Mecánicos. Pueden medir cantidades como fuerza, velocidad, posición, masa, etc.

2.4.5.2. Eléctricos. Pueden medir voltajes, corrientes, conductividad, entre otros.

2.4.5.3. Magnéticos. Sirven para medir campo, flujo y permeabilidad magnética.

2.4.5.4. Térmicos. Sirven para medir temperatura, flujo, conductividad y calor específico.

2.4.5.5. Acústicos. Tienen como característica convertir sonidos en señales eléctricas, tal es el caso del micrófono.

2.4.5.6. Ultrasónicos. Trabajan como interruptores electrónicos, y emiten pulsos de sonido en el rango del ultrasonido.

2.4.5.7. Químicos. Estos sensores están conformados por un receptor el cual reconoce el tipo de químico y un transductor el cual convierte la señal química en señal eléctrica.

2.4.5.8. Ópticos. Estos sensores se basan en deducir el la interacción entre la luz y la materia para determinar sus propiedades. Tal es el caso de la fibra óptica.

2.4.5.9. Radiación. Se encarga de determinar ciertas propiedades de las partículas radiactivas.

2.4.5.10. Laser. Tiene aplicaciones dedicadas a la medicina, investigaciones científicas y especialmente en el sector militar.

En la Figura 2, se muestra el diagrama de distribución de los sensores antes detallados según su tipo.

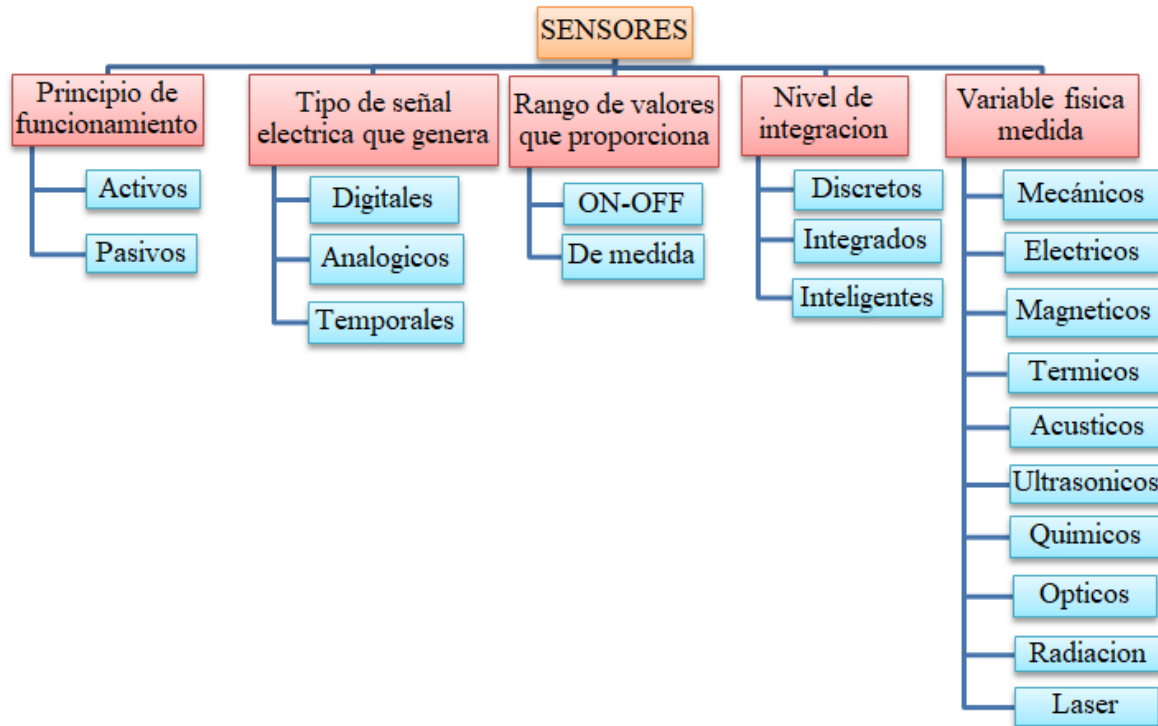


Figura 2. Clasificación de los sensores

Fuente: Adaptado de (Alberto & Farfán, 2010)

2.5. Sistemas inteligentes

Los sistemas de procesamiento ya sean mecánicos, eléctricos y/o electrónicos se constituyen de dos componentes fundamentales los cuales son hardware y software. Estos sistemas se basan en cambios constantes es decir no se mantienen en estado estacionario, mismos que son llamados variables y que a su vez representan ciertas características del sistema. El correcto funcionamiento del sistema depende de parámetros del tiempo y de las variables determinadas. Las variables en su mayoría suelen ser fenómenos inentendibles para el ser humano, por lo que es necesaria la utilización de conversores o transductores que transformen los fenómenos a señales eléctricas, para de esta manera medirlas y posteriormente representarlas a tal punto que el usuario pueda percibir y entender con facilidad (D.A,Handbook, 2012).

Los sistemas inteligentes poseen la capacidad de reflejar de manera objetiva la realidad, desarrollar un pensamiento sistemático y la capacidad de enfrentar problemas desconocidos y complejos, a través de ello obteniéndose resultados óptimos y un desarrollo de sus capacidades. Entre las características más importantes de los sistemas inteligentes se encuentra el intercambio de energía con el entorno, es decir desarrollan la capacidad de repetir procesos de regulación (Ramirez Silva, 2005).

2.6. Aprendizaje de Maquina

Cada ser humano es una fuente de datos con diferentes características, mismas que pueden ser sobre intereses, valores y preferencias de consumo que se registran a través de redes sociales, transacciones bancarias y dispositivos tales como Wi-Fi, Bluetooth, GPS, entre otros. Este fenómeno se conoce como *Big Data*. Para dar tratamiento a esta enorme cantidad de datos en información útil y conocimiento, se aplica aprendizaje de máquina, lo cual técnicamente se basa en enseñar a una computadora a detectar patrones, tomar decisiones y mediante esto resolver problemas, por medio de algoritmos que sean capaces de realizar una tarea, aprender y por ende volverse más inteligente.

El aprendizaje de máquina se ha convertido en una herramienta muy importante en el mundo de los negocios debido a su gran capacidad de aprovechar los datos y convertirlos en información valiosa para cada empresa en particular, de esta manera prediciendo continuamente los cambios en el negocio. Además, utiliza una combinación de algoritmos que aprenden automáticamente de datos con el objetivo de mejorar describir y predecir resultados. A medida que los algoritmos adquieren más datos de entrenamiento estos se vuelven cada vez más precisos.

2.6.1. Análisis de Datos

El análisis de datos realiza un papel muy importante en un sistema inteligente, debido a que posee un proceso el cual consta de una inspección, limpieza, transformación y modelación de los datos con el fin obtener la información útil y necesaria para el sistema, además de sugerencias y conclusiones que aportaran en la toma de decisiones (Ángel M. Rayo, 2016).

Mediante la minería de datos se pretende explotar el conocimiento de los datos, por medio del análisis desde diferentes perspectivas con el fin de obtener y transformar información y conocimiento. Este método no solo tiene como objetivo extraer el conocimiento sino que también busca relaciones o correlaciones que tienen con otros datos aparentemente no conectados. Para obtener resultados muy cercanos los esperados hay que seguir un estricto proceso tal como se muestra en la Figura 3.

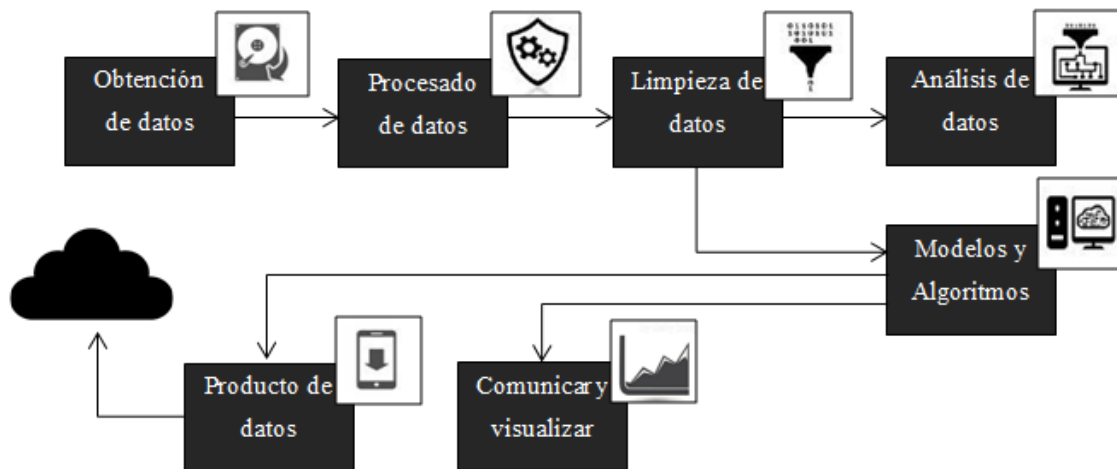


Figura 3. Proceso para el análisis de datos

Fuente: Adaptado de (Ángel M. Rayo, 2016)

2.6.2. Técnicas de Análisis de Datos

Como antes ya se mencionó el proceso debe ser estricto para obtener buenos resultados y se lo debe aplicar en todos los casos de análisis de datos, a continuación se muestran la técnica de análisis de datos más común que utiliza *big data*.

El objetivo principal es encontrar información relevante mediante la utilización de algoritmos óptimos que a su vez permitan la presentación de los resultados de manera visual, tomando en cuenta que la calidad no debe depender de los volúmenes de datos o por el ruido de los mismos sino por la robustez de los algoritmos implementados en el sistema (Vallejos, 2006). En la Figura 4, se muestra el proceso de análisis de Descubrimiento de Conocimiento en Base de Datos (KDD por sus siglas en inglés).

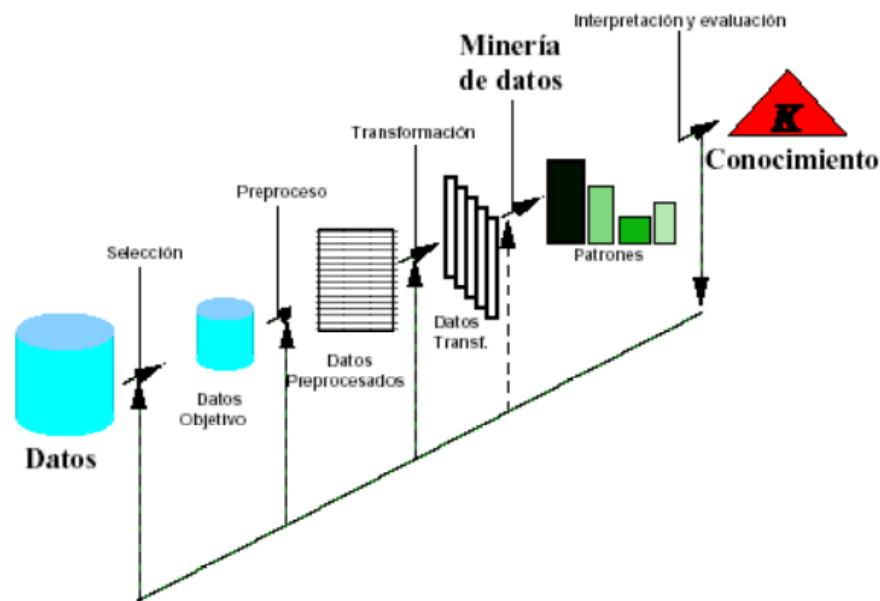


Figura 4. Diagrama del proceso KDD

Fuente: Adaptado de (Vallejos, 2006)

A continuación se detallan las etapas del proceso KDD según (Martin Aldana, Calderon Rivas, & Vargas Hidalgo, 2018).

2.6.2.1. Etapa de selección. En esta etapa se realiza la selección de los datos con los que se va realizar el proyecto, es importante seleccionar minuciosamente debido a que la falta o exceso de datos puede ocasionar resultados diferentes a los esperados.

2.6.2.2. Etapa de preprocesado. Aquí se analiza la calidad de los datos mediante técnicas que puedan reducir los datos que se encuentran fuera del rango esperado más conocido como ruido de la señal. Luego de identificar el ruido se realiza un proceso de limpieza de los datos.

2.6.2.3. Etapa de transformación. Los datos resultantes de la limpieza y de la selección son evaluados a través de sus características para posteriormente representarlos según la necesidad. Tras la evaluación se ejecutan técnicas de reducción las cuales toman una muestra de los datos con características similares y los demás son obviados de esta manera simplificando los datos iguales.

2.6.2.4. Etapa de minería de datos. En esta etapa se encuentra el proceso mediante el cual se analizan posibles patrones existentes en los datos hasta aquí obtenidos, a través de técnicas como clasificación, patrones secuenciales, entre otras. Estos modelos pueden ser predictivos o descriptivos ya que basados en características de otros datos podemos estimar valores futuros.

2.6.2.5. Etapa de interpretación. Aquí se interpretan los datos y patrones antes analizados. Además, es posible quitar los patrones cíclicos e irrelevantes existentes, así traduciendo solo los más importantes.

2.6.3. Algoritmos de Clasificación

La clasificación en minería de datos se considera una técnica supervisada, en la cual generalmente se pretende determinar si los atributos y características pertenecen o no a un determinado concepto o grupo.

La clasificación es utilizada para obtener una función que clasifique un dato de varios predefinidos basándose en características especiales de cada dato, dichas características o variables dependen del tipo de clasificación (Haro, Zúñiga, Vera Rojas, & Villa, 2018). Para este estudio se considera los clasificadores más importantes que se detallan a continuación.

2.6.3.1. Algoritmo Bayesiano. El algoritmo bayesiano ofrece como solución la probabilidad de que la muestra pertenezca a todas las clases existentes. Se considera un método importante debido a que no solo realiza un análisis cualitativo sino que también se enfoca en un análisis cuantitativo de los atributos. Esta es quizá una de la diferencia más fundamental que ofrece el algoritmo bayesiano con respecto a otros métodos como el árbol de decisiones y las redes neuronales (Luque, 2003).

2.6.3.2. CN2 Regla de Inducción. El algoritmo CN2 es un método diseñado para la inducción de reglas sencillas, este modelo se enfoca solo en clasificar. La búsqueda reglas puede ser por medida de evaluación, es decir, selección una regla heurística para evaluar las hipótesis encontradas; o a su vez por el ancho del haz, el cual evalúa la mejor regla encontrada y monitorea el número de alternativas (Haro et al., 2018).

2.6.3.3. Bosque aleatorio. El algoritmo selvas aleatorias es un método de clasificación el cual está conformado por la combinación de varios árboles predictivos, por ende depende de las decisiones tomadas por los árboles que conforman dicho bosque, es decir, la decisión o predicción es seleccionada por mayoría de votos.

2.6.3.4. Redes neuronales. Las redes neuronales han sido ampliamente utilizadas para métodos de clasificación, es decir, es decir se encarga de determinar a qué grupo o clase pertenece un elemento o muestra desconocida, tomando en cuenta sus características.

La capacidad de aprendizaje de las redes neuronales permiten llevar a cabo ciertas actividades ya que aprenden a diferenciar patrones mediante entrenamiento puesto que son dinámicos de esta manera cambian constantemente con el objetivo de adaptarse a nuevas condiciones (Salazar, 2017).

2.6.3.5. K-NN o vecinos más cercanos. El algoritmo k-NN se enfoca en las distancias existentes entre determinado número de vecinos (K vecinos), para clasificar y determinar su pertenencia a una clase que se le encuentre más vecinos, tomando en consideración el criterio de la mínima distancia. Este método es aplicable solo en datos numéricos más no para clasificar textos. Entre los algoritmos más utilizados están ID3, C4.5 y CARD.

2.6.3.6. Tree o árbol de decisión. El árbol de decisiones es un a método de clasificación que permite determinar la decisión en base a las condiciones que se cumplen desde el inicio (raíz), hasta el final (hojas). El árbol de decisión construye dos o más subconjuntos partiendo de la base de datos inicial, posteriormente se vuelve a particionar los subconjuntos (ramas), hasta finalmente obtener las hojas las cuales proporcionan las predicciones.

2.7. Análisis de sensado de Apnea

Para determinar los sensores a utilizar en el presente proyecto se citan investigaciones anteriormente realizadas, las cuales ayudan a un proceso de selección más preciso para un correcto análisis del trastorno.

La apnea obstructiva del sueño (AOS) es un trastorno común del sueño causado por el colapso de la vía aérea superior durante el sueño. A pesar de que existen muchos métodos de monitorización y deducción de dicho trastorno, los investigadores utilizan el índice de desaturación de oxígeno (ODI) y el tiempo total por debajo de los niveles de saturación, para detectar y tratar la apnea con la ayuda de una almohada inteligente. Para ello se utilizó la oximetría de pulso ya que es uno de los sensores más populares que mide la saturación de oxígeno en la sangre (SpO₂). Estos sensores son portátiles y simples de implementar, pudiendo ser colocados por los pacientes sin la necesidad de asistencia profesional (Mostafa, Mendonça, Morgado-Dias, & Ravelo-García, 2017).

Para el proceso de investigación se considera los valores establecidos en estudios ya realizados. A continuación se muestra en la Tabla 2.

Tabla 2. Porcentajes de niveles de oxígeno en la sangre.

% saturación	Actuación
> 90%	No actuación inmediata.
85-90%	Tratamiento inmediato y monitorización de la respuesta al mismo, según ésta, valorar derivación al hospital. Los pacientes con enfermedad respiratoria crónica toleran bien saturaciones en torno a estos valores.
< 85%	Enfermo grave. Hipoxia severa. Oxigenoterapia + tratamiento y traslado al hospital.
< 80%	Valorar intubación y ventilación mecánica.
En niños con < 92%: Remitir al hospital aunque presenten mejoría con maniobras iniciales, por ser más incierta su respuesta al tratamiento.	

Fuente: Obtenido de (Noguerol & Seco, 2017)

Por otro lado, se deduce que los movimientos corporales y la postura durante el sueño también son importantes para evaluar los patrones irregulares del sueño. Para este estudio, se explora la posibilidad de utilizar acelerómetros para detectar la posición y parámetros cardiorrespiratorios durante el sueño. De esta manera, detecta posturas estáticas en movimientos dinámicos. Posteriormente, se utiliza un clasificador de análisis discriminante lineal (IDA) para dividir aún más las posturas estáticas en cuatro posiciones comunes para dormir (Zhang & Yang, 2015).

En otro estudio, presenta un nuevo algoritmo que extrae características basadas en el filtrado y la dispersión estadística de la señal de respiración del flujo de aire nasal que detecta los eventos de apnea por segundo. Su implementación puede permitir una predicción precisa de un próximo episodio de apnea que, en consecuencia, evita el colapso de la vía aérea (Selvaraj & Narasimhan, 2013).

Para efectos de investigación se toma como referencia los valores normales de frecuencia respiratoria, como se detalla en la Tabla 3.

Tabla 3. Frecuencia respiratoria

EDAD	RESPIRACIONES POR MINUTO
Recién nacido	30-80
Lactante Menor	20-40
Lactante Mayor	20-30
Niños de 2 a 4 años	20-30
Niños de 6 a 8 años	20-25
Adulto	15-25

Fuente: Obtenido de (Aguayo P & Lagos T., 2012)

También se puede verificar un estudio en el cual se evalúan nuevos métodos de detección de ronquidos basados en redes neuronales híbridas. El método propuesto utiliza características de codificación de predicción lineal. El rendimiento del método propuesto se evaluó clasificando diferentes eventos entre los cuales están: ronquidos, apnea y silencio (Watanabe et al., 2005).

Como se pudo constatar, una señal elegida por algunos investigadores es el Electrocardiograma (ECG por su abreviación), que normalmente se utiliza para calcular la variabilidad de la frecuencia cardíaca y de esta manera detectar la apnea. Otros investigadores utilizan SPO2 y otros usan sensores de presión. En otros casos el sonido también se utiliza para diagnosticar la apnea debido a que por medio de este se determina el ronquido y se deduce el flujo de aire.

Según estas importantes investigaciones, se determinó para el análisis del trastorno del sueño denominado apnea, la utilización de los siguientes sensores, los cuales cumplen sus funciones de monitorización en diferentes partes de la anatomía corporal.

2.7.1. Sensor SPO2

Este sensor es el encargado de monitorizar la pulsi-oximetría de manera no invasiva, es decir mide el oxígeno transportado por la hemoglobina de los vasos sanguíneos. Además, el sensor informa sobre la frecuencia y el pulso del ritmo periférico (Meza, Llamosa, & Ceballos, 2007).

La saturación de oxígeno se define como la medición de la cantidad de oxígeno disuelto en la sangre, basada en la detección de hemoglobina (Hb) y desoxihemoglobina (HbO₂). Se

utilizan dos longitudes de onda de luz, una de 660 nm (espectros de luz roja) y 940 nm (espectros de luz infrarroja) para medir la diferencia real en los espectros de absorción de HbO₂ y Hb (Libelium Comunicaciones, 2017).

2.7.2. Sensor de Detección de Posición del Cuerpo

El sensor de posición del cuerpo utiliza un acelerómetro de triple eje para obtener la posición del paciente. Permite la medición de las aceleraciones en 3 ejes perpendiculares y, por lo tanto, detecta la inclinación, el movimiento y el impacto. Debido a su bajo consumo potencial, es uno de los sensores más usados para aplicaciones de mercado (Libelium Comunicaciones, 2017).

A través de este sensor se analiza las posiciones y movimientos realizados debido a su relación con enfermedades particulares tales como la apnea del sueño y síndrome de piernas inquietas.

2.7.3. Sensor de Detección de Ronquido

El sensor convierte los ronquidos y otros sonidos en el rango de audio captados a través de la piel, a un pequeño voltaje analógico que proporciona una indicación clara y confiable de la presencia de estos sonidos. Este método, al permanecer el mayor tiempo posible en contacto con la piel se considera el más óptimo, ya que capta menos ruido proveniente del entorno.

El ronquido es un síntoma importante de la apnea obstructiva del sueño (AOS). Debido a esta importancia, es necesario realizar estudios que analizan las propiedades acústicas de los ronquidos que deben adquirir datos a altas tasas de muestreo, por lo que se debe procesar una gran cantidad de datos (Libelium Comunicaciones, 2017).

Posterior a los estudios y análisis los datos se deberán clasificar como ronquidos, ruido y silencio según sus HMM.

2.8. Almacenamiento de datos

El almacenamiento de datos ha ido evolucionando con el pasar de los años, desde sistemas de discos mecánicos hasta actualmente utilizar sistemas de almacenamiento lógicos controlados a través de software. De esta manera archivando, organizando y compartiendo información a través de una red estrictamente controlada. A continuación, se describen los tipos más importantes de almacenamiento de datos (Vázquez-Moctezuma, 2015).

2.8.1. Almacenamiento definido por software

El almacenamiento definido por software (SDS por sus siglas en inglés) usa sistemas de software de gestión por extracción para separar los datos del hardware antes de cambiar su formato y organizarlos para su uso en la red. En particular, el SDS es útil para las cargas de trabajo de contenedores y microservicios que utilizan datos sin estructurar, ya que puede expandirse a un nivel que las soluciones de almacenamiento conectadas simplemente no pueden alcanzar.(Galmés, 2016)

2.8.2. Almacenamiento en la nube

Es la organización de los datos almacenados en cierto lugar al que puede acceder cualquier persona que tenga los permisos adecuados, a través de Internet. No es necesario que esté conectado a una red interna ni que acceda a los datos desde un sistema de hardware conectado directamente a la computadora (Vázquez-Moctezuma, 2015).

2.8.3. Almacenamiento adjunto a la red

El almacenamiento adjunto a la red (NAS por sus siglas en inglés) facilita el acceso a los datos por parte de las redes internas instalando un sistema operativo liviano en un servidor que lo convierte en una caja, unidad o cabezal de NAS. La caja de NAS se convierte en una parte importante de las intranets porque procesa todas las solicitudes de almacenamiento.

2.8.4. Almacenamiento de objetos

El almacenamiento de objetos divide los datos en unidades independientes y las combina con los metadatos para brindar contexto sobre su contenido. Los datos almacenados en estos objetos no están comprimidos ni cifrados, lo cual permite que las cargas de trabajo que cambian rápidamente, como los contenedores, accedan a ellos a gran escala (Rengifo García, 2013).

2.8.5. Almacenamiento en bloques

En el almacenamiento en bloques, se dividen los volúmenes de almacenamiento en instancias individuales conocidas como bloques. Cada bloque es independiente, por lo que los usuarios tienen la autonomía total sobre la configuración. Dado que los bloques no tienen los mismos requisitos de identificador único que los archivos, constituyen un sistema de almacenamiento más rápido. Esto los convierte en el formato ideal para las bases de datos de contenido multimedia.

2.9. Modelos de desarrollo

Durante el desarrollo de un proyecto suelen producirse ciertas dificultades, mismas que se pueden solventar con la utilización de una metodología que permite guiar el proyecto. Una metodología se enfoca en ejecutar, controlar y administrar el proyecto con suficientes

posibilidades de éxito (Portilla, 2018). Existen diversas metodologías que proponen dividir un proyecto en sub-módulos o etapas con el fin de normalizar el método de administración del proyecto. A continuación se detallan los modelos de desarrollo más utilizados e importantes.

2.9.1. Modelo lineal

Es considerado el más simple, ya que al no existir retroalimentación, cada una de las etapas se desarrolla independientemente. En la Figura 5, se muestra las etapas de este modelo.

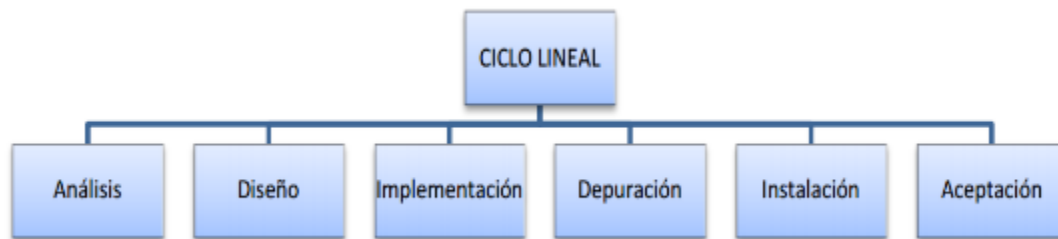


Figura 5. Etapas del modelo lineal

Fuente: Obtenido de (Portilla, 2018)

Este modelo se recomienda implementar en proyectos pequeños. Sin embargo, no es recomendable utilizar en proyectos que necesiten retroalimentación en algún sub-módulo de su estructura.

2.9.2. Modelo en cascada

Este modelo ordena cuidadosamente cada una de sus etapas debido a que el inicio de las mismas depende de la finalización de las etapas anteriores; es decir, es un modelo secuencial por lo que los resultados esperados solo se obtienen al finalizar la última etapa (Ruíz & González, 2014). En la Figura 6, se muestran las etapas del modelo.

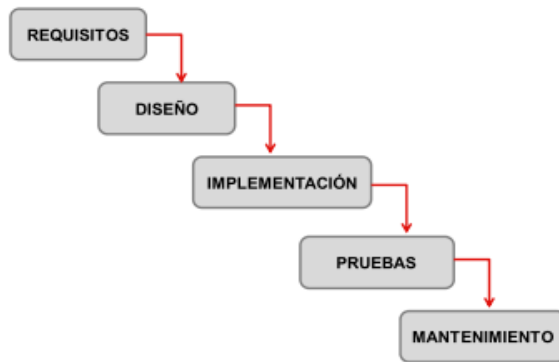


Figura 6. Etapas del modelo en cascada

Fuente: Obtenido de (Ruíz & González, 2014)

2.9.3. Modelo en V

En modelo en V es una modificación del modelo en cascada. Sin embargo, posee ciertas características diferentes tales como el proceso de verificación y de retroalimentación. Este modelo es ampliamente utilizado para proyectos de alta confiabilidad. En la Figura 7, se muestran las etapas del mismo.



Figura 7. Modelo en V

Fuente: Obtenido de (Ruíz & González, 2014)

CAPÍTULO III. DESARROLLO EXPERIMENTAL

En el presente capítulo, se realiza la descripción general del sistema tanto en funcionamiento como en estructura, a través de diagramas de bloques, determinación y descripción de subsistemas, con el fin de cumplir ciertas etapas en desarrollo con ayuda de métodos y/o técnicas que a continuación se utilizan para el sistema.

3.1. Análisis general

El sistema se enfoca en el monitoreo y clasificación de bioseñales relacionadas a la apnea, por este motivo el entorno en el cual se realizarán las lecturas de datos serán lo más habitual para el paciente. De esta manera evitando afectar los valores detectados por cada uno de los sensores y obtener un resultado más óptimo. Para ello, se implementa el prototipo que tiene como principal objetivo detectar las tendencias a tener apnea. Además, brindar un dispositivo totalmente cómodo y seguro al usuario.

3.2. Metodología

Las metodologías antes planteadas tienen diferentes características importantes para la ejecución e implementación del proyecto. Sin embargo, la metodología espiral, cascada y lineal no poseen una estructura fundamental para implementar un proyecto de manera segura y confiable, debido a que las mismas son orientadas a proyectos con diferentes enfoques. Es por esto que, al presente proyecto se aplica el modelo en V como metodología de trabajo debido a que este realiza validaciones en cada fase y su proceso permite tener una consideración adecuada de las distintas perspectivas que son requeridas para el proyecto, de esta manera facilitando la

detección de fallas mediante pruebas para solucionarlas a tiempo, así minimizando lo riesgos de fallas en la etapa final.

El modelo en V posee dos áreas y 4 niveles respectivamente, en el área de la izquierda constan el análisis, los requerimientos, el diseño del sistema y finalmente la implementación y en la área derecha se encuentra cada uno de los niveles de validación pertenecientes a cada etapa. A continuación en la Figura 8 se muestra la estructura antes mencionada.



Figura 8. Estructura modelo en V

Fuente: Obtenido de (Portilla, 2018)

En el primer nivel se definen y detallan los parámetros del proyecto, a través de la recopilación de información de características y/o factores que afecten de manera positiva o negativa al desarrollo e implementación del presente.

En el segundo nivel se presentan los requerimientos del sistema, el cual es considerado una de las fases más importantes en el desarrollo del proyecto, debido a que aquí se define la funcionalidad del sistema propuesto.

El tercer nivel se enfoca en obtener un diseño y visión general del proyecto a través de la fase de pruebas unitarias la cual verifica y valida cada parte del proyecto y su correcta implementación.

Finalmente el nivel de implementación se encarga de materializar el diseño del proyecto. Además, se implementa la debida programación para cada una de las partes del sistema.

3.2.1. Situación actual

Las personas con problemas de salud comúnmente sufren de trastornos del sueño los cuales afectan directamente a las acciones y actitudes de cada individuo en particular, en algunos casos particulares ocasionando la muerte súbita en neonatos. Olivi H. (2013) argumenta que “El desarrollo de los síntomas de este trastorno es lento y progresivo”, por ende dichos síntomas pueden generar graves consecuencias de no ser detectados y tratados a tiempo. Para ello, es necesario realizar un proceso de análisis exhaustivo de los parámetros estrechamente ligados al trastorno del sueño denominado apnea, para detectar y prevenir posibles complicaciones.

Para el desarrollo del presente proyecto se determina la necesidad de apoyar la investigación e investigación en la metodología antes mencionada, en la cual se establecen algunos requerimientos importantes para el correcto diseño e implementación del prototipo.

Con el objetivo de definir los requerimientos de los participantes en el desarrollo del sistema. Se establecen requisitos que permitirán un desarrollo satisfactorio del proyecto y la interacción del mismo con el usuario. En la Tabla 4, se definen los requerimientos de los stakeholders.

Para cada uno de los requerimientos se establecen abreviaturas para ello se muestran las mismas en la Tabla 4.

Tabla 4. Abreviaturas de requerimientos

ABREVIATURAS DE REQUERIMIENTOS	
Abreviatura De Stakeholders	STSR
Abreviatura De Sistema	SYSR
Abreviatura De Arquitectura	SRSR

Fuente: Adaptado de (Dibujes, 2018)

Para determinar las prioridades de los requerimientos de los usuarios, se toman en cuenta algunos criterios importantes tales como el rendimiento, riesgo e implementación del proyecto.

En la Tabla 5, se muestra las prioridades asignadas.

Tabla 5. Prioridad de los Requerimientos

Prioridad	Descripción
Alta	Se considera un requerimiento crítico, ya que si no se implementa afecta a la funcionalidad del sistema.
Media	La exclusión de este requerimiento puede producir problemas en la decisión final. Sin embargo, puede obviarse en caso de fuerza mayor.
Baja	La no inclusión de este requerimiento al sistema no afecta a la decisión final del sistema.

Fuente: Adaptado de (Portilla, 2018)

3.2.2. Encuesta de requerimientos

Esta encuesta va dirigida a personal médico o personas con conocimiento del trastorno a tratarse. Esta tiene como objetivo recolectar información de los requerimientos que el usuario desea y así definirlos para satisfacer las necesidades de las personas que sufren dicho trastorno.

Se realiza la encuesta con el objetivo principal de responder a los requerimientos de los usuarios, tomando como base fundamental la comodidad y satisfacción de los mismos.

La encuesta correspondiente contiene 5 preguntas de selección múltiple, en las cuales los encuestados señalaran la que cada uno crea conveniente para dicha investigación. El formato se encuentra en el Anexo 1.

3.2.2.1. Tamaño de la muestra. El tamaño de la muestra asegura la confiabilidad y valides de la encuesta. Para ello, se calcula el número de muestras necesarias para cumplir con la validación pertinente. A continuación se muestra ecuación utilizada para dicho cálculo.

$$n = \frac{N\sigma^2 Z^2}{(N - 1)e^2 + \sigma^2 Z^2}$$

Ecuación 1. Calculo de tamaño de muestra

Fuente: Obtenido de (Maldonado, 2018)

Dónde:

n= número de muestras

N= Tamaño de la población

Z= Nivel de confianza, constante que puede ser (99%=2.58; 95%=1.96). Dichos valores pueden ser elegidos a criterio del encuestador, y de acuerdo al enfoque que la encuesta tenga.

e= error muestral (no mayor al 9%)

σ = Desviación estándar de la población, valor constante de 0.5

Los valores utilizados en la ecuación se tomaron de la tabla de distribución normal del

Anexo 1

$$n = \frac{60(0.5)^2(1.96)^2}{(60 - 1)0.09^2 + (0.5)^2(1.96)^2} = 40.6$$

El tamaño de la muestra requerido para la validación de las encuestas es de 41 personas con conocimiento del trastorno denominado apnea.

En base a la encuesta realizada se determinó los requerimientos necesarios para satisfacer las necesidades del sistema, cada uno de estos requerimientos fueron valorados con prioridad alta.

3.2.3. Stakeholders

Los requerimientos de stakeholders son todos aquellos individuos cuyo interés es el resultado final de la investigación o proyecto. En la Tabla 6, se muestran los stakeholders implicados en el presente proyecto.

Tabla 6. Lista de stakeholders

Lista de stakeholders	
Directos	
1.	Usuario/paciente
2.	Familiares de usuario
Indirectos	
3.	Ing. Paul Rosero Director del presente trabajo de titulación.
4.	Ing. Carlo Vásquez Codirector de Presente Trabajo de Titulación
5.	Ing. Jaime Michelena Codirector de Presente Trabajo de Titulación
6.	Jhonny Gerardo Ipiates Matango

Fuente: Autoría

A continuación en la Tabla 8 se detallan las prioridades de los requerimientos de stakeholders antes establecidos por la encuesta, entre los cuales constan los requerimientos operacionales y de usuario.

Tabla 7. Requerimientos de stakeholders

REQUERIMIENTOS DE STAKEHOLDERS (StSR)				
#	REQUERIMIENTOS DE USO	PRIORIDAD		
		Alta	Media	Baja
REQUERIMIENTOS OPERACIONALES				
StSR 1	Adecuada ubicación de los sensores para la monitorización	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
StSR 2	Comodidad al usar el dispositivo	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
StSR 3	Protección para evitar manipulación interna del sistema	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
StSR 4	El sistema debe presentar bajo consumo de energía.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
StSR 5	Alimentación eléctrica portable(batería)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
REQUERIMIENTOS DE USUARIOS				
StSR 6	El usuario no debe manipular ningún elemento interno del sistema.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
StSR 7	El usuario debe descansar de manera normal en su entorno natural, sin alteración alguna.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
StSR 8	El usuario debe recibir el resultado de su diagnóstico tras finalizar el monitoreo (correo electrónico, mensaje de texto).	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Fuente: Adaptado de (Portilla, 2018)

3.3. Requerimientos del sistema

Por medio del análisis de los requerimientos del sistema, se establecen consideraciones específicas para el diseño y desarrollo del sistema, entre las cuales se encuentran los requerimientos iniciales del sistema, de arquitectura y de stakeholders. Toda esta información se presenta en tablas con el fin de presentar de manera resumida los aspectos más importantes para posteriormente seleccionar el software y hardware más óptimo para el presente proyecto.

3.3.1. Requerimientos iniciales del sistema

Los requerimientos iniciales del sistema indican los límites del sistema para el comportamiento y propiedades del proyecto mediante el análisis de las necesidades del proyecto.

A través de los requerimientos se determinan parámetros de funcionamiento de interfaz con el usuario, mismos que permitirán un buen desarrollo del proyecto. La interfaz más importante en el proyecto se refiere a la misma que el usuario manipulará para ejecutar una acción. Además se considera como interfaz de software a la compatibilidad de la programación realizada con las placas de desarrollo. Además, especifican parámetros de rendimiento de sistema y las funciones que el mismo debe realizar en cada situación en particular. Los requerimientos que se pretende analizar en el proyecto son los mismos ya establecidos por cada bioseñal necesaria para la detección de la apnea.

A continuación en la Tabla 9 se detalla los requerimientos considerados importantes con las respectivas prioridades establecidas según la Tabla 7.

Tabla 8. Requerimientos iniciales del sistema.

REQUERIMIENTOS INICIALES DEL SISTEMA(SySR)				
#	REQUERIMIENTOS	PRIORIDAD		
		Alta	Media	Baja
REQUERIMIENTOS DE INTERFAZ				
SySR 1	Sistema cuenta con módulo de transmisión de información acoplado al sistema embebido.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SySR 2	Sistema cuenta con sensores que se acoplan a la placa principal.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SySR 3	El sistema electrónico (Sistema embebido, los sensores y la alimentación eléctrica) deberán interactuar entre sí.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SySR 4	El sistema posee un switch de encendido y apagado	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
REQUERIMIENTOS DE USO				
SySR 5	La carga del sistema debe ser de 5v	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SySR 6	Conexión del sistema embebido a su fuente de energía para su encendido.	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
SySR 7	El sistema debe mantener su integridad durante el proceso de monitoreo.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
REQUERIMIENTOS DE PERFORMANCE				

SySR 8	Tratamiento de datos obtenidos a través de los sensores.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SySR 9	Representación de datos procesados por el sistema hacia la pantalla de visualización del computador.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SySR 10	Brinda diagnóstico de trastorno evaluado.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
REQUERIMIENTOS DE MODOS/ESTADOS				
SySR 11	Evaluación de todos los parámetros de los sensores	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SySR 12	El sistema debe recolectar datos cada minuto aproximadamente.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SySR 13	Los datos deben ser enviados a la nube al finalizar el diagnostico.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SySR 14	El sistema debe iniciarse automáticamente al encenderse	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
REQUERIMIENTOS FISICOS				
SySR 15	El dispositivo debe ubicarse en un lugar del cuerpo que no interfiera con el descanso del usuario.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SySR 16	La ubicación de los sensores debe estar en partes estratégicas del cuerpo del paciente.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SySR 17	Tamaño reducido del sistema embebido para evitar incomodidad al usuario.	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Fuente: Adaptado de (Portilla, 2018)

3.3.2. Requerimiento de arquitectura.

En los requerimientos de arquitectura se establece los componentes de software, hardware y sistema eléctrico. Es por esto que la ubicación de los sensores es parte fundamental para una óptima recolección de los datos y de algunas modificaciones en el hardware. Además, es necesario especificar los elementos electrónicos que se van a emplear en el sistema tales como: sensores y sistema embebido, los cuales se encargaran de la recolección y procesamiento de los datos. También se considera la ejecución del algoritmo de clasificación implementado a través del software utilizado en el sistema y finalmente la fuente de alimentación para el sistema embebido y regulador de voltaje. En la Tabla 10 se muestran los requerimientos de los elementos indispensables tanto en software y hardware para el sistema.

Tabla 9. Requerimientos de arquitectura

REQUERIMIENTO DE ARQUITECTURA(SRSH)				
REQUERIMIENTOS LOGICOS				
#	REQUERIMIENTOS	PRIORIDAD		
		Alta	Media	Baja
SRSH 1	Microcontrolador compatible con módulo de comunicación.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
REQUERIMIENTOS DE DISEÑO				
SRSH 2	El sistema debe adquirir datos de entrenamiento para construcción del algoritmo de aprendizaje automático.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SRSH 3	Los sensores deben estar correctamente ubicados en el cuerpo para la correcta recolección de datos	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SRSH 4	Los cables en los sensores deben ubicarse de forma estratégica para no incomodar al usuario	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SRSH 5	Los sensores de monitoreo deben conectarse directamente a la placa principal de procesamiento.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
REQUERIMIENTOS DE HARDWARE				
SRSH 6	Sistema embebido compatible con la librería de software libre de sensores.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SRSH 7	El sistema embebido requiere más de un puerto para comunicación serial.	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

SRSH 8	Sistema embebido compatible con multiplexor analógico para la adquisición de datos con sensores.	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
SRSH 9	Sistema embebido capaz de procesar con rapidez los datos obtenidos con los sensores	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SRSH 10	Considera la sensibilidad de los sensores.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SRSH 11	Se requiere bajo consumo de energía.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SRSH 12	El sistema puede manejar distintos tipos de sensores a la vez	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
SRSH 13	Los sensores debe adaptarse al sistema embebido	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SRSH 14	Los sensores deben tener compatibilidad o librerías con respecto a la placa utilizada	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
REQUERIMIENTOS DE SOFTWARE				
SRSH 15	Sistema Operativo y Lenguaje de programación de código abierto.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SRSH 16	Compatibilidad con Librerías de sensores	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SRSH 17	Se requiere software que permita ejecutar el código desde un sistema embebido	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
SRSH 18	Compatibilidad de software con la biblioteca para aprendizaje automático.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
REQUERIMIENTOS ELÉCTRICOS				
SRSH 19	Fuente de alimentación 5V	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Fuente: Adaptado de (Portilla, 2018)

3.3.3. Selección de hardware y software

En la selección de los elementos tanto de software como de hardware es necesario realizar una comparación de las especificaciones de los mismos, de esta manera se elige a los elementos que cumplan como los requerimientos establecidos.

3.3.3.1. Sistema Embebido. Para este caso satisfacen las necesidades del proyecto 3 opciones de las cuales se elegirá la más óptima para la utilización en el presente. Para la valoración se utilizó un 1 si cumple con el requerimiento o 0 si no cumple con el mismo. A continuación se muestra en la Tabla 11 la valoración para la selección del sistema.

Tabla 10. Elección del sistema embebido.

HARDWARE	REQUERIMIENTOS			VALORACION TOTAL
	SRSH 6	SRSH 7	SRSH 14	
Arduino	1	1	0	2
Raspberry Pi 3	1	1	0	2
Módulo MCU	1	1	1	3

1 Cumple
0 No Cumple

Elección: Según la valoración realizada muestra al NodeMCU como el hardware más idóneo para la implementación del sistema.

Fuente: Adaptado de (Arduino Company, 2015)

En el Anexo 11 se muestran las especificaciones técnicas definidas por los fabricantes para la placa NodeMCU.

3.3.3.2. Sensores SPO2. Para la selección del sensor SPO2 se toma en consideración los requerimientos importantes necesarios para la implementación. A continuación, se detalla en la Tabla 12 los sensores considerados.

Tabla 11. Selección de sensor SPO2

HARDWARE	REQUERMIENTOS			VALORACION TOTAL
	SRSH 10	SRSH 13	SRSH 14	
SpO2 OXI-PRO	1	0	1	2
EMISOR Y RECEPTOR (IR Y R)	1	1	1	3
SPO2 INTRUSIVO	1	1	0	2
1 Cumple				
0 No cumple				

Elección: En este caso los dos sensores fueron establecidos con el mismo valor por lo que para la utilización se selecciona al emisor y receptor rojo e infrarrojo por facilidad de accesibilidad y costo reducido.

Fuente: Obtenido de (Libelium Comunicaciones, 2017)

En el Anexo 12 se muestra las principales características de los sensores previamente seleccionados.

3.3.3.3. Sensor de posición del cuerpo. Para la posterior selección del sensor de posición del cuerpo se toma como aspectos importantes los requerimientos antes establecidos y la respectiva valoración de cada uno. En la Tabla 13 se muestran la valoración para cada sensor.

Tabla 12. Selección de sensor de posición del cuerpo

HARDWARE	REQUERIMIENTOS			VALORACION TOTAL
	SRSR 10	SRSR 13	SRSR 14	
ADXL335 GY61	1	1	1	3
MLX90316	1	1	0	2
Sensor HCW4	1	0	1	2
1 Cumple				
0 No cumple				

Elección: En este caso el sensor que fue seleccionado fue el Sensor acelerómetro ADXL335 GY61 por obtener la mayor valoración. Además, se toma en consideración la facilidad de accesibilidad y costo reducido.

Fuente: Obtenido de (Libelium Comunicaciones, 2017)

Luego de la valoración respectiva, en el Anexo 13 se muestran las características principales del sensor de posición de cuerpo seleccionado.

3.3.3.4. Sensor de medición de ronquido. Finalmente se procede a la selección y valoración del sensor de medición de ronquido. En la Tabla 14 se detalla la valoración establecida para cada caso.

Tabla 13. Selección de sensor de ronquido

HARDWARE	REQUERMIENTOS			VALORACION TOTAL
	SRSH 10	SRSH 13	SRSH 14	
Micrófono electret	1	1	1	3
RESBP-sensor	1	1	0	2
Snore-sensor 1250	1	0	1	2

1 Cumple

0 No cumple

Elección: En este caso el sensor que fue establecido como más óptimo para el proyecto es el Micrófono electret por obtener mayor valoración, facilidad de accesibilidad y costo reducido.

Fuente: Obtenido de (Libelium Comunicaciones, 2017)

Luego de la selección a través de la valoración en cada uno de los respectivos sensores, en la Anexo 14 se muestran las características más importantes del sensor previamente seleccionado.

Como resultado de la selección a través de la valoración y comparación de cada uno de los sensores se determina la efectividad en la utilización de la placa NodeMCU y de los sensores tales como: SPO2, flujo de aire, presión de sangre, posición del cuerpo y ronquido.

3.3.3.5. Selección de Software

Luego de la selección del hardware se realiza la selección del software, basado en los requerimientos previamente establecidos en la Tabla 10. Entre algunas de sus características esta la compatibilidad con las bibliotecas y librerías necesarias que dan paso al funcionamiento de cada uno de los sensores de Mysignals. A continuación, en la Tabla 15 se muestra la selección del software con respecto al hardware seleccionado.

Tabla 14. Elección del software.

SOFTWARE	REQUERIMIENTOS			VALORACION TOTAL
	SRSH 15	SRSH 16	SRSH 17	
C++	1	1	1	3
Python	1	0	1	2

1 Cumple
0 No cumple

Elección: De acuerdo a la valoración y a los requerimientos establecidos, se selecciona C++ como software para el sistema debido a que es un sistema de código libre, compatible con la placa seleccionada y con librerías necesaria para el correcto funcionamiento del sistema.

Fuente: Obtenido de (Arduino Company, 2015)

3.4. Diseño global del sistema

Una vez realizada la selección de los elementos que componen el sistema a implementarse, se realiza el diseño del sistema, hay que tomar en cuenta las limitaciones, criterios y requerimientos del sistema, mismos que permitirán un desarrollo e implementación satisfactoria.

3.4.1. Descripción general del sistema

El sistema embebido inteligente de representación de datos biomédicos, se basa en el análisis de parámetros específicos para la predicción del trastorno del sueño denominado apnea. Este trabajo presenta un novedoso sistema que monitorea en tiempo real el comportamiento de las bioseñales, mediante algoritmos de aprendizaje automático. Para ello, se establecen diferentes casos de medición donde cada uno brinda datos de las condiciones propias de cada condición para luego ser procesados internamente por el sistema. En este sentido, se considera variables de oxígeno de sangre, ritmo cardiaco, presencia de ronquido, respiración por minuto y posición del cuerpo. Adicionalmente, se realiza una etapa de integración y calibración de sensores para eliminar errores de lectura.

Es importante tener presente que los valores que el sistema puede detectar a través de los sensores son niveles muy bajos los cuales varían entre 0,1 y 0,2 milivoltios (mV), dificultándose de esta manera el tratamiento de los datos. Por esta razón, es necesaria la implementación de una etapa de amplificación, la cual permitirá su correcto tratamiento. Además, la amplificación es necesaria para que la señal tenga un nivel de amplitud lo suficientemente grande como para poder diferenciarla del ruido. Éste puede atribuirse a muchas fuentes, como por ejemplo el ruido por movimiento del dedo del usuario y el ruido de cualquier fuente de luz del entorno.

Luego se utiliza el filtro lógico Exponential Moving Average (EMA), el cual básicamente toma un número de muestras N , y obtiene el promedio de dichos datos, de esta manera reduciendo el ruido producido al recopilar los datos a través de los sensores, y evitando variaciones demasiado grandes producidas por falsos positivos o datos totalmente discordes a los demás. Posteriormente se realiza el análisis de los datos acondicionados en las anteriores etapas a través de KDD, con el fin de obtener conclusiones y un conocimiento más preciso antes de tomar decisiones en base a dicha información. Para ello se busca el algoritmo de clasificación más óptimo para este caso. Entre los más usuales a utilizar están los algoritmos bayesiano, árbol de decisiones, regla de inducción y k-NN.

A continuación, se realiza el proceso de pruebas en las que se comparan los datos nuevos obtenidos de los usuarios con los ya existentes en el sistema y finalmente enviar los datos a un servidor en la nube, utilizando una red Wifi, donde se visualiza la información.

3.4.2. Arquitectura

Para la representación de la arquitectura, en la Figura 9 se muestra el diagrama general del sistema, en el cual se define los bloques necesarios para el correcto funcionamiento de sistema de representación de datos.

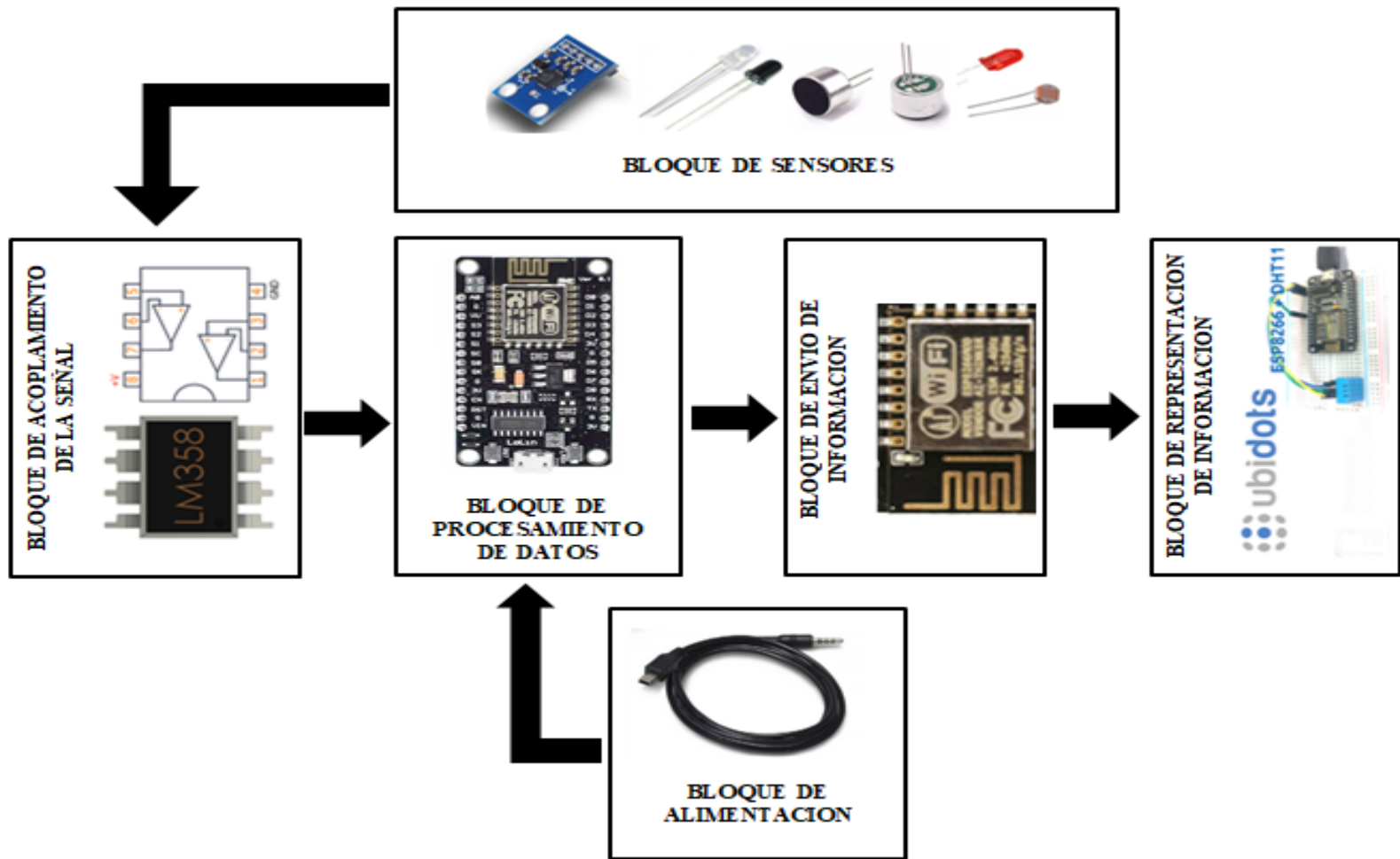


Figura 9. Diagrama de bloques general

Fuente: Autoría

Como se puede observar en la Figura 9, el sistema consta de 6 bloques que aportan al correcto funcionamiento del presente proyecto. A continuación se detalla cada uno de los bloques antes propuestos en el diagrama de bloques.

3.4.2.1. Bloque de alimentación. En este bloque se especifican las fuentes de alimentación tanto para la placa MCU, la cual es el core o núcleo del sistema, como para cada uno de los sensores que se utilizan en el presente.

3.4.2.2. Bloque de sensores. Aquí se encuentran los sensores necesarios para la recolección de datos, estos han sido valorados y seleccionados de tal forma que su funcionamiento sea el más adecuado para el proyecto. Los sensores correctamente ubicados tomarán datos de los pacientes evaluados para el posterior estudio del trastorno.

3.4.2.3. Bloque de acondicionamiento de señal. En este bloque se acondiciona a la señal con el propósito fundamental de suavizar la misma e interpretarla de manera más fácil y precisa. Aquí se encuentran las etapas de filtrado por hardware y amplificación los cuales fueron diseñados y elaborados con diferentes elementos electrónicos, de esta manera se mejora la señal.

3.4.2.4. Bloque de procesamiento. En este bloque se analizan los valores recibidos por medio de los sensores, para posteriormente brindar un resultado preciso. Para el caso en específico es nodeMCU, el cual contiene el algoritmo de aprendizaje de máquina el cual brinda el resultado o valor final.

3.4.2.5. Bloque de envío de información. A través de este bloque se realiza el envío o transmisión de los datos hacia el bloque de representación. Se realiza dicho envío por el módulo Wi-fi integrado en el bloque de procesamiento.

3.4.2.6. Bloque de representación de información. Luego de obtener el resultado generado por el bloque de procesamiento, el bloque de representación es el encargado de interpretar y representar la información resultante. El propósito de ello es hacer más fácil interpretación de la información.

3.5. Diseño

Para el desarrollo del presente se implementara el uso de diferentes dispositivos, para ello es necesario conocer su estructura tales como pines y puertos necesarios para la posterior conexión del sistema.

En este apartado se procede a realizar el diseño del sistema, en el cual consta de los sensores y la placa previamente seleccionados tomando como punto de referencia los requerimientos establecidos para el diseño e implementación del proyecto. El diseño del sistema se ha subdividido en dos partes, el primero el cual está conformado por el diseño del hardware y el segundo referente al diseño del software.

3.5.1. Bloque de acoplamiento y acondicionamiento de la señal. (Bloque 1)

En el presente bloque se realiza el acondicionamiento de las señal para ello se diseña un amplificador de señal debido a que la señal recogida por los sensores está en el rango poco manejable. A continuación se realiza el diseño del circuito.

3.5.1.1. Hardware. La componente pulsátil de manera usual es muy pequeña ya que suele variar entre 0,1mV y 0,8mV como se muestra en la Figura 10, siendo así indetectable para la placa de procesamiento y para el posterior cálculo de las variables.

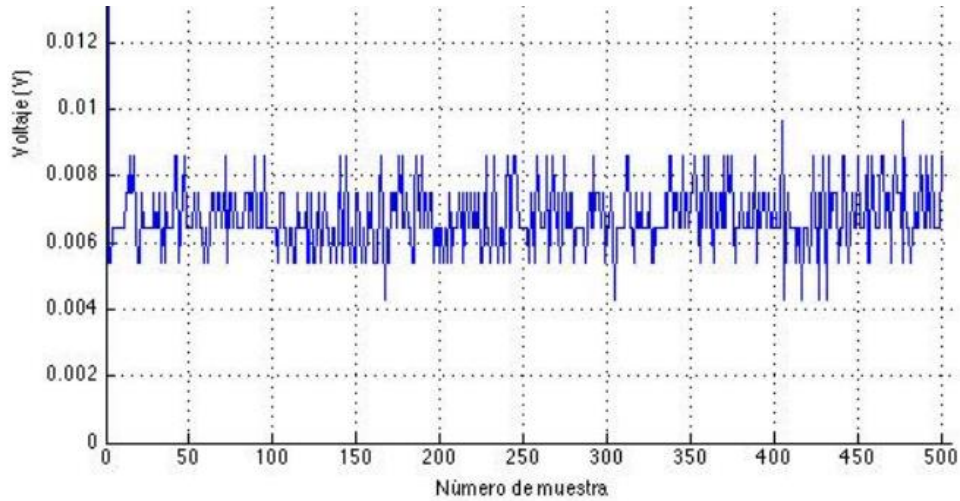


Figura 10. Señal sin amplificador

Fuente: Obtenido de (Moreno, 2015)

Por este motivo es necesario amplificar la señal recibida por el fotodetector, para ello se utiliza el integrado LM358 con una configuración de amplificador no inversor. Donde la tensión de salida responde a la siguiente ecuación.

$$V_{out} = V_{in} \left(1 + \frac{R_2}{R_1} \right)$$

Ecuación 2. Ecuación de tensión de salida

Fuente: Obtenido de (Moreno, 2015)

$$V_{out} = 0,15mV \left(1 + \frac{10K}{100} \right)$$

$$V_{out} = 150mV$$

Para esta configuración se puede realizar diferentes combinaciones de resistencias de las cuales el objetivo principal es obtener como resultado una ganancia de 1000.

A continuación se realiza la implementación de un filtro pasa bajo y pasa alto para ello se utiliza la ecuación que a continuación se plantea.

$$f_{corte\ filtro} = \frac{1}{2\pi R_{34} C_1}$$

Ecuación 3. Ecuación para frecuencia de corte

Fuente: Obtenido de (Moreno, 2015)

El objetivo del filtro pasa bajo es mantener las frecuencias comprendidas entre 0 y 5 HZ, eliminando las demás que se encuentran por encima de este rango. Dicho filtro pasa bajo se logra combinando dos resistencias de 47K y un condensador de 1uF. A continuación se muestran los cálculos realizados.

$$f_{corte\ filtro} = \frac{1}{2\pi \left(\frac{1}{47K} + \frac{1}{47K} \right) 1\mu f}$$

$$f_{corte\ filtro} = 5,7Hz$$

Además, se realiza el cálculo para adicionar un filtro pasa alto, para lo cual se utiliza una resistencia de 10k y un condensador de 10uF. A continuación se observa la respectiva resolución.

$$f_{corte\ filtro} = \frac{1}{2\pi R_5 C_2}$$

$$f_{corte\ filtro} = \frac{1}{2\pi (10k)(10\mu f)}$$

$$f_{corte\ filtro} = 1,03Hz$$

Finalmente se realiza el dimensionamiento del circuito amplificador final para lo cual los valores de resistencias y condensadores se establecieron, como se muestra a continuación en la Figura 11.

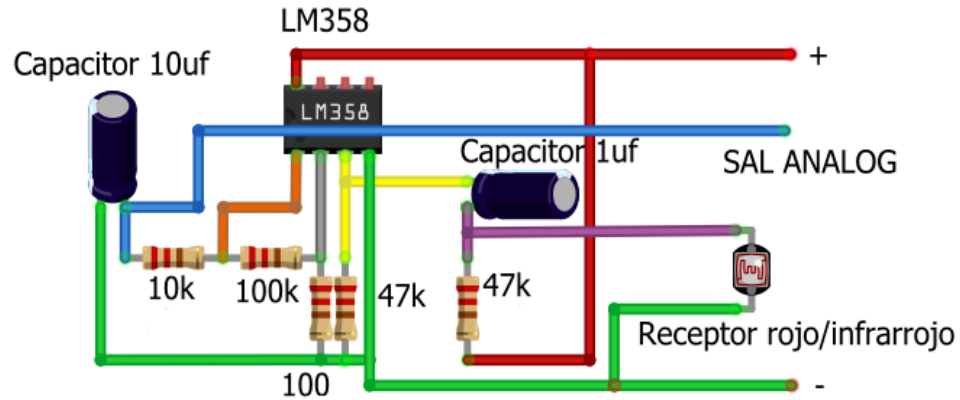


Figura 11. Esquema de amplificador de señal

Fuente: Elaboración propia obtenido de Fritzing

Además, se utiliza un amplificador de audio para el sensor de ronquido antes mencionado. El presente amplifica la amplitud de la señal de audio para así determinar si el paciente posee lapsos de ronquido.

Para el diseño del amplificador necesitamos que V_c , o el voltaje en el colector, sea $1/2 V_{cc}$. Por lo tanto tenemos que poner una resistencia en V_c para dejar $V_c = 1/2 V_{cc}$, y usando la ley de Ohm, $V=IR$, tenemos que:

$$R_c = \frac{V_{cc} - V_c}{I_c}$$

Ecuación 4. Ecuación ley de ohm

Fuente: Obtenido de (Moreno, 2015)

$$R_C = \frac{3V - 1,5V}{0,001A} = 9500\Omega$$

$$R_C = 9,5K\Omega \approx 10K\Omega$$

Hasta el momento, la ganancia del circuito es de $\Delta V_c / \Delta V_e$ que es equivalente a la proporción de R_c / R_e .

$$Ganancia = \frac{R_C}{R_e}$$

Ecuación 5. Ecuación ganancia del circuito

Fuente: Obtenido de (Moreno, 2015)

Ya dejamos $R_c = 4.7 \text{ k}\Omega$, y R_e ya está integrada al transistor. Su R_e se conoce como transresistencia, que se calcula así:

$$r_{tr} = \frac{0,026V}{I_e}$$

$$r_{tr} = \frac{0,026V}{0,01A} = 26\Omega \approx R_e$$

Podemos calcular la ganancia así:

$$Ganancia = \frac{10000\Omega}{26\Omega}$$

$$Ganancia = 384,6$$

Lo que falta es el condensador de entrada para limpiar cualquier compensación DC en la señal de entrada, y mantener el circuito estable. Vamos a fijarla en 0,1uF.

En paralelo, una resistencia y un condensador actúan como un filtro paso alto y, como dijimos, queremos dejarlo en 160 Hz.

$$f = \frac{1}{2\pi RC}$$

Ecuación 6. Ecuación filtro pasa alto

Fuente: Obtenido de (Moreno, 2015)

$$160 = \frac{1}{2\pi R(0,1\mu f)}$$

$$R_1 = 9900\Omega \approx 10K\Omega$$

Para la resistencia de base tenemos como referencia $I=0,01$ y $V=1,5V$. Para ello utilizamos la fórmula de la ley de Ohm.

$$R = \frac{V}{I}$$

Ecuación 7. Ley de OHM

Fuente: Obtenido de (Moreno, 2015)

$$R_B = \frac{1,5V}{0,015mA} = 100k\Omega$$

Finalmente se realiza el dimensionamiento del circuito amplificador final para lo cual los valores de resistencias y condensadores se establecieron de la siguiente forma, como se observa en el diagrama de amplificación de audio en la Figura 12.

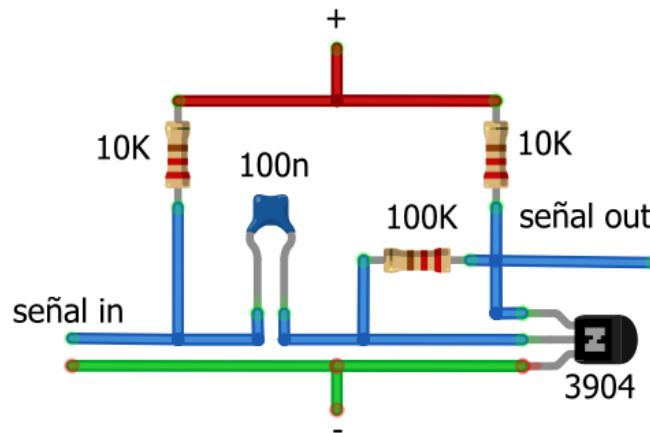


Figura 12. Diagrama de amplificador de audio

Fuente: Elaboración propia obtenido de Fritzing

Luego de establecer correctamente el diseño de los filtros lógicos para el acondicionamiento de la señal, es necesario también establecer la distribución de pines que el hardware de acoplamiento necesita en el sistema. Para ello se utiliza un multiplexor analógico de 16 canales, ya que el núcleo central de procesamiento solo posee un canal analógico, resulta imposible monitorizar múltiples variables sin un medio que lo permita. A continuación se muestra la distribución y conexión necesaria para el correcto funcionamiento del mismo.

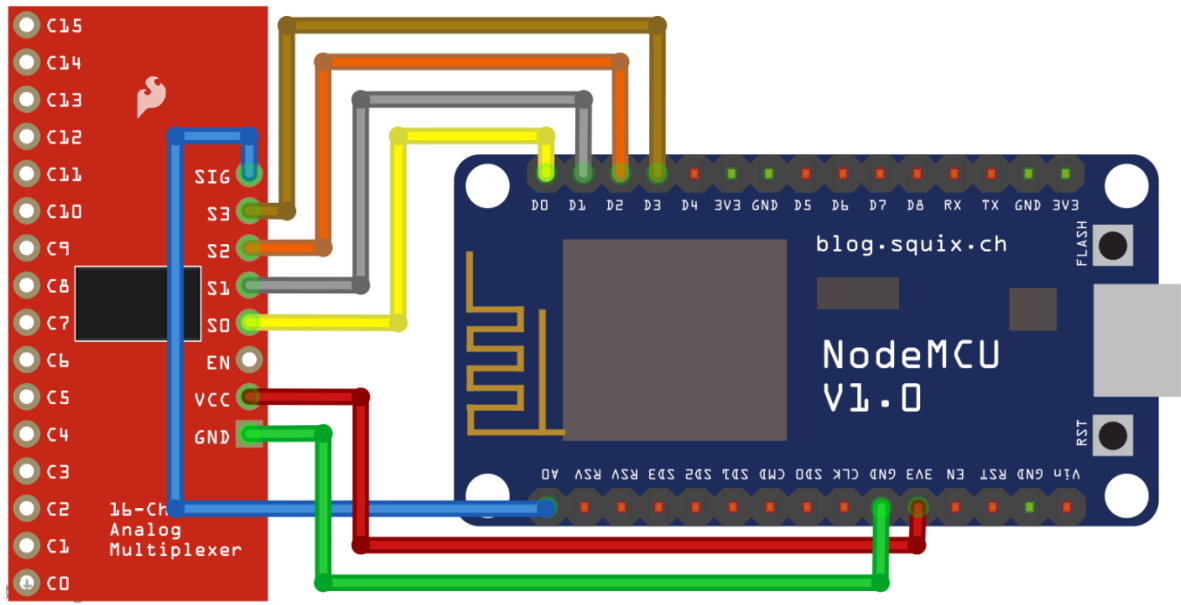


Figura 13: Distribución de pines de Mux analógico

Fuente: Elaboración propia obtenido de Fritzing

Como se observa en la Figura 13, la distribución de pines está diseñada para el funcionamiento adecuado. Siendo así los pines S0, S1, S2 y S3 del multiplexor analógico se conectan directamente a los pines D0, D1, D2 y D3 del NodeMCU de manera correspondiente. Dichos pines garantizan la lectura de diferentes valores en diferentes periodos de tiempo.

Además, SIG es el pin que se conecta directamente al A0 del NodeMCU para la transmisión de los datos recibidos por el multiplexor.

Finalmente, el multiplexor necesita una fuente de energía la cual es brindada por el nodeMCU como se puede observar en la figura.

3.5.1.2. Software. En este caso se verifica la funcionalidad del circuito anteriormente diseñado, para ello utilizamos lenguaje C++ para la obtención de los datos a través del filtro y amplificador. Además se añade un filtro lógico (*Average Moving Filter*), el cual disminuye en

su mayoría el ruido existente. En la Figura 14 se muestra el código utilizado para el filtro antes mencionado.

```
//=====FILTRO LOGICO=====

for(int i =N-1; i>0; i--){           //Sentencia condicional
  valsnore[i]=valsnore[i-1];       //cambia de posicion en el arreglo
}
valsnore[0]=senal;                 //guarda valor en variable
long sumsnore=0;                   //Declara variable
for(int i=0;i<N;i++){              // setencia condicional
  sumsnore=sumsnore+valsnore[i];   // suma valores obtenidos y guarda
}
filteredrsnore=sumsnore/N;         // la suuma total se divide
```

Figura 14: Filtro lógico (Averange Moving)

Fuente: Obtenido de (Arduino, 2018)

A continuación, en la Figura 15 se muestra las señales recibidas sin un filtro lógico establecido.

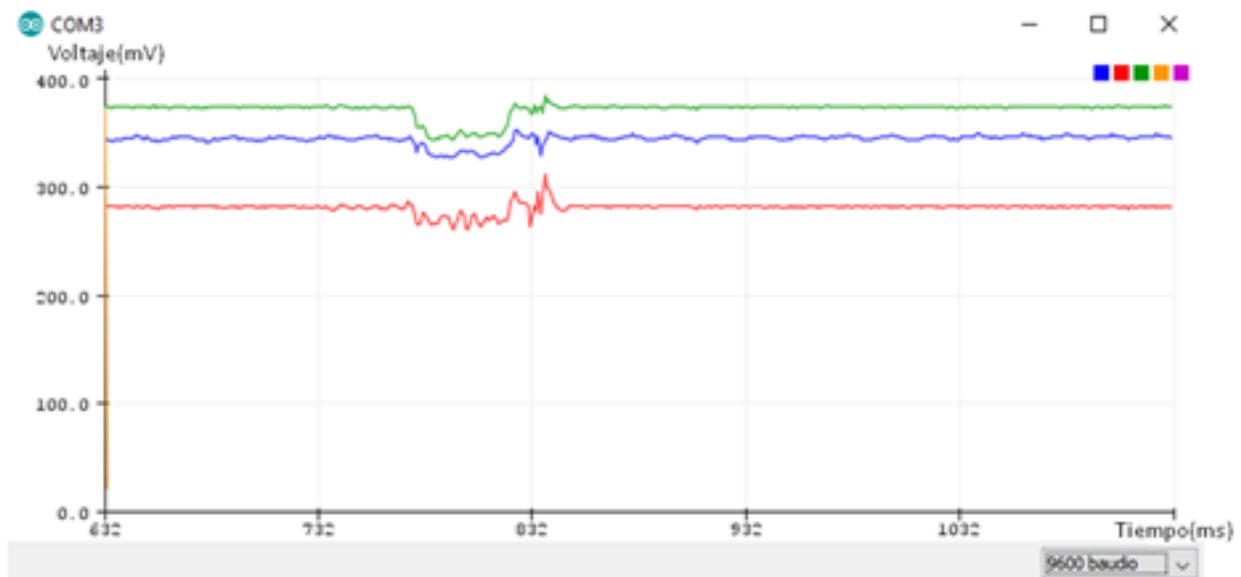


Figura 15: señal captada sin filtro

Fuente: Elaboración propia obtenido de Arduino

Como se puede observar la señal contiene ruido generado por el mismo sistema o por factores externos, afectando los datos recibidos y por ende el resultado esperado. Para suprimir en su mayoría esta presencia de ruido se ve la necesidad de utilizar filtros lógicos que ayuden a suavizar la señal.

Para el presente caso se opta por el principio de funcionamiento del filtro Moving Average y está basado en la Ecuación que a continuación se observa.

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i$$

Ecuación 8. Formula Moving Average

Fuente: Obtenido de (Jiménez, 2018)

El proceso del filtro Moving Average se basa en el promedio de un número de N mediciones, es así que si se aumenta más el valor de N, la probabilidad de la presencia de ruido en la señal es más cercana a cero.

Además, es necesario también establecer el funcionamiento del software de acoplamiento a través del multiplexor analógico. Es por esto que a continuación en la Figura 16 se muestra el código utilizado para un adecuado monitoreo.

```

Serial.begin(9600);
pinMode(A,OUTPUT);// salida conectado a A/S0 pin 11 multiplexor
pinMode(B,OUTPUT);// salida conectado a B/S1 pin 10 multiplexor
pinMode(C,OUTPUT);// salida conectado a C/S2 pin 09 multiplexor
pinMode(D,OUTPUT);// salida conectado a D/S3 pin 08 multiplexor
pinMode(A0,INPUT);// Pin analogico A0 NodeMCU

//=====Lectura de Multiplexor=====

digitalWrite(A, LOW);//Apagar Pin
digitalWrite(B, LOW);//Apagar Pin
digitalWrite(C, LOW);//Apagar Pin
digitalWrite(D, LOW);//Apagar Pin
Entrada[0] = analogRead(A0); // Valor de ronquido
digitalWrite(A, HIGH);//Encender Pin
digitalWrite(B, LOW);//Apagar Pin
digitalWrite(C, LOW);//Apagar Pin
digitalWrite(D, LOW);//Apagar Pin
Entrada[1] = analogRead(A0); // Valor luz roja
digitalWrite(A, LOW);//Apagar Pin
digitalWrite(B, HIGH);//Encender Pin
digitalWrite(C, LOW);//Apagar Pin
digitalWrite(D, LOW);//Apagar Pin
Entrada[2] = analogRead(A0); // Valor de luz infrarroja
digitalWrite(A,HIGH);//Encender Pin
digitalWrite(B,HIGH);//Encender Pin
digitalWrite(C,LOW);//Apagar Pin
digitalWrite(D,LOW);//Apagar Pin
Entrada[3] = analogRead(A0); // Valor de acelerometro "X"
digitalWrite(A,LOW);//Apagar Pin
digitalWrite(B,LOW);//Apagar Pin
digitalWrite(C,HIGH);//Encender Pin
digitalWrite(D,LOW);//Apagar Pin
Entrada[4] = analogRead(A0); // Valor de acelerometro "Y"
digitalWrite(A,HIGH);//Encender Pin
digitalWrite(B,LOW);//Apagar Pin
digitalWrite(C,HIGH);//Encender Pin
digitalWrite(D,LOW);//Apagar Pin
Entrada[5] = analogRead(A0); // Valor de acelerometro "Z"
digitalWrite(A, HIGH);//Encender Pin
digitalWrite(B, LOW);//Apagar Pin
digitalWrite(C, HIGH);//Encender Pin
digitalWrite(D, HIGH);//Encender Pin
Entrada[13] = analogRead(A0); // Valor de respiracion

```

Figura 16: Código multiplexor analógico

Fuente: Obtenido de (Arduino, 2018)

3.5.1.3. Diagrama de flujo. En este apartado se muestra el flujo de diagrama del bloque de acondicionamiento y acoplamiento de la señal. El principal objetivo es esquematizar el presente bloque de manera conceptual. A continuación, se muestra en la Figura 17.

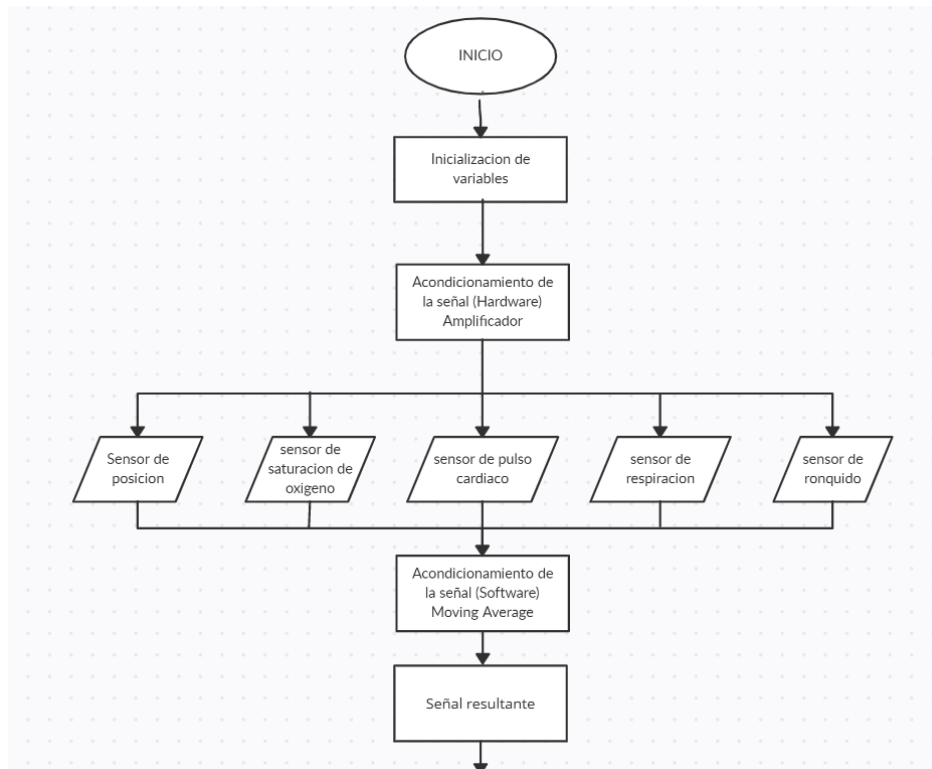


Figura 17. Diagrama de flujo del bloque 1

Fuente: Elaboración propia obtenido de creately.com

3.5.2. Bloque de sensores (Bloque 2).

3.5.2.1. Hardware.

- **Diagrama de conexión-Posición del cuerpo.** Para medir este parámetro se utiliza el sensor de posición del cuerpo, mediante el cual se establecerán los patrones de movimiento del paciente. Los pines del acelerómetro X, Y, Z se conectan directamente a

los pines C3, C4 Y C5 correspondientemente. Además, se conectan los pines de alimentación VCC Y GND al sistema, según corresponda. A continuación, en la Figura 18 se muestra el diagrama de conexión del sensor con la placa NodeMCU.

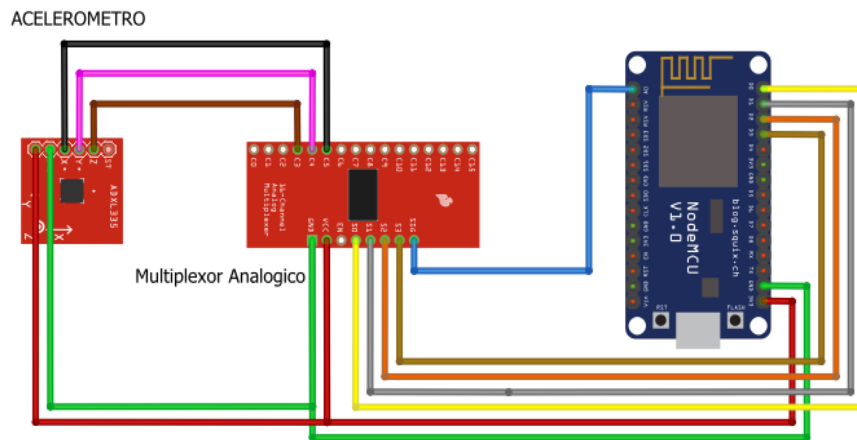


Figura 18. Diagrama de conexión de posición del cuerpo

Fuente: Elaboración propia obtenido de Fritzing

El presente sensor obtendrá datos poco entendibles para el usuario por lo que es necesario establecer valores fáciles de interpretar en cada una de las posiciones del individuo somnoliento. Los valores correspondientes se detallan a continuación en la Tabla 15.

Tabla 15. Asignación de valores a cada posición.

**POSICIÓN DEL CUERPO BASADO EN VALORES DE
ACELERÓMETRO**

Valor obtenido			Posición	Valor asignado
X	Y	Z		
480 a 585	0 a 1024	630 a 660	Boca arriba	1
610 a 640	0 a 1024	470 a 610	Giro derecha	2
420 a 460	0 a 1024	470 a 610	Giro izquierda	3
480 a 585	0 a 1024	0 a 460	Boca abajo	4

Fuente: Elaboración propia

- **Diagrama de conexión-Sensor OXIMETRIA.** Para el caso del parámetro de oximetría se utiliza el sensor de oximetría, para la verificación de su conexión se observa en la Figura 19 el diagrama de conexión con la placa de procesamiento NodeMCU. Previo a este esquema se utiliza el diagrama de amplificación situado en el bloque de acoplamiento de la señal, misma que se muestra en la Figura 11. La resente conxion se realiza de la siguiente manera: pines CI Y C2 del multiplexor analógico se conecta directamente a las salidas IR Y R del sensor diseñado. Además, se conecta la alimentación del sensor según corresponda.

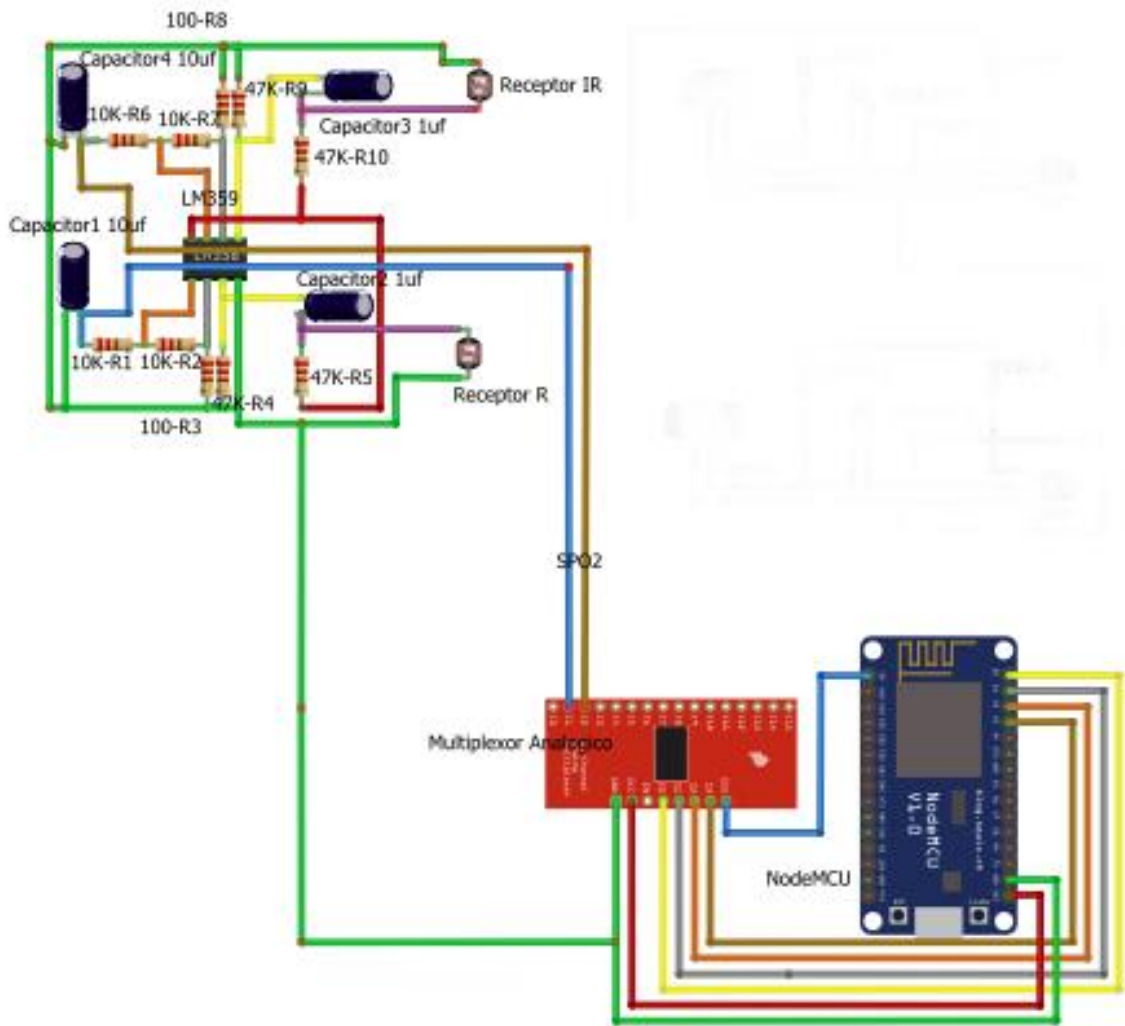


Figura 19. Diagrama de conexión sensor de OXIMETRIA

Fuente: Elaboración propia obtenido de Fritzing

- **Diagrama de conexión interna de Sensor OXIMETRIA.** Para el presente proyecto se realiza la implementación del sensor de fabricación propia, para ello se realiza la conexión que se muestra a continuación en la Figura 20.

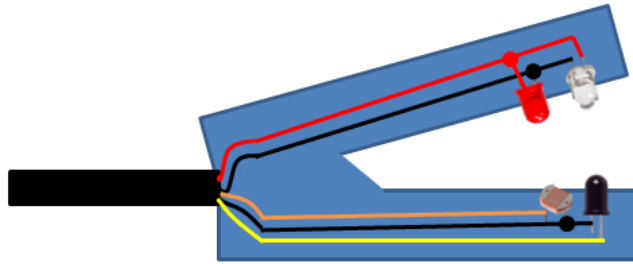


Figura 20. Diagrama de conexión interno SPO2

Fuente: Elaboración propia

- **Diagrama de conexión-medición de ronquido.** Para el sensor de ronquido se utilizan el pin C0 del multiplexor analógico y el pin A0 de la placa de procesamiento, como a continuación se detalla en la Figura 21. Para el correcto funcionamiento del presente sensor se acopla un amplificador de audio, situado en el esquema de acoplamiento de la señal, misma que se observa en la Figura 12.

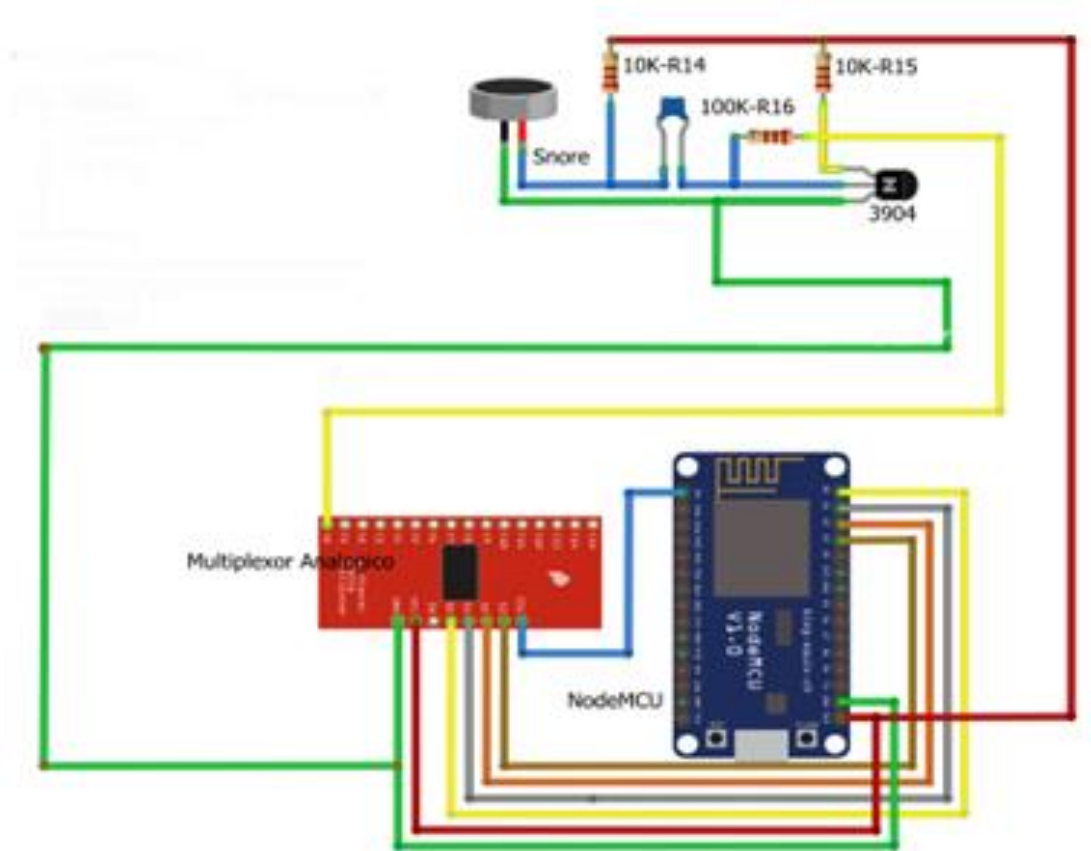


Figura 21. Diagrama de conexión de sensor de ronquido

Fuente: Elaboración propia obtenido de Fritzing

- **Diagrama de conexión-medición de Flujo de aire.** Para el sensor de flujo de aire se utilizan el pin C13 del multiplexor analógico y el pin A0 de la placa de procesamiento, como a continuación se detalla en la Figura 22. Además, se acopla al amplificador de señal mismo que se observa en la Figura 12.

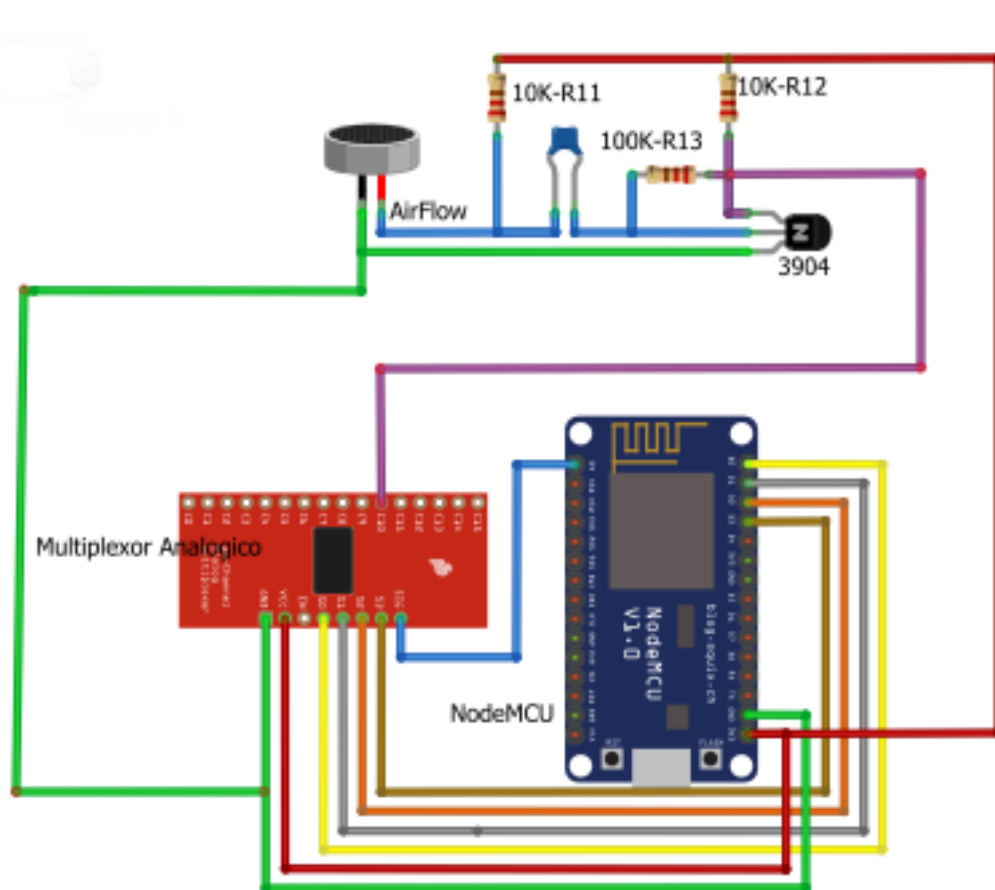


Figura 22. Diagrama de conexión de sensor de flujo de aire

Fuente: Elaboración propia obtenido de Fritzing

3.5.2.2. Software.

Para el caso particular del software en el bloque de sensores, es conveniente mencionar que cada uno de los sensores utiliza el mismo procedimiento para la lectura de los datos con la única diferencia que se utiliza diferentes variables de manera correspondiente. Para ello a continuación, en la Figura 23 se muestra el código utilizado para las lecturas de los sensores.

```

int lectura= A0;      //inicio de variables
float senal=0;       //inicio de variables
int lectural= A1;    //inicio de variables
float senall=0;      //inicio de variables
#define N 10         //inicio de variables
int filteredsnore;   //inicio de variables
int valsnore[N];     //inicio de variables

void setup()
{
  Serial.begin(9600);      //inicio de sistema en 9600 baudios
  pinMode(lectura,INPUT);  // establece modo de trabajo del PIN
}

void loop(){

float senal= analogRead(A0); //asigna valor dado por el sensor
float senall= analogRead(A1); //asigna valor dado por el sensor
//=====FILTRO LOGICO=====
  for(int i =N-1; i>0; i--){
    valsnore[i]=valsnore[i-1];
  }
  valsnore[0]=senal;
  long sumsnore=0;
  for(int i=0;i<N;i++){
    sumsnore=sumsnore+valsnore[i];
  }
  filteredsnore=sumsnore/N;
//=====

Serial.println(filteredsnore);    //Imprime valor en grafico
}

```

Figura 23. Lectura de datos a través de sensores

Fuente: Elaboración propia obtenido de Arduino.

3.5.2.3. Diagrama de flujo.

A continuación, en la Figura 24 se muestra el diagrama de flujo del bloque 2, en el cual constan los sensores y el bloque de acoplamiento.

Fuente: Elaboración propia obtenido de creately.com

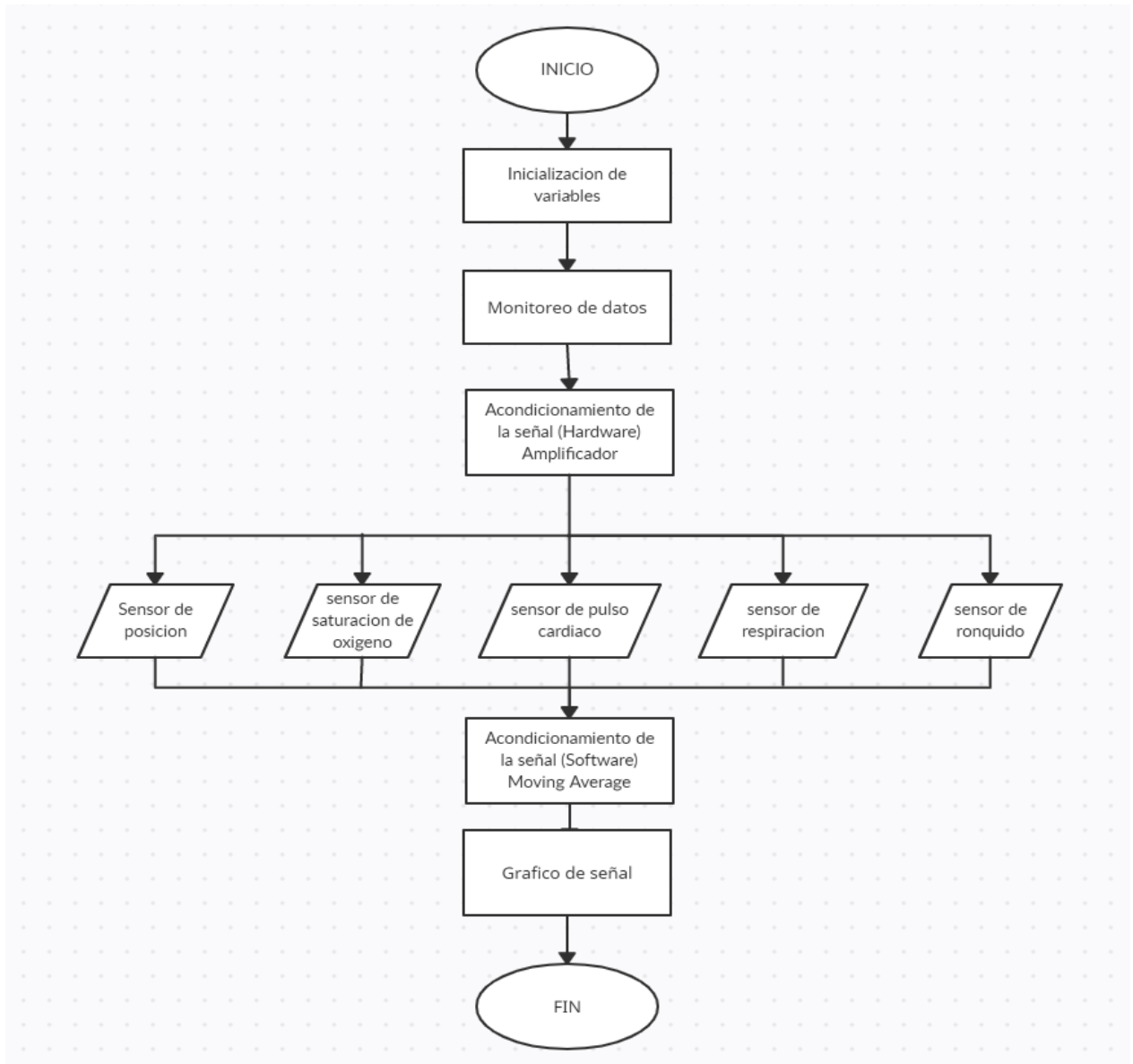


Figura 24. Diagrama de flujo de bloque 2

3.5.3. Bloque de procesamiento (Bloque 3)

3.5.3.1. Software: Después de haberse establecido el módulo MCU como núcleo central de procesamiento del sistema, también se realiza un proceso lógico a través del mismo, por esta razón se diseña el siguiente código en base a las necesidades del proyecto. A continuación, se muestra en la Figura 25 el código utilizado.

```
//=====RESTRICCIONES DE CONTEO=====
//*****Ronquido*****
if (filtradosnore>=290&&valor_ant<290||filtradosnore<=210&&valor_ant>210){ //condicion de limites
    num_ronquido=num_ronquido+1; //contador ascendente
}
valor_ant=filtradosnore; //guarda valor en variable
//*****SPO2*****
suma_senales=filtradoR+filtradoIR; //suma de señales IR y R
promedio=(filtradoR/suma_senales)*100; //Resultado de ecuacion spo2
suma_promedio=suma_promedio+promedio; //almacena la suma de resultados
//*****Pulsímetro*****
if (filtradoIR>=300&&valoranteriorIR<300){ //condicion de limites
    num_pulsos=num_pulsos+1; //contador ascendente
}
valoranteriorIR=filtradoIR; //guarda valor en variable
//*****Posicion*****
if (filtradox>=460&&filtradox<=550){ //condicion de limites
    if (filtradoz>=570&&filtradoz<=660){ //condicion anidada de limites
        posicion=1; //asigna valor de posicion
        sprintf(contexto, "ARRIBA"); //envio de respuesta en contexto
    }
    if (filtradoz<460){ //segunda condicion anidada de limites
        posicion=4; //asigna valor de posicion
        sprintf(contexto, "ABAJO"); //envio de respuesta en contexto
    }
}
if (filtradoz>=500&&filtradoz<=550){ //condicion de limites
    if (filtradox>=425&&filtradox<=440){ //condicion anidada de limites
        posicion=3; //asigna valor de posicion
        sprintf(contexto, "IZQUIERDA"); //envio de respuesta en contexto
    }
    if (filtradox>630&&filtradox<640){ //segunda condicion anidada de limites
        posicion=2; //asigna valor de posicion
        sprintf(contexto, "DERECHA"); //envio de respuesta en contexto
    }
}
}
```

```

//*****Respiracion*****
if(filtraresp>=235&&resp_anterior<235){ //condicion de limites
    num_respiracion=num_respiracion+1; //contador ascendente
}
resp_anterior=filtroresp; //guarda valor en variable
delay(10); //delay de espera
}

//=====Almacenamiento y Procesos=====

total_ronquido=total_ronquido+num_ronquido; //almacenamiento de valores ronquido en variable

total=suma_promedio/b; //almacenamiento de valores spo2 en variable
sumatotal=sumatotal+total; //almacenamiento spo2 en 30 min

sum_pulsos=sum_pulsos+num_pulsos; //almacenamiento de valores de pulso

total_resp=total_resp+num_respiracion; //almacenamiento de valores de respiracion
contador=contador+1; //Contador ascendente

if(contador==5){ //condicion de tiempo
    ronquido=(total_ronquido)*1.2; //producto de formula
    SPO_2=(sumatotal/5)+36; //guarda resultado de SPO2
}

pulso_total=(sum_pulsos+5); //guarda resultado de pulsionometria

if(sum_pulsos<=20){ //condicion de impresion
    SaturacionO2=0; //asignacion de valor
}
else{ //de lo contrario
    SaturacionO2=SPO_2; //asignacion de valor
}
respiracion=(total_resp); //guarda valor

//=====ALGORITMO KNN=====

for (int l=0; l < 353; l++) {
    pow1 = pow(((BDD[l][0]) - pulso_total), 2); // guardar valor de resta de nuevo dato de la BDD
    pow2 = pow(((BDD[l][1]) - SPO_2), 2); // guardar valor de resta de nuevo dato de la BDD
    pow3 = pow(((BDD[l][2]) - respiracion), 2); // guardar valor de resta de nuevo dato de la BDD
    pow4 = pow(((BDD[l][3]) - posicion), 2); // guardar valor de resta de nuevo dato de la BDD
    pow5 = pow(((BDD[l][4]) - valor_r), 2); // guardar valor de resta de nuevo dato de la BDD
    dist = sqrt(pow1 + pow2 + pow3 + pow4 + pow5); //calcula de distancia

    if (dist < distFin) { // comparacion de distancias
        distFin = dist; // almacenamiento de distancia en posicion j
        etiq = BDD[l][5]; //asignacion de valor a etiqueta
    }
}
}

```

Figura 25: Código de bloque de procesamiento

Fuente: Obtenido de (Arduino, 2018)

En la Figura 25 se puede observar el procesamiento que realiza el bloque de procesamiento. Dicho proceso se subdivide en etapas las cuales son restricciones de conteo, almacenamiento y proceso y algoritmo KNN.

En la etapa denominada restricciones de conteo se realiza el proceso mediante el cual se tomen solo datos validos durante el proceso de monitoreo, de esta manera descartando datos inválidos y disminuyendo carga al núcleo procesador. A continuación en la etapa de almacenamiento y procesos se guardan los datos capturados en diferentes variables con el objetivo principal de luego procesar según sea necesario. Finalmente en la etapa del algoritmo KNN se utilizan los datos antes obtenidos y almacenados para la clasificación final del sistema.

3.5.4. Bloque de Alimentación. (Bloque 4)

En este bloque se define la fuente de alimentación para la placa de procesamiento y los respectivos sensores. Para ello se necesita calcular el amperaje necesario total para el sistema. Los requerimientos para cada componente fueron obtenidos de los respectivos datasheets mismos que se observan en los Anexo 5, Anexo 6 y Anexo 7.

3.5.4.1. Hardware. Para el diseño del hardware del bloque de alimentación. De manera general se realiza la suma total de los requerimientos de los componentes, para lo cual se propone la ecuación que se muestra a continuación.

$$I_{total} = I_{ledR} + I_{ledIR} + (I_{micro} * 3) + I_{accel} + (I_{lm358} * 2) + I_{wifi}$$

Ecuación 9: Suma total de requerimientos de sensores

Fuente: elaboración propia obtenida de (Burbano Y., 2018)

Como requerimiento general del led rojo de alta luminosidad según su respectivo datasheet se establece un voltaje $V_{ledR} = 2V$ y una corriente $I_{ledR} = 20mA$. Por otro lado para el led infrarrojo se obtiene requerimientos de funcionamiento $V_{ledIR} = 2V$ y una corriente $I_{ledIR} = 100mA$.

Para el caso de los sensores de ronquido y flujo de aire se utiliza micrófonos para ello los requerimientos están establecidos con un voltaje $V_{micro} = 3V$ y una corriente $I_{micro} = 0,5mA$. Además, los valores establecidos para el acelerómetro se encuentran establecidos en un voltaje $V_{accel} = 3V$ y una corriente necesaria $I_{accel} = 350uA$.

Para la etapa de amplificación se utiliza operacionales mismos que utilizan valores de voltaje $V_{lm358} = 3V$ y la corriente $I_{lm358} = 1.2mA$. Finalmente para la transmisión o envío de los datos el módulo Wi-fi utiliza $V_{wifi} = 3,3v$ y $I_{wifi} = 135mA$.

A continuación se muestra el cálculo de la intensidad de corriente necesaria para todo el sistema, para lo cual utilizamos la Ecuación 2 ya antes planteada.

$$I_{total} = 20mA + 100mA + (0,5mA * 3) + 0,35mA + (1,2mA * 2) + 135mA$$

$$I_{total} = 259,25mA$$

En el caso específico del voltaje requerido se puede observar en las especificaciones ya planteadas que todos los elementos funcionan con un voltaje promedio de 3.3V por lo cual la

selección del módulo amplificador de voltaje de 3-5V es únicamente por requerimiento del nodeMCU el cual necesita 5V para su correcto funcionamiento. Sin embargo, cabe aclarar que cada uno de los sensores y elementos están conectados al nodeMCU el cual brinda únicamente 3V de esta manera satisfaciendo la necesidad de cada dispositivo conectado al mismo.

Como se muestra en la Figura 26, se utiliza una batería de litio de 3V, la cual estarán conectada a un módulo de carga y posteriormente a un módulo de amplificación para finalmente conectarse directamente a la placa central.

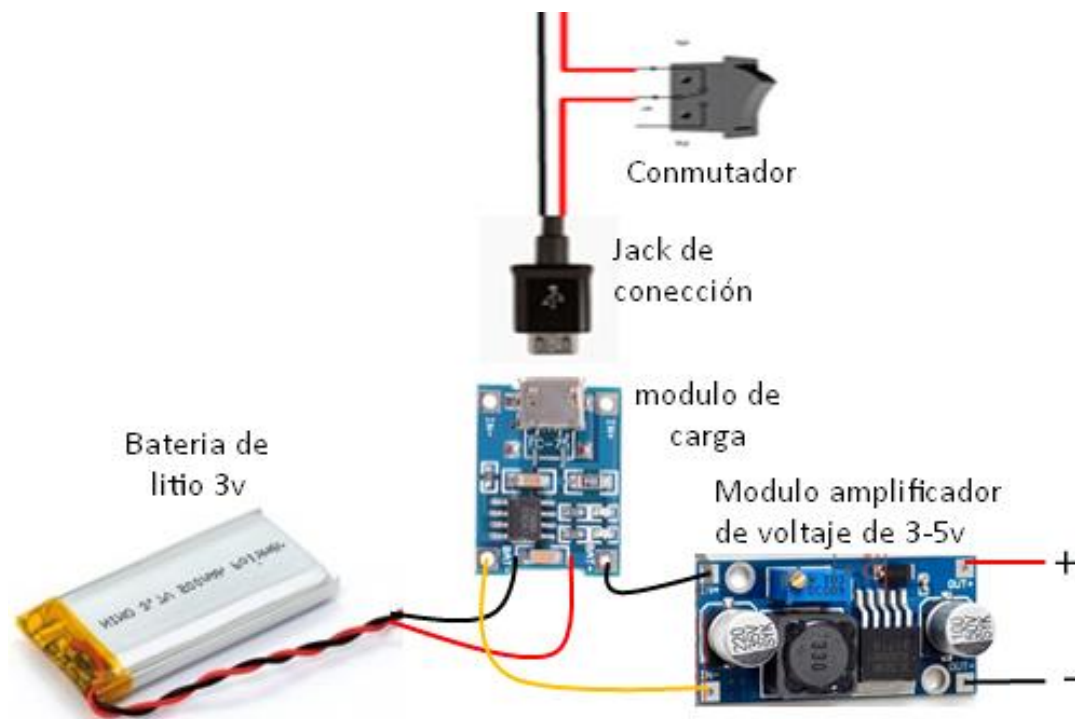


Figura 26. Diagrama de conexión de alimentación

Fuente: Elaboración Propia

3.5.4.2. Diagrama de Flujo

Para mejor interpretación del funcionamiento del bloque de alimentación. A continuación, se presenta en la Figura 27 el diagrama de flujo del funcionamiento básico de dicho bloque.

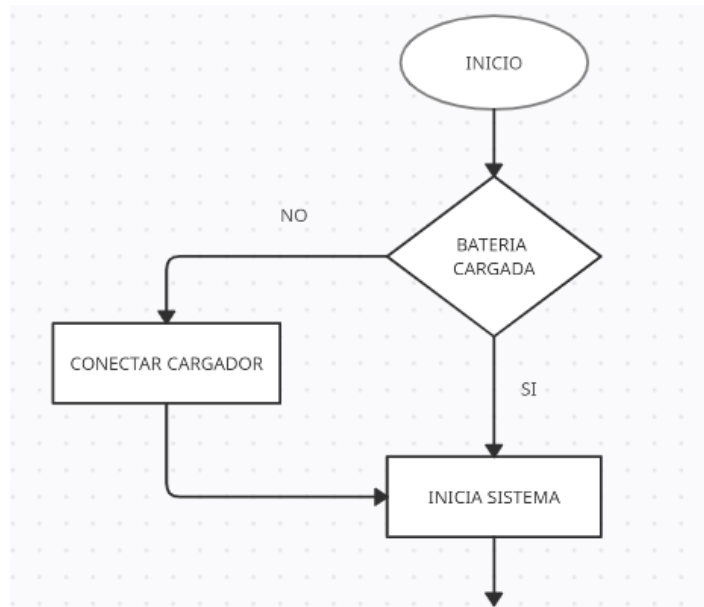


Figura 27. Diagrama de flujo del bloque 3

Fuente: Elaboración propia obtenido de creately.com

3.5.5. Bloque transmisión de datos. (Bloque 5)

Para el envío de los datos anteriormente evaluados y correctamente tratados, se ve la necesidad de utilizar un medio de transmisión el cual permita transmitir hacia la nube para posteriormente representarlos a través de gráficos.

3.5.5.1. Software. Anteriormente se establece la presencia de un módulo de transmisión (WiFi) en el bloque de procesamiento, por lo cual no es necesario realizar ningún diseño de hardware en el presente bloque. Sin embargo, es importante realizar el diseño de software para su correcto funcionamiento. A continuación, se presenta en la Figura 28 el código utilizado.

```
//=====envio a ubidots=====

WiFi.mode(WIFI_STA);
WiFi.begin(ssid, password);
Serial.print("Conectando a:\t");
Serial.println(ssid);

// Esperar a que nos conectemos
while (WiFi.status() != WL_CONNECTED)
{
    delay(200);
    Serial.print('.');
}

//===== Mostrar mensaje de éxito y dirección IP asignada=====
Serial.println();
Serial.print("Conectado a:\t");
Serial.println(WiFi.SSID());
Serial.print("IP address:\t");
Serial.println(WiFi.localIP());
```

Figura 28. Programa de funcionamiento de módulo WiFi

Fuente: Elaboración propia obtenido de Arduino

3.5.6. Bloque representación de datos. (Bloque 6)

En este bloque se realiza el proceso de recepción de los datos que fueron enviados a través del sistema inteligente. Para el bloque de recepción existen diferentes plataformas que

pueden ser muy útiles para este caso en específico. Sin embargo, no todas se ajustan a las exigencias de los requerimientos establecidos por el sistema.

3.5.6.1. Software.

Para este caso se toma como las plataformas más optimas a ThingSpeak, Ubidots y Cayenne debido a sus principales características entre las cuales resaltan la interfaz gráfica amigable con el usuario, plataformas de código y su uso libre de pago (Kalaithasan, Radzi, & Abidin, 2018). A continuación en la Tabla 16 se observa la comparación más a detalle.

Tabla 16. Comparación de plataformas de recepción de datos

Comparación entre ThingSpeak, Ubidots and FavorIoT			
Aspectos	Plataformas IoT		
	ThingSpeak	Ubidots	Cayenne
Interfaz de usuario y Presentación de datos	Bueno	Interfaz de usuario pesada, muy buenas opciones de representación de gráficos.	Bueno
	Sujeto a la velocidad de la red, se esperan retrasos significativos como cada sensor tiene su propia ID que es para ser codificado por separado.	Se adapta a la velocidad de la red, permitiendo la carga simultánea de datos por lo tanto no hay demora significativa.	Sujeto a la velocidad de la red, permite carga de datos simultánea por lo tanto no hay retraso significativo.
complejidad de código	Codificación compleja, la complejidad aumenta con más sensores, a medida que aumenta se observan errores y	Conocimiento simple y básico de codificación es suficiente. No mucha codificación involucrada para subir	Conocimiento simple y básico de codificación es suficiente. No mucha codificación involucrada para subir

	errores de red que tiene que ser depurados.	datos.	datos.
Características	<ul style="list-style-type: none"> •La biblioteca permite carga simultánea de datos de los sensores relacionados, por lo tanto menor retraso. •Las herramientas analíticas son disponibles para realizar análisis en mayor profundidad. •Los datos se pueden exportar a un Archivo .CSV para hoja de cálculo representación. 	<ul style="list-style-type: none"> •Lista de flujo de datos disponible para cada sensor de datos que es subido •Los datos se pueden exportar a un CSV archivo para hoja de cálculo representación. •Herramientas analíticas disponibles con una variedad de gráficos y opciones de representación. •Compatible con codificación múltiple de idiomas. 	<ul style="list-style-type: none"> •Carga simultánea de datos del sensor es posible con algunos ajustes para los códigos proporcionados. •La lista de flujo de datos está disponible para cada sensor de datos que es subido. •Compatible con 3 codificaciones lenguajes, python, HTML y JavaScript. •Los proyectos se pueden gestionar por completo a través de la página con acceso autorizado
Costo de servicio	Servicio gratuito limitado, y características adicionales pagadas.	Servicio gratuito en desarrollo, y características adicionales pagadas.	Servicio gratuito, y características adicionales pagadas.

Fuente: Elaboración propia obtenido de (Kalaithasan et al., 2018)

De acuerdo a la comparación realizada anteriormente se puede concluir, que la plataforma “Cayenne” es la opción más adecuada debido a que todas las características planteadas satisfacen los requerimientos tanto del sistema como del usuario.

Los datos a recibir en la plataforma antes mencionada son enviados a través del bloque de transmisión, utilizando el código en arduino que a continuación se detalla en la Figura 29.

```
//=====IMPRESION Y ENVIO=====

Serial.print("DATA,TIME,");          //envia datos de fecha y hora
client.add ("pulso cardiaco", pulso_total); //envia valor de variable 1
client.add ("SPO2", SaturarO2);      //envia valor de variable 2
client.add ("respiracion", respiracion); //envia valor de variable 3
client.add ("snore", ronquido);       //envia valor de variable 4
client.add ("posicion", posicion);    //envia valor de variable 5
```

Figura 29: Código de envío desde Arduino

Fuente: Obtenido de (Arduino, 2018)

Cada uno de los datos enviados contienen una etiqueta con la cual serán reconocidos en la plataforma de destino, además es importante tener en cuenta la fecha en la cual el sistema a realizado la monitorización al usuario, es por esto que también se añade esta información al sistema.

Luego de enviarse los datos, la plataforma se encargara de recibir, identificar y graficar cada uno de los datos establecidos. A continuación en la Figura 30, se muestran los datos recibidos y representados en Cayenne.

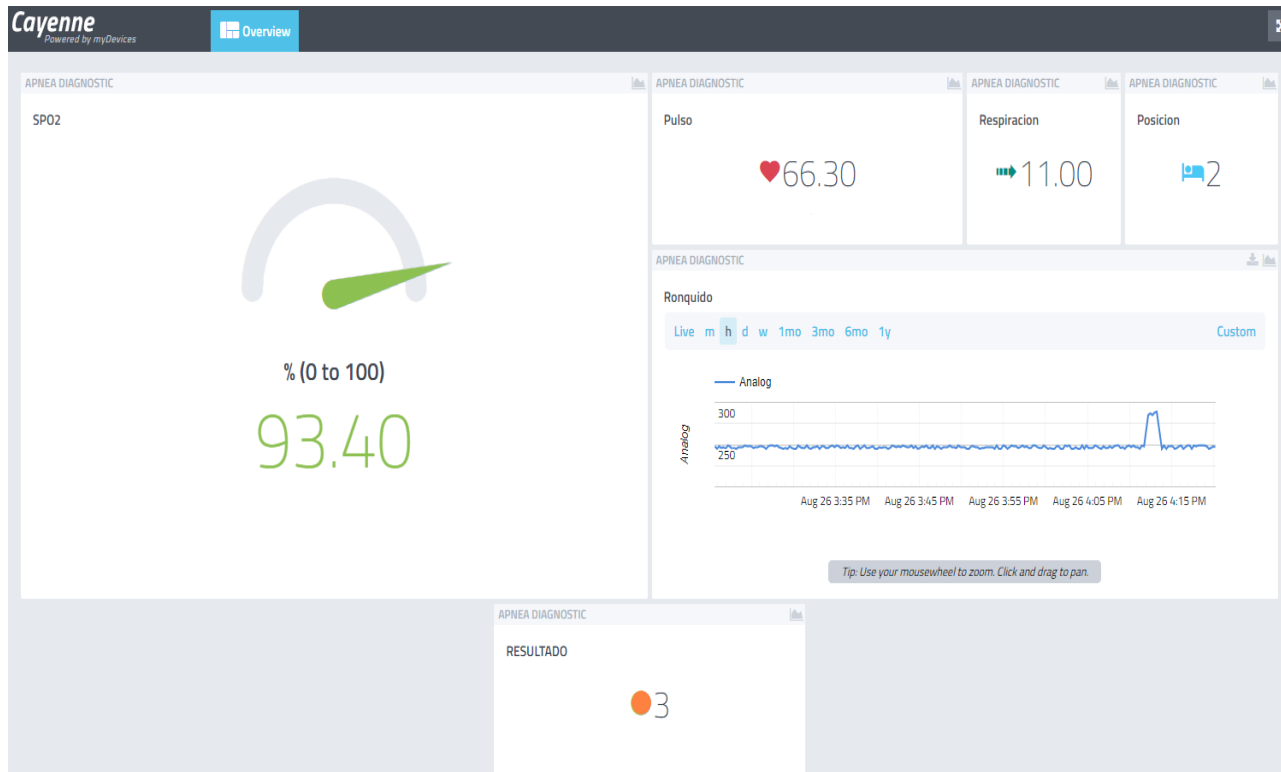


Figura 30. Representación de datos recibidos por Cayenne en tiempo real

Fuente: Elaboración propia obtenido de Cayenne

3.5.7. Diseño general del sistema.

Aquí se integran todo lo necesario para el correcto funcionamiento del proyecto, tomando en cuenta cada uno de los bloques especificados en el diagrama de bloques general, entre los mismos se encuentran: bloque de alimentación, bloque de sensores, bloque de acondicionamiento de la señal, bloque de procesamiento y el bloque de representación, como se muestra en la Figura 31 del diagrama circuital del sistema.

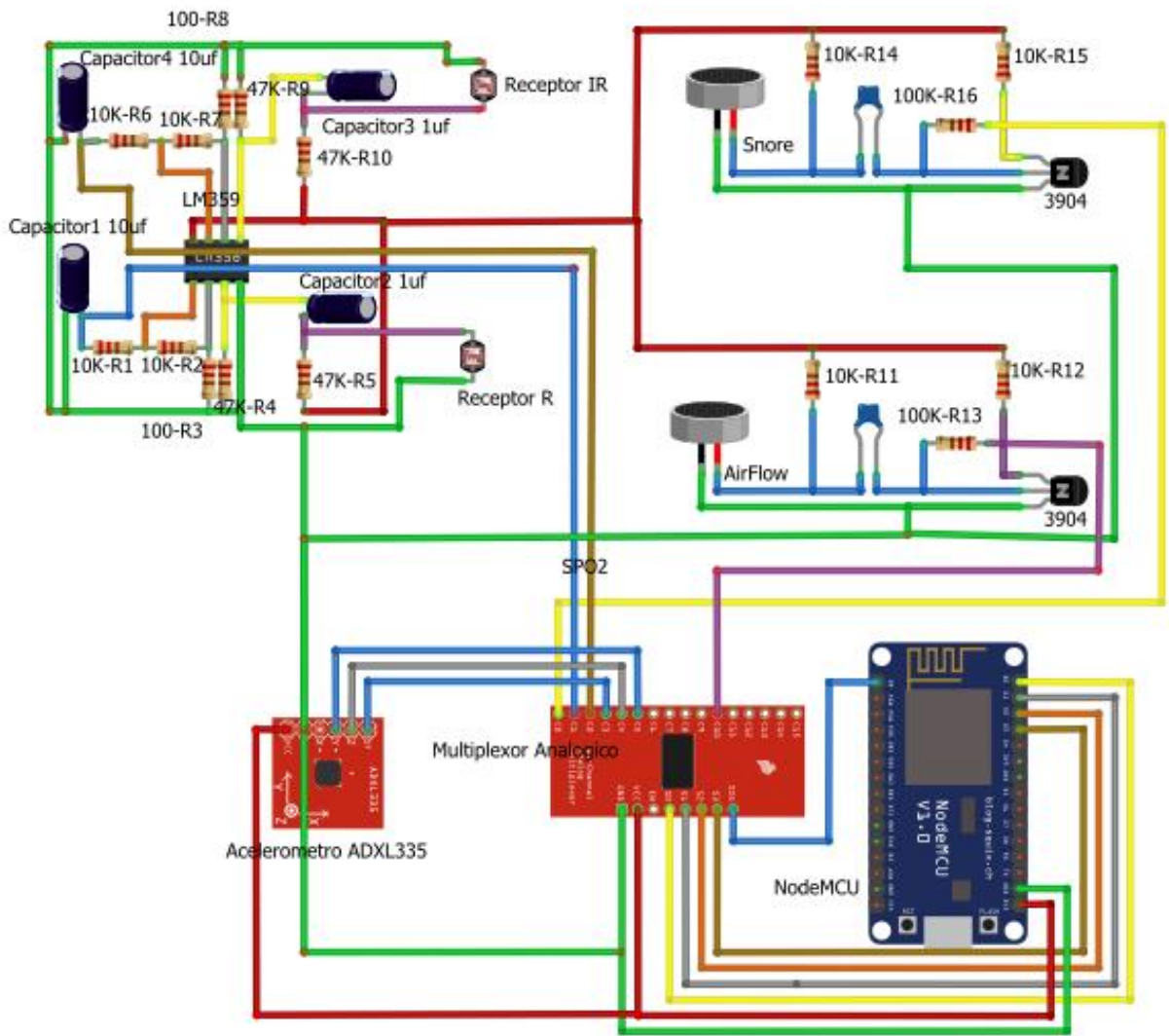


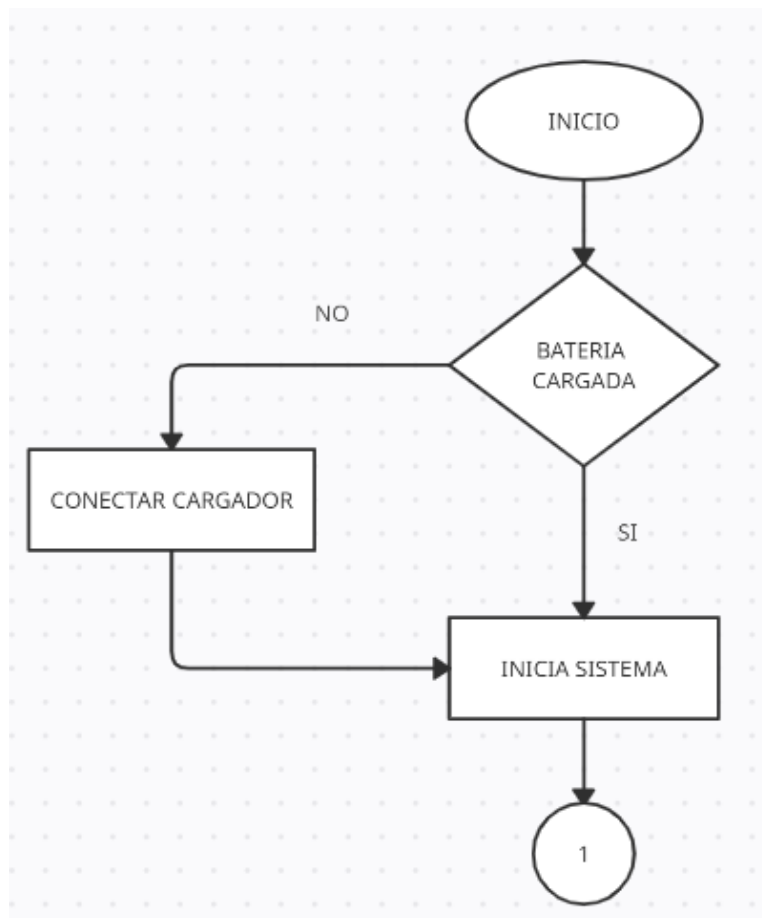
Figura 31. Diagrama circuital del sistema

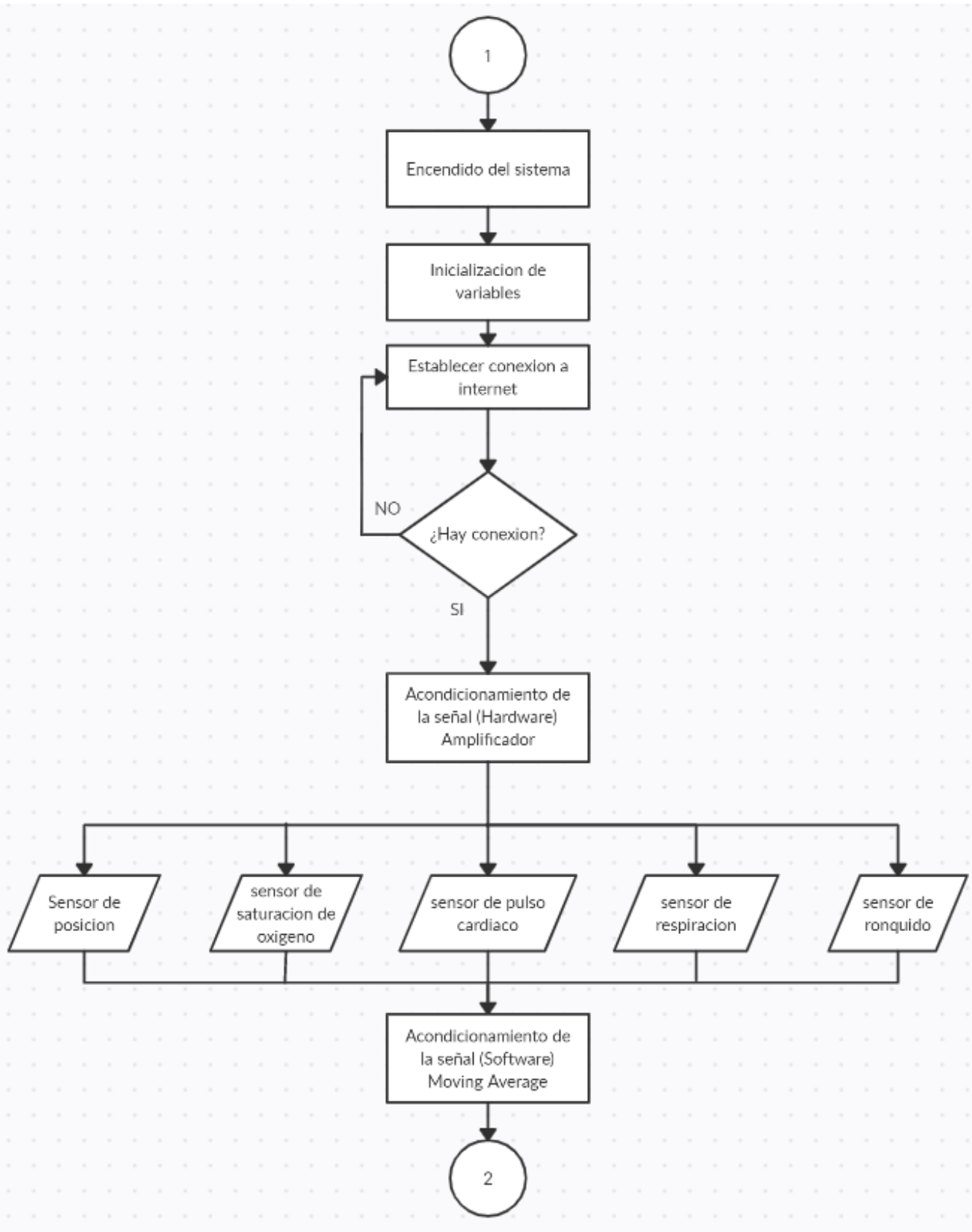
Fuente: Elaboración propia obtenido de Fritzing

3.5.8. Diagrama de flujo general del sistema.

Es necesario realizar el análisis de la lectura de cada uno de los sensores que se implementan en el presente proyecto. Posterior a dicho análisis se elabora el diagrama de flujo del sistema con el objetivo de establecer y explicar el proceso específico que cumple el sistema con ayuda del lenguaje de programación asignado al mismo.

En la Figura 32, se muestra el proceso que se cumple, a través de un flujograma, en el cual se establecen condiciones y asignaciones de valores a las diferentes variables utilizadas en el programa.





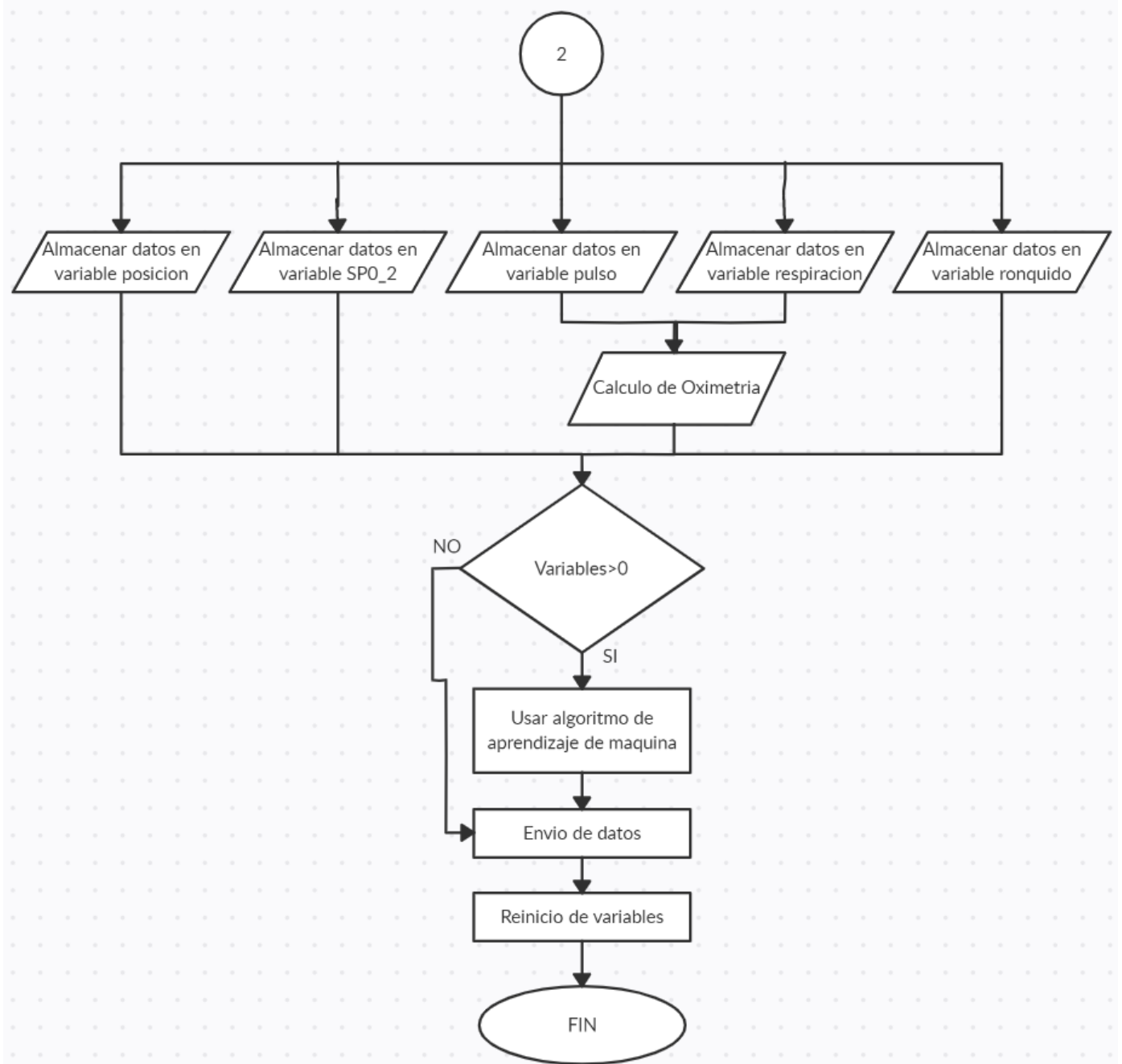


Figura 32. Diagrama de flujos del sistema

Fuente: Elaboración propia obtenido de Diagram FlowCharts

3.6. Implementación

A continuación se realiza la implementación de cada uno de los bloques del sistema inteligente mismo que consta de dos partes fundamentales tales como Hardware y software.

En este apartado se añaden los componentes físicos de los bloques establecidos necesarios para el funcionamiento del sistema, ya sean estos considerados elementos pasivos o activos.

3.6.1. Implementación de armazón protector del sistema.

Como parte principal del sistema, se considera al denominado chasis del prototipo, mismo que se encarga del alojamiento de cada componente de los respectivos bloques a implementarse. En la Figura 33, se muestra el diseño que tiene como objetivo la protección de los componentes del sistema.

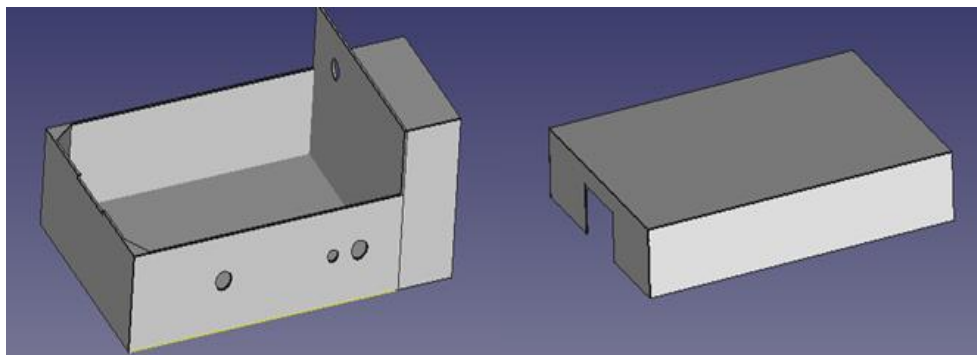


Figura 33. Diagrama de chasis de prototipo

Fuente: Elaboración propia obtenido de FreeCAD

3.6.2. Bloque de acoplamiento de la señal.

A continuación se presenta el diseño de la placa de acoplamiento de la señal implementada desde EAGLE como se muestra en la Figura 34.

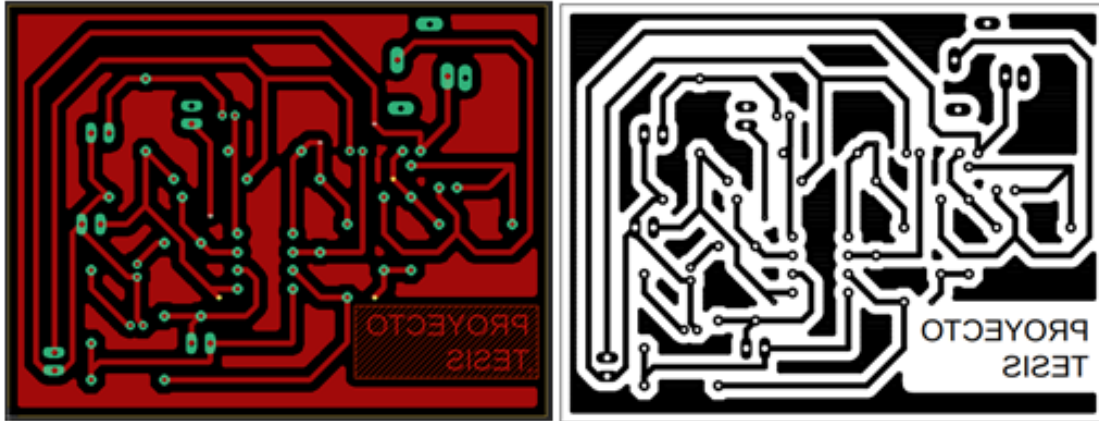


Figura 34: Diseño e impresión de amplificador en placa

Fuente: Elaboración propia obtenida de EAGLE

En la Figura 35, se puede observar la señal receptada sin ningún tipo de tratamiento ni acondicionamiento. Luego se puede apreciar la señal obtenida tras la implementación del filtro, el cual reduce el ruido existente en la señal recibida, perteneciente al sensor de posición del cuerpo.

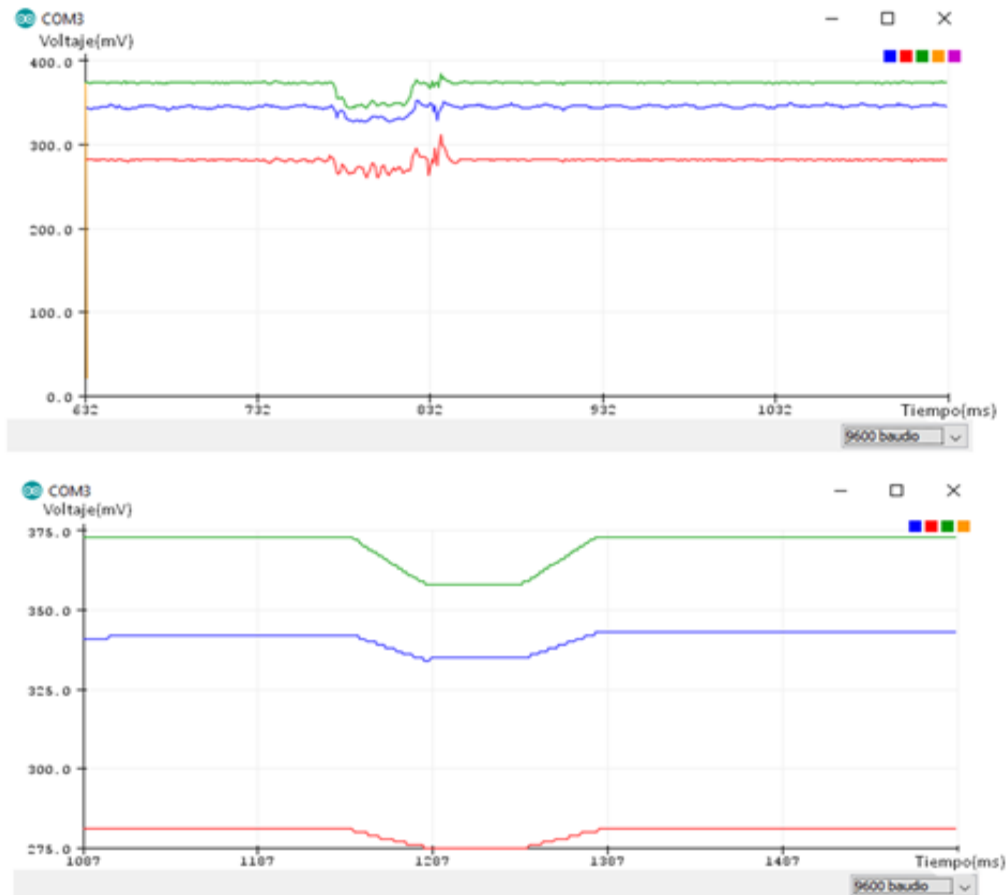


Figura 35. Señal de sensor acelerómetro

Fuente: Elaboración propia obtenida de Arduino serial plotter

3.6.3. Bloque de sensores.

Para el funcionamiento de cada uno de los sensores es necesario establecer un circuito en el cual se adhieren los mismos de manera estable, siendo esta la razón por la cual se realiza la implementación de la placa de circuito impreso.

3.6.3.1 Diseño de circuito impreso de placa electrónica Shield. Para el diseño del presente circuito se utiliza el software EAGLE en el cual se establecen medidas y los componentes que formaran parte del sistema. En la Figura 36, se observa el presente diseño.

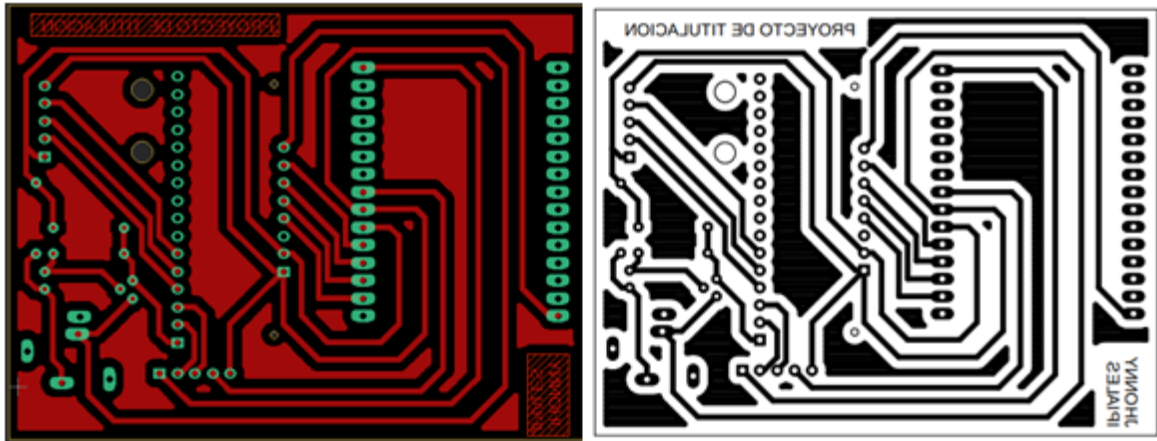


Figura 36. Diseño e impresión de placa

Fuente: Elaboración propia obtenida de EAGLE

Luego del diseño en software se procede a realizar la implementación en la placa impresa. Como se muestra a continuación en la Figura 37.



Figura 37. Implementación y acoplamiento de sensores a placa

Fuente: Elaboración propia

3.6.4. Bloque de alimentación.

Para el bloque de alimentación se realizara la conexión de los componentes antes especificados en el diagrama de conexión (Figura 21). A continuación en la Figura 38, se muestra la implementación del bloque de alimentación según las especificaciones y cálculos realizados.

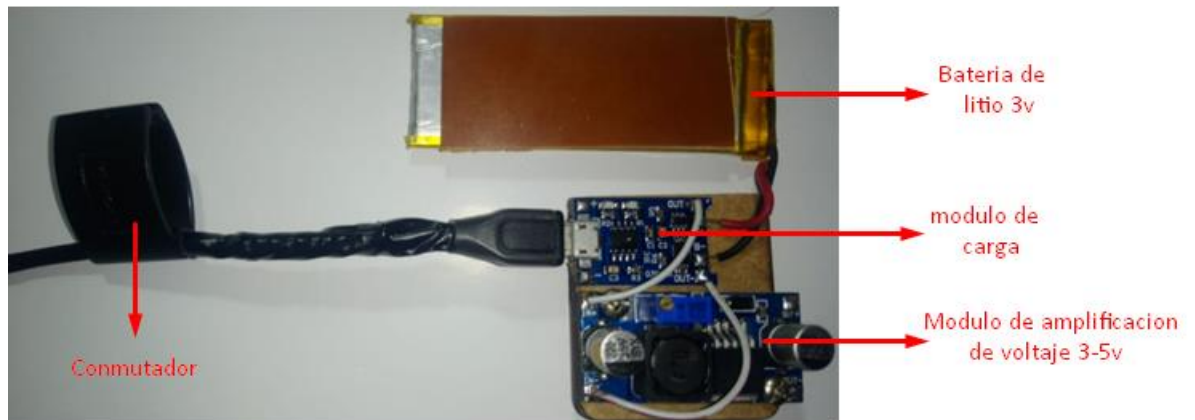


Figura 38. Implementación de Fuente de alimentación

Fuente: Propia

3.6.5. Bloque de transmisión.

Para este bloque se utiliza una de las características especiales del NodeMCU el cual es un módulo WiFi, mismo que permitirá enviar la información necesaria para la posterior representación en la plataforma “Cayenne”. En la Figura 39 se representa la implementación del código aplicado.

```

#include "UbidotsMicroESP8266.h"           //incluye libreria
#define TOKEN "A1E-IQIKLiptLPHWp5AMtES5c22LR1syPA" //incluye token
Ubidots client (TOKEN);                   //inicia token como cliente
const char* ssid      = "DESK";           //designa usuario
const char* password  = "987654321";     //designa contraseña
    client.add ("etiqueta", etiqueta);    // envia el valor de etiqueta a ubidots
    hora=0;
    j=0;
    k=0;
}
client.sendAll(true);                     // comando true si es correcto el envio

```

Figura 39. Implementación de código para módulo WiFi

Fuente: Obtenido de (Arduino, 2018)

3.6.6. Bloque de procesamiento de datos.

En este bloque se desarrolla el algoritmo con el cual se realiza la clasificación de los datos obtenidos a través de los sensores seleccionados para el proyecto. Para este caso se realizara el proceso del algoritmo KDD como se representó en la Figura 23.

3.6.6.1. Selección. En la etapa de selección se toma los datos obtenidos por cada uno de los sensores de diferentes personas. A continuación en la Figura 40, se muestra la BDD obtenida por el sistema inteligente perteneciente a valores de pulso cardiaco, SPO2, ronquido, posición del cuerpo y numero de respiraciones.

PULSO_CARD	SAO2.	posicion	respiracion	ronquido	label
68.30	97.80	1	16	288.31	1
70.67	98.50	1	16	272.43	1
71.10	98.20	1	16	288.70	1
68.10	97.40	1	16	276.96	1
68.10	96.70	1	16	282.42	1
70.00	97.40	1	16	281.75	1
68.10	97.63	1	16	280.85	1
71.10	97.60	1	16	273.63	1
71.10	97.88	1	16	271.85	1
70.72	98.00	2	16	274.26	1
71.17	97.70	2	16	272.68	1

Figura 40. BDD obtenida por el sistema

Fuente: Elaboración propia

3.6.6.2. Transformación. En esta fase es donde se realiza la reducción de la base de datos haciendo uso del software RStudio. Se realiza la reducción de información haciendo uso de diferentes algoritmos, entre ellos tenemos CNN, DROP1, DROP2, DROP3. Después de realizar un análisis se determina el algoritmo óptimo es CNN, siendo el que remueve 1058 instancias, con un porcentaje de reducción del 99.81%, tal como se muestra en la Figura 41.

```
Results:
Number of removed instances: 1058 (99.81132 %)
Number of repaired instances: 0 (0 %)
> |
```

Figura 41. Porcentaje de reducción

Fuente: Elaboración propia

Es importante tener en cuenta que un 75% de la BDD inicial es utilizada como datos de entrenamiento del sistema mientras que el 25% restante es la base de datos que se usará para ejecutar el algoritmo k-NN.

3.6.6.3. Data Mining (Minería de datos). En esta fase se realiza la elección del algoritmo que tenga mayor eficiencia comparando entre diferentes algoritmos, una vez determinado se implementa en el sistema para proceder a realizar las pruebas donde se clasificará el nivel de apnea que posee cada persona.

Para ello se realizaron 4 simulaciones con diferentes algoritmos que son k-NN, Bayesiano, SVM y Árbol de decisión, en la Tabla 17 se indica la eficiencia que obtuvo cada una de ellas:

Tabla 17. Algoritmos de aprendizaje

Semilla	Algoritmos de aprendizaje automático				
	k-NN K=1	k-NN K=3	Bayesiano	SVM	Árbol de decisión
280	93,6%	93,2%	91,6%	90,1%	90,6%
600	93,2%	92,7%	91,3%	89,9%	90,5%
990	93%	92,4%	91,1%	89,5%	90,2%
Promedio	93,2%	92,7%	91,3%	89,8%	90,4%

Fuente: Elaboración propia adaptado de (RStudio, 2019)

Luego de la verificación de cada uno de los algoritmos, se concluye que el porcentaje promedio más alto pertenece al algoritmo k-NN con K=1 con un 95,2% de efectividad. Por lo cual es seleccionado para la implementación en el presente sistema inteligente.

3.6.7. Bloque de Representación de datos.

En este bloque se encarga de recibir y representaran los datos obtenidos por el sistema, para una mejor visualización e interpretación del usuario.

Para ingresar a la página web, accedemos al siguiente enlace: <https://cayenne.mydevices.com/shared/6126744b2a964b08bcde228f> en esta página se encuentran los resultados finales del monitoreo. En la Figura 42, se muestran las señales enviadas desde el sistema inteligente a nuestra interfaz de visualización de datos.

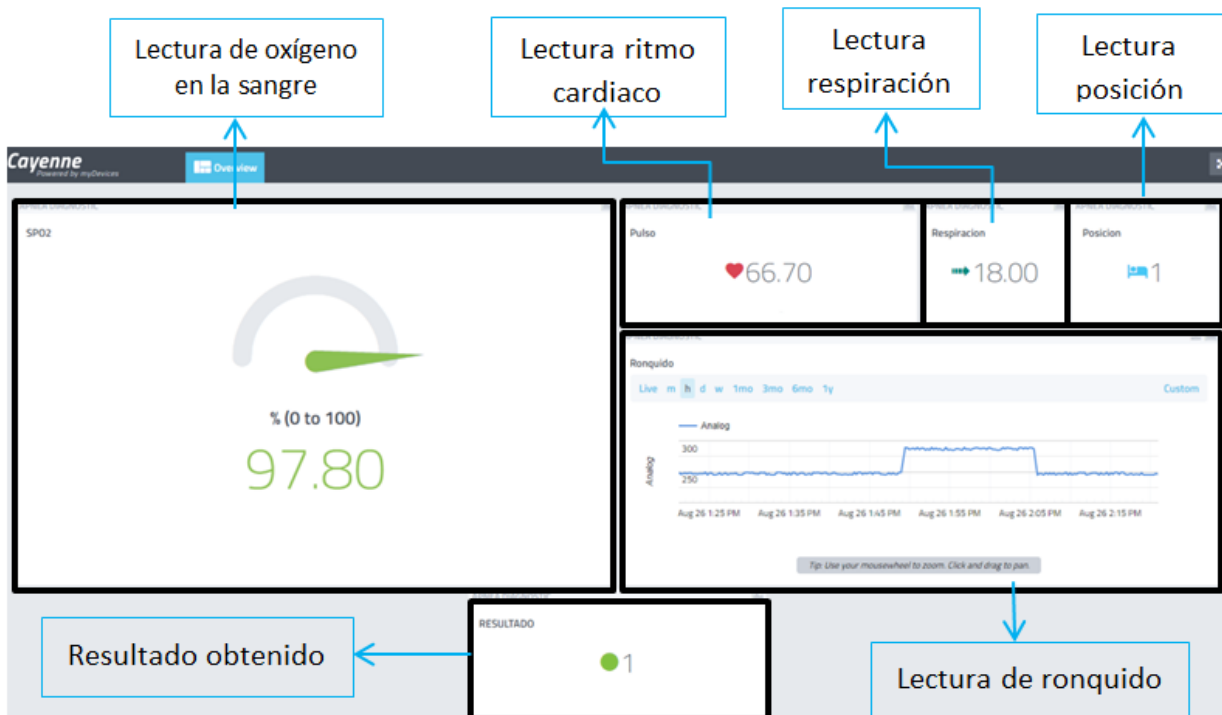


Figura 42. Interfaz de visualización de datos

Fuente: Elaboración propia obtenido de Cayenne

3.7. Implementación final de sistema

Al finalizar el diseño de las placas, se acoplan al chasis final del prototipo como se muestra en las Figuras 43 y 44, de esta manera obteniéndose el prototipo terminado y listo para las pruebas de funcionamiento.



Figura 43. Diseño de Prototipo

Fuente: Elaboración propia

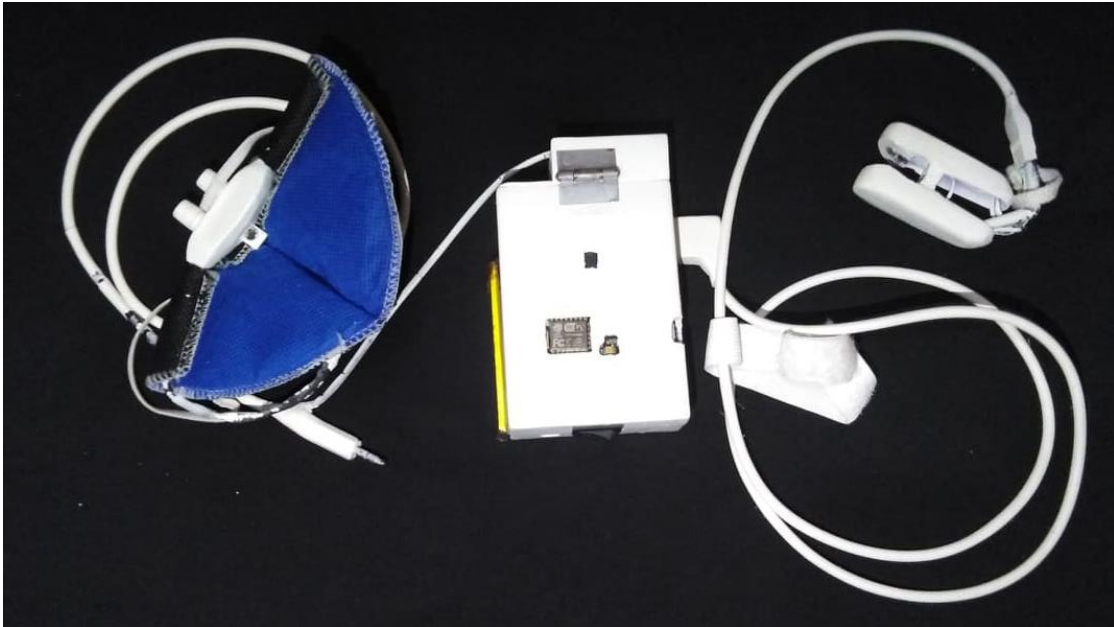


Figura 44. Diseño y presentación final

Fuente: Elaboración propia

3.8. Ubicación del prototipo.

Para la posterior recolección de datos y evaluación del sistema. A continuación se muestra la ubicación del prototipo en el entorno real como se observa en la Figura 45.

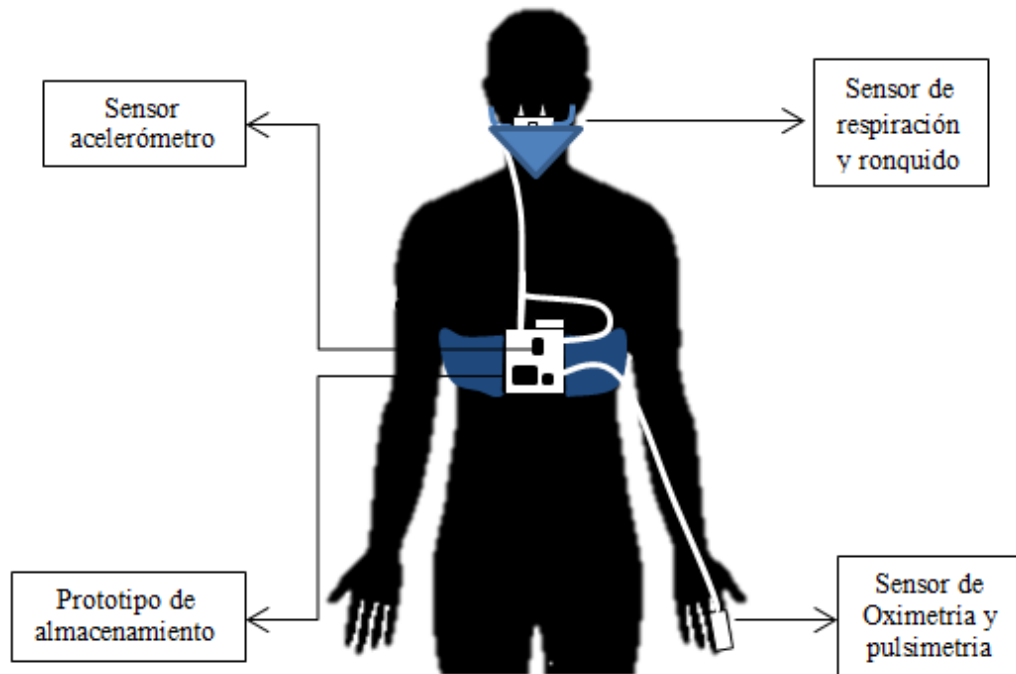


Figura 45: Ubicación de sensores en el cuerpo

Fuente: Elaboración propia

CAPÍTULO IV. PRUEBAS DE FUNCIONAMIENTO Y RESULTADOS

En el presente capítulo, se indica las pruebas de funcionamiento del software y hardware del sistema embebido inteligente de representación de datos biomédicos para el análisis de la apnea. Con las pruebas de funcionamiento y los resultados obtenidos, se finaliza la metodología de desarrollo basada en el modelo en V en la fase de Pruebas de operación y validación.

Para el caso específico del estudio de la apnea se han determinado los rangos de las variables con las que se monitorea a los pacientes. Siendo estos los niveles de referencia que tomara el sistema para determinar si el paciente posee alguna tendencia de contraer apnea. a continuación en la tabla se muestran dichos valores.

Tabla 18: Niveles de referencia de bioseñales de monitoreo

	Nivel	Spo2	Pulso	Respiración	Posición	Ronquido
Rango Asintomático		97-100	70-75	15-25	1,2,3,4	1,0
Rango Sintomático	leve	94-96	67-69	12-14	1,2,3	1,0
	moderado	92-93	65-67	10-12	1,2,3	1,0
	grave	91-92	64-65	7 a 10	1,2,3	1

El sistema debe monitorear al usuario en un lapso de tiempo específico como se muestra en la Figura 46. Además, hay que tener en cuenta que los datos deben ser tomados en horas de descanso, en un entorno totalmente normal y habitual para el usuario, es decir en condiciones naturales para el descanso nocturno.

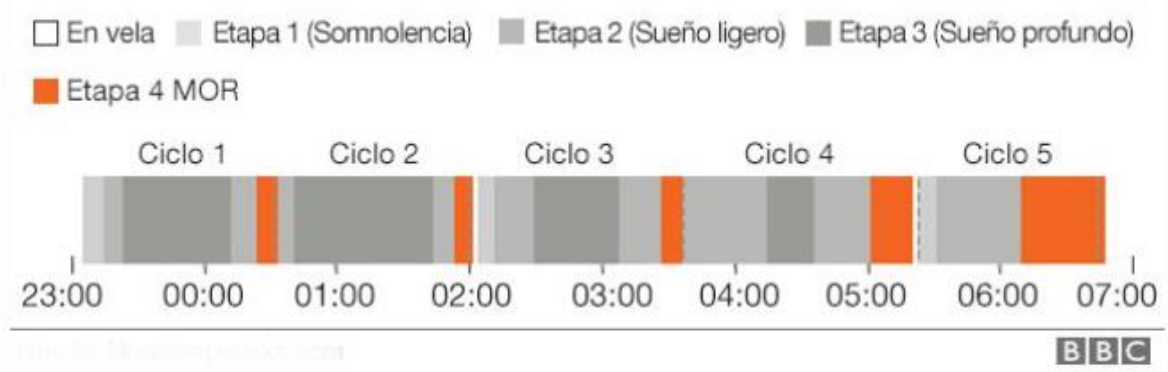


Figura 46. Etapas y ciclos del sueño

Fuente: Obtenido de (Carrillo et al., 2010)

Siendo la etapa 3 la que cumple con las características necesarias para el correcto monitoreo, se puede observar en la Figura 34 que las personas cumplen ciclos del sueño. De los cuales toman de 20 a 30 minutos iniciar la etapa del sueño profundo y aproximadamente una hora en finalizar la misma (Carrillo et al., 2010). Por esta razón el tiempo mínimo estimado de evaluación del usuario se establece en una hora (1h).

Las paradas respiratorias de un tiempo superior a 10 segundos son denominados apneas y para determinar el grupo al que el usuario pertenece, es necesario dividirlos en tres clases según el número de apneas presentes:

- **Leve:** cuando hay entre 5 y 15 apneas por hora, es decir entre 2 y 7 apneas por cada 30 minutos.
- **Moderado:** el paciente tiene entre 15 y 30 apneas por hora, es decir entre 8 y 15 apneas por cada 30 minutos.
- **Grave o severo:** cuando pasa de 30 apneas por hora, es decir mayor a 15 apneas cada 30 minutos.

4.1. Resultados

Para demostrar los resultados, se ha realizado diferentes pruebas en los bloques determinados en el capítulo anterior. Además en personas con las diferentes características del trastorno a estudiar, consiguiendo de esta manera resultados importantes con respecto al mismo.

4.1.1. Pruebas de funcionamiento del sistema

A continuación, se muestran las pruebas realizadas en el sistema, basadas en cada uno de los bloques anteriormente propuestos.

4.1.1.1. Bloque de alimentación. En este apartado se verifica el correcto funcionamiento del bloque de alimentación, mismo que también abarca el encendido de los sensores conectados al sistema. A continuación, en las Figuras 47 y 48 se observa el sistema completamente encendido, lo cual es indicador que el bloque de alimentación funciona adecuadamente.

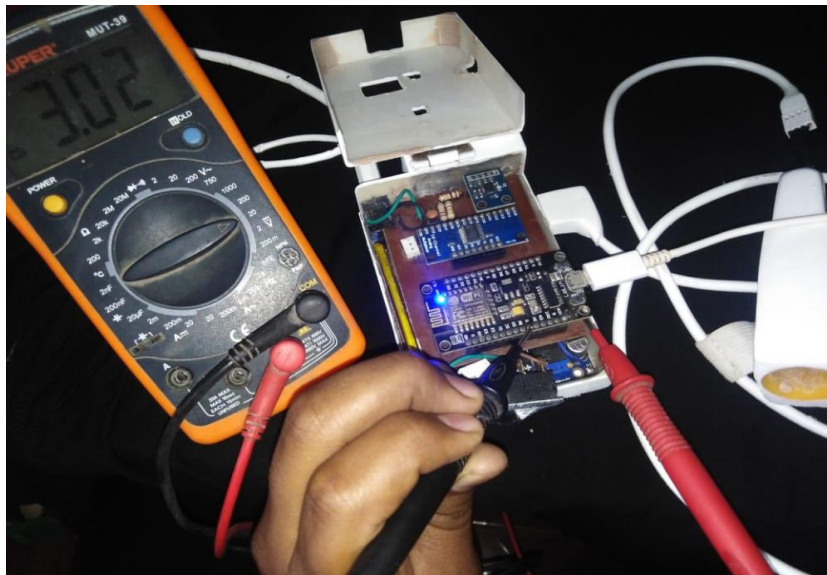


Figura 47: Medición de voltaje de funcionamiento de sistema

Fuente: Elaboración propia



Figura 48: Indicadores y/o Luces de encendido

Fuente: Elaboración Propia

4.1.1.2. Bloques de sensores, acoplamiento y procesamiento. En este apartado se demuestra el funcionamiento de tres bloques en específico debido a que resulta importante el trabajo de estos de manera conjunta. Es por ello que el resultado de la prueba realizada demuestra el correcto funcionamiento del bloque acoplamiento y procesamiento. En la Figura 49, se muestra el resultado de la prueba realizada.

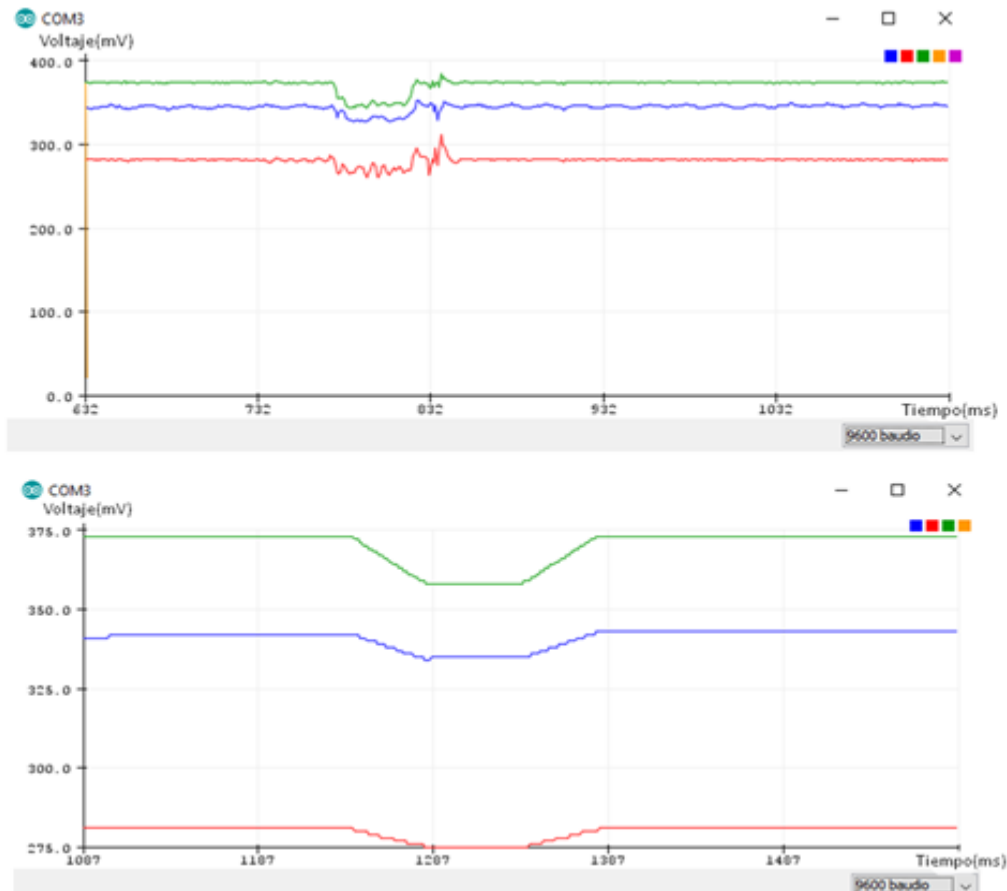


Figura 49: Funcionamiento de filtro lógico

Fuente: Obtenido de (Arduino, 2018)

Luego de observar la figura, se verifica el correcto trabajo que realiza el filtro lógico aplicado para el proceso de acoplamiento de la señal en el sistema. A continuación, en la Figura 50 se muestra los datos obtenidos por los respectivos bloques establecidos.

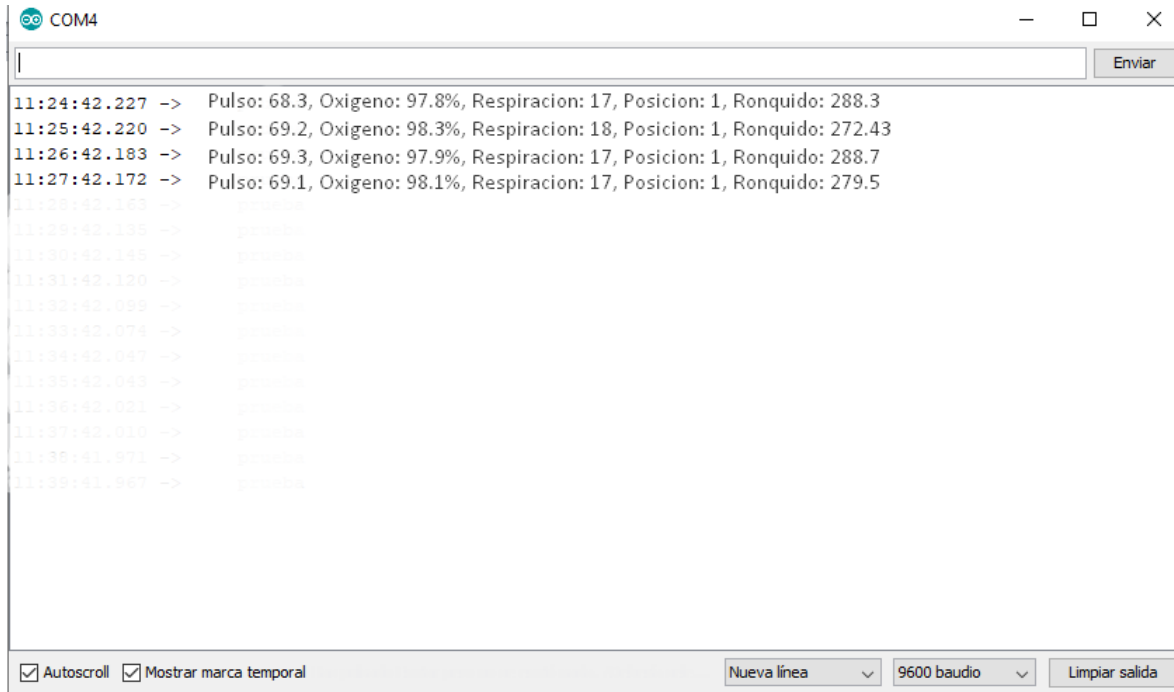


Figura 50: Evaluación de monitoreo de variables

Fuente: Elaboración propia obtenido de (Arduino, 2018)

Estos datos obtenidos son almacenados y procesados para posteriormente enviarlos a la plataforma de representación y así brindar un resultado final al usuario.

4.1.1.3. Bloque de envío y representación. Luego de las pruebas realizadas anteriormente, es importante demostrar que los datos obtenidos y procesados en el sistema, sean enviados hacia la plataforma que se encarga de recibir y representar la información. La Figura 51 muestra el mensaje que confirma el envío de las variables obtenidas.

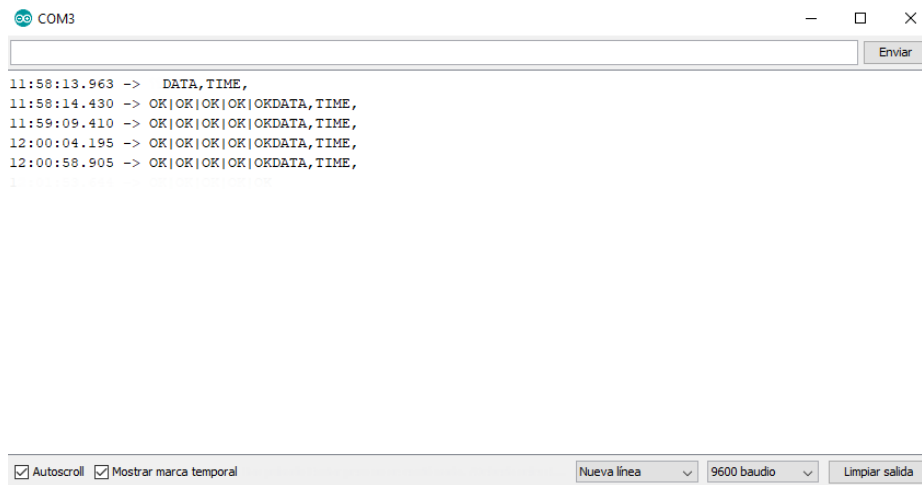


Figura 51: Confirmación de envío a plataforma

Fuente: Obtenido de (Arduino, 2018)

Luego de realizarse el correcto envío de los datos obtenidos a través del sistema, a continuación se procede a verificar la recepción de los mismos en la plataforma de recepción. En la Figura 52 se muestran los datos recibidos.

Marca de tiempo	Nombr...	Canal	Nombre del sensor	ID del sensor	Valores
2021-08-26 1:15:39	DIAGNÓSTI...	4	Posicion	7ae78d80-05ce-11ec-90...	1
2021-08-26 1:15:39	DIAGNÓSTI...	1	Pulso	ac1b2490-05cc-11ec-be0...	67.900001525879
2021-08-26 1:15:24	DIAGNÓSTI...	3	Ronquido	d412a8f0-05cd-11ec-be0...	247
2021-08-26 1:15:24	DIAGNÓSTI...	4	Posicion	7ae78d80-05ce-11ec-90...	1
2021-08-26 1:15:24	DIAGNÓSTI...	0	SPO2	1dc11370-05cd-11ec-ab9...	99.699996948242
2021-08-26 1:14:53	DIAGNÓSTI...	5	RESULTADO	be4ce360-05cc-11ec-927...	1
2021-08-26 1:14:53	DIAGNÓSTI...	4	Posicion	7ae78d80-05ce-11ec-90...	1
2021-08-26 1:14:53	DIAGNÓSTI...	0	SPO2	1dc11370-05cd-11ec-ab9...	99.699996948242
2021-08-26 1:14:53	DIAGNÓSTI...	1	Pulso	ac1b2490-05cc-11ec-be0...	67.900001525879
2021-08-26 1:14:53	DIAGNÓSTI...	2	Respiracion	859c4b90-05cd-11ec-be...	19
2021-08-26 1:14:53	DIAGNÓSTI...	3	Ronquido	d412a8f0-05cd-11ec-be0...	245

Figura 52: Datos recibidos desde el sistema de monitoreo

Fuente: Obtenido de Cayenne.mydevices.com

4.1.1.4. Pruebas de funcionamiento. Para verificación de funcionamiento se realiza un check list, en el cual consta de requerimientos que el sistema cumple correctamente. De esta manera asegurando el funcionamiento de manera óptima. A continuación se muestra en la Tabla 19 la lista de requerimiento de funcionamiento.

Tabla 19: Check list de Pruebas de funcionamiento



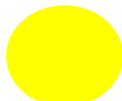

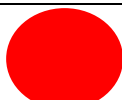
CHECK-LIST PRUEBAS DE FUNCIONAMIENTO			
Parámetro	Cumplimiento		Observación
	SI	NO	
Valor de alimentación al núcleo del sistema es correcto	X		Valores pueden variar entre 0 a 0,03 Volts
Alimentación al multiplexor de acoplamiento	X		Valores pueden variar entre 0 a 0,03 Volts
El bloque de acondicionamiento funciona	X		La señal brindada es correcta
Funcionamiento de cada uno de los sensores	X		Función de recolección de datos optima
El sistema de recepción capta datos correctos	X		Funciona adecuadamente
Módulo de envío realiza su función correctamente.	X		Valores correctamente enviados
Funciona el filtro lógico aplicado al sistema	X		Cumple su función correctamente
Algoritmo de clasificación funciona correctamente.	X		Algoritmo funciona adecuadamente
Código de envío realiza su función	X		Envío se realiza correctamente

Fuente: Elaboración propia

4.1.2. Pruebas generales en usuarios

A continuación en la Tabla 20, se muestra la simbología utilizada para la clasificación de los resultados brindados por el sistema. Dicha simbología cambiara dependiendo de la gravedad del usuario que se realice el respectivo monitoreo.

Tabla 20. Simbología de resultados

SIMBOLOGIA	
	Error de lectura
	Asintomático
	Síntomas (Leve)
	Síntomas (Moderado)
	Síntomas (Grave)

Fuente: Elaboración propia

Luego de establecer los correspondientes niveles de sintomatología y le tiempo óptimo para la presente fase de pruebas, a continuación en la Tabla 21 se procede a enlistar las características de los pacientes que se someterán al monitoreo a través de sistema.

Tabla 21. Características de usuarios sometidos a monitoreo

	SPO2 (%)	PULSO (Bpm)	RESPIRACION POR MINUTO	RONQUIDO	RESULTADO
Usuario 1	98,7	69	18	NO EXISTE	ASINTOMATICO
Usuario 2	99,7	68	19	NO EXISTE	ASINTOMATICO
Usuario 3	97,8	66	18	NO EXISTE	ASINTOMATICO
Usuario 4	94,6	68	13	SI EXISTE	SINTOMA LEVE
Usuario 5	95,4	69	14	SI EXISTE	SINTOMA LEVE
Usuario 6	93,4	66	11	SI EXISTE	SINTOMA MODERADO

Fuente: Elaboración propia

A continuación, se procede a realizar las respectivas pruebas en cada uno de los usuarios antes indicados con el fin de determinar la sintomatología que corresponde a cada uno de los mismos.

Cada uno de los presentes pacientes fueron monitoreados y analizados durante una semana (7 días) con el objetivo de revisar posibles cambios que pudiesen afectar el resultado final del monitoreo.

4.1.2.1. Prueba con características biomédicas de usuario 1.

Se realizó las pruebas durante un periodo de tiempo de una hora (1h), en el cual se obtienen los siguientes valores que se muestran en la Figura 53.

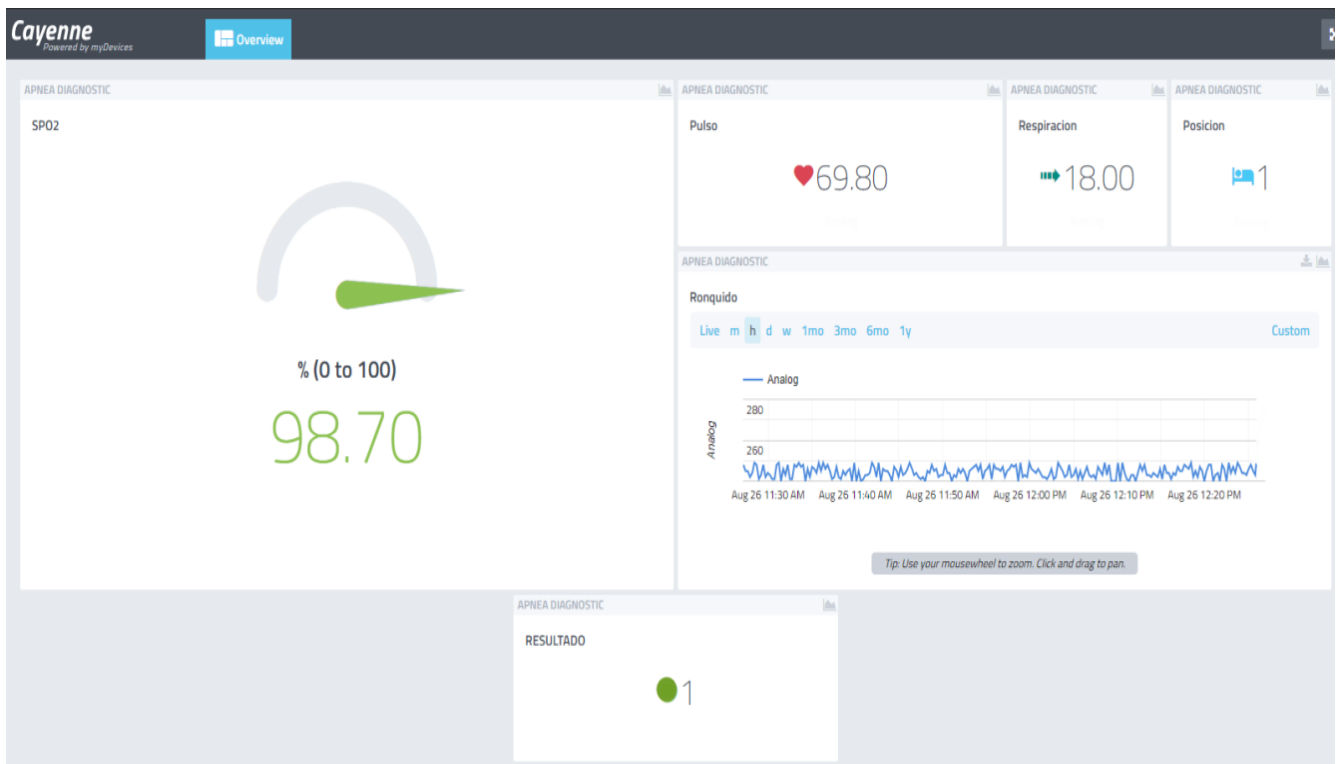


Figura 53. Datos obtenidos de usuario 1

Fuente: Elaboración propia obtenido de Cayenne

La Figura 53 muestra los datos obtenidos por el sistema, en el cual se puede observar las variables en un nivel normal en la etapa del sueño, el led encendido de color verde indica que, debido a los datos obtenidos el paciente no posee características de presencia del trastorno del sueño.

4.1.2.2. Prueba con características biomédicas de usuario 2.

En la Figura 54, se muestra los datos obtenidos por el sistema del paciente (usuario 2) con características habituales de la etapa del sueño.

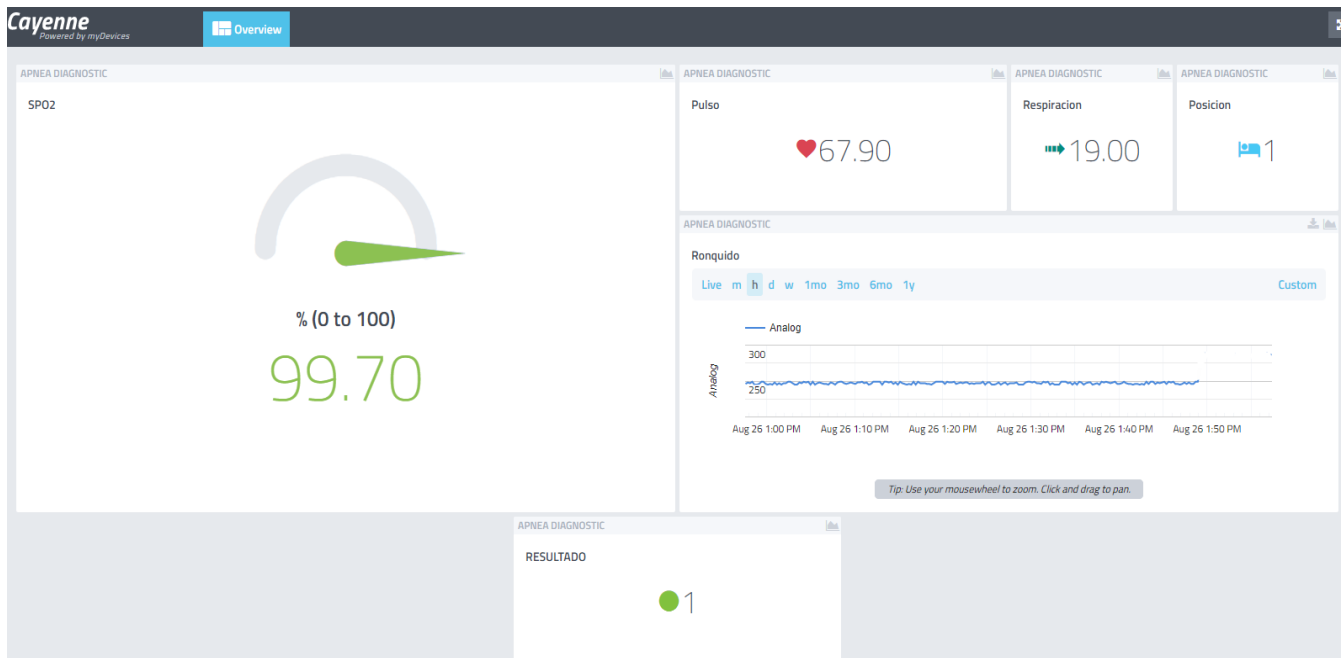


Figura 54: Datos obtenidos del usuario 2

Fuente: Elaboración propia obtenido de Cayenne

De la misma manera que el anterior caso, se pueden observar ciertas variaciones en comparación con el otro paciente. Sin embargo, la clasificación realizada por el sistema nos da como resultado que el usuario no posee características con tendencia a la presencia de trastorno de apneas del sueño.

4.1.2.3. Prueba con características biomédicas de usuario 3.

Los datos obtenidos por el sistema del paciente (usuario 3) se representan a continuación en la Figura 55.

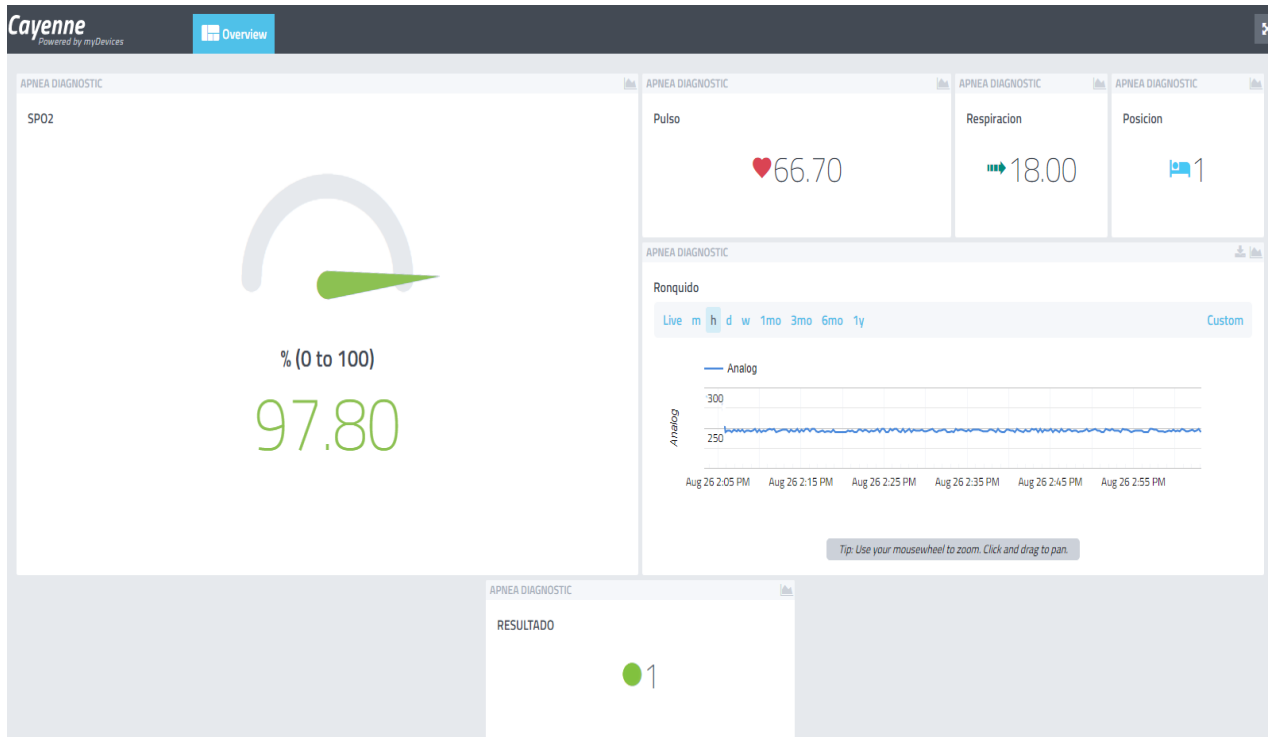


Figura 55. Datos obtenidos del usuario 3

Fuente: Elaboración propia obtenido de Cayenne

En la Figura 55, se observa las características biomédicas de un usuario clasificado como asintomático por el sistema, ya que todos sus valores son lo más cercanos a lo óptimo. En este caso en específico, se observa el ritmo cardiaco más bajo en comparación con los otros casos. Sin embargo, los otros datos indican que el estado del paciente está en un rango normal.

4.1.2.4. Prueba con características biomédicas de usuario 4.

En la Figura 56, se observa los datos obtenidos por el sistema dichos datos, poseen ciertas variaciones en comparación de los casos expuestos anteriormente.

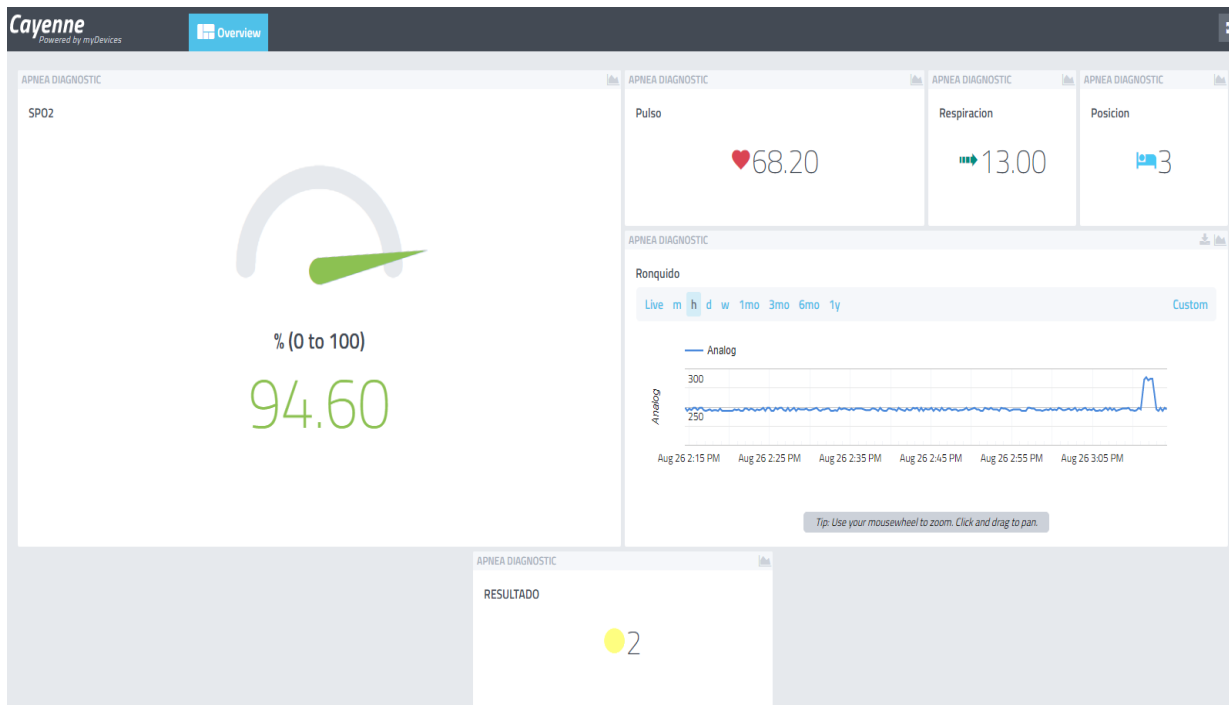


Figura 56. Datos obtenidos del usuario 4

Fuente: Elaboración propia obtenido de Cayenne

En la imagen se puede observar el caso de un paciente con problemas de ronquido y un número bajo de respiraciones por minuto, por lo cual indica que el usuario posee ciertas características de trastorno de apnea. Por este motivo el sistema ha clasificado y ha indicado esta anomalía a través del led de color amarillo.

4.1.2.5. Prueba con características biomédicas de usuario 5.

A continuación, se muestra la verificación de datos obtenidos por el sistema. En la Figura 57, se puede observar la representación de los datos de usuario 5.

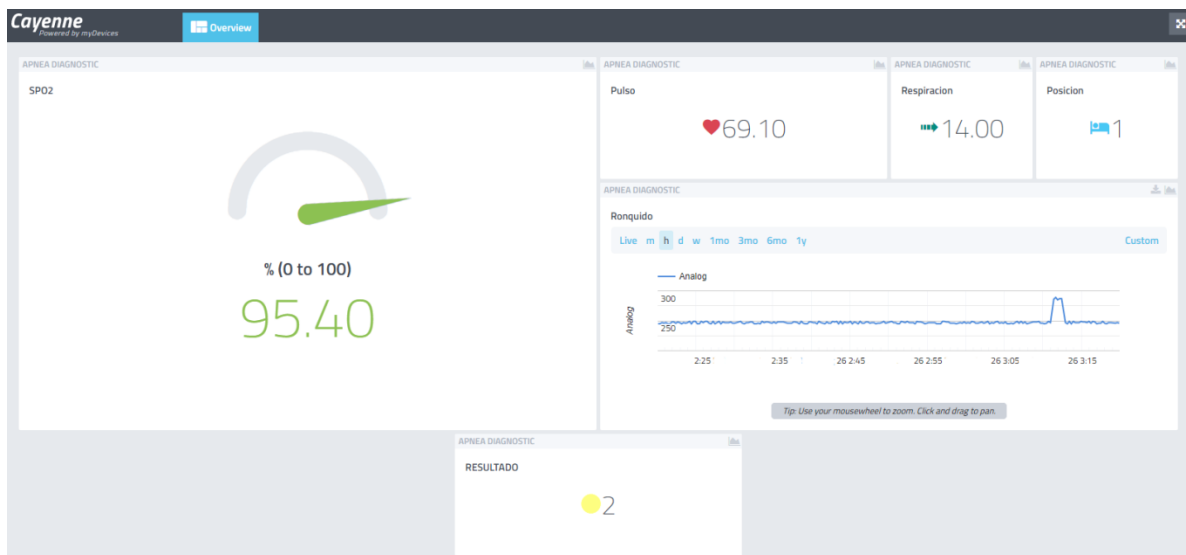


Figura 57. Datos obtenidos de usuario 5

Fuente: Elaboración propia obtenido de Cayenne

Al igual que en el caso anterior se observa variaciones en las variables obtenidas del usuario, por ello el sistema a clasificado al paciente con tendencia de trastorno de apneas en el sueño. En el presente caso se observa un buen ritmo cardiaco. Sin embargo, las demás variables presentan variaciones lo cual permite al sistema tomar una diferente decisión.

4.1.2.6. Prueba con características biomédicas de usuario 6.

La prueba final es la más importante debido a que esta prueba indica el correcto funcionamiento del sistema. En la Figura 58 se muestra los valores obtenidos.

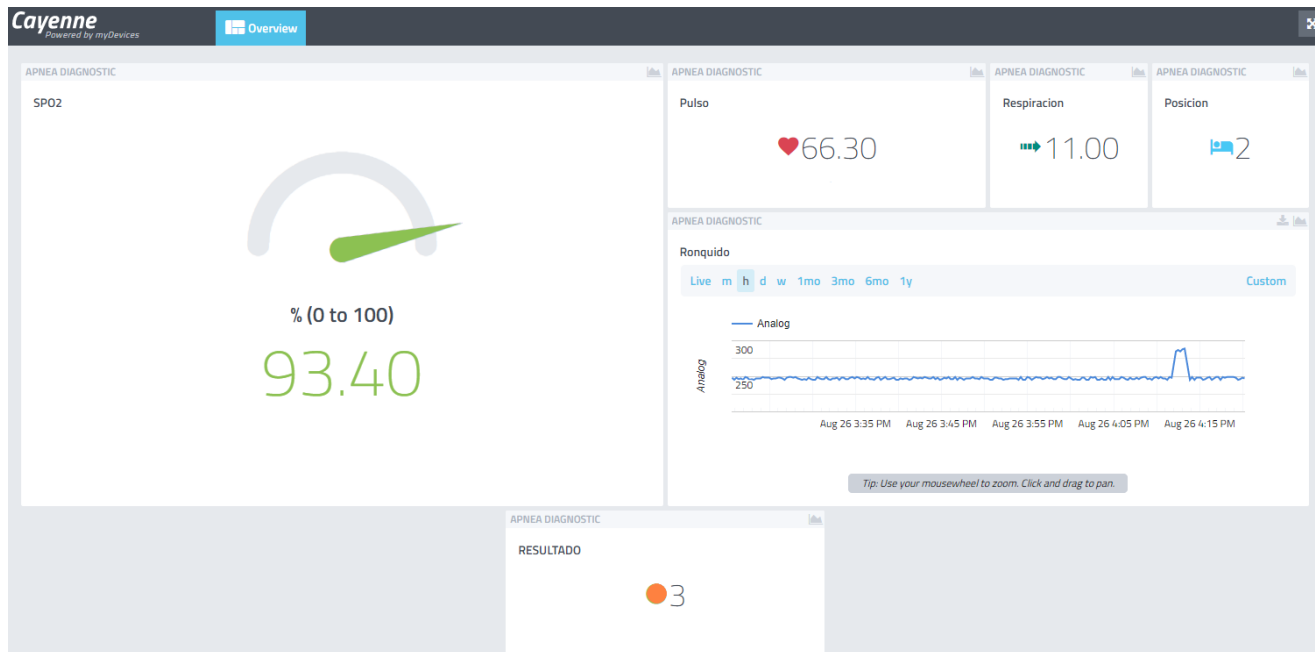


Figura 58: Datos biomédicos del usuario 6

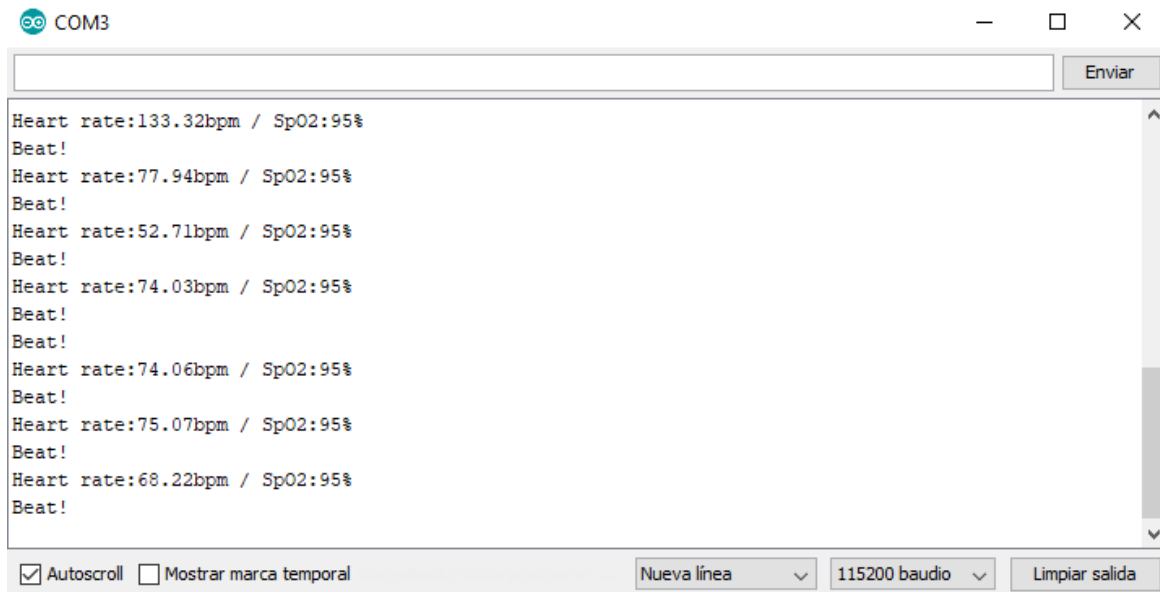
Fuente: Elaboración propia obtenido de Cayenne

Los valores obtenidos a través del sistema se consideran óptimos debido a que el usuario presenta trastorno de apneas del sueño como el diagnostico medico así lo dictamina. Como se puede observar las variaciones obtenidas en comparación a los demás pacientes sometidos al monitoreo del sistema, son notablemente diferentes. Es por ello que el sistema en función a las variables evaluadas presenta un resultado diferente.

4.2 Prueba validación funcional de sensor Oxímetro

El sensor diseñado ha sido puesto a prueba para asegurar su validación funcional, para ello se realiza la comparación con un sensor comercial, en el cual se verifica la saturación de oxígeno en la sangre y la frecuencia cardiaca.

A continuación en la Figura 59, se muestran algunos de estos valores obtenidos en pruebas realizadas en dos individuos, el valor de pulso cardíaco son pulsaciones por minuto y el de la saturación de oxígeno es en porcentaje.



The image shows a screenshot of a serial monitor window titled "COM3". The window contains a list of data points for heart rate and SpO2 saturation. The data points are as follows:

Heart rate (bpm)	SpO2 (%)
133.32	95%
77.94	95%
52.71	95%
74.03	95%
74.06	95%
75.07	95%
68.22	95%

The window also includes a "Enviar" button at the top right, a scroll bar on the right, and control options at the bottom: "Autoscroll" (checked), "Mostrar marca temporal" (unchecked), "Nueva línea" (dropdown), "115200 baudio" (dropdown), and "Limpiar salida" (button).

Figura 59. Valores de saturación de oxígeno y frecuencia cardiaca

Fuente: Elaboración propia obtenido de Arduino IDE

Se comprueba que los valores de saturación de oxígeno se encuentran dentro de los valores normales.

Si se compara los valores obtenidos con el prototipo diseñado en ambas pruebas con los valores que se obtienen del dispositivo comercial, observamos que los valores obtenidos son muy similares como a continuación se muestra en la Tabla 22.

Tabla 22. Comparativa entre diseño comercial y prototipo

Individuo 1				Individuo 2			
Dispositivo comercial		Prototipo diseñado		Dispositivo comercial		Prototipo diseñado	
Pulso	SaO2	Pulso	SaO2	Pulso	SaO2	Pulso	SaO2
75	98	76,43	97,73	79	97	78,01	96,68
74	98	74,43	97,43	78	97	77,14	96,40
76	98	76,43	96,65	79	98	78,72	97,90
73	98	73,59	97,62	78	97	78,42	96,41

Fuente: Elaboración propia

4.3. Matriz de confusión.

Para realizar el análisis de la matriz de confusión debemos evaluar los falsos positivos, falsos negativos, verdaderos positivos y verdaderos negativos como se observa en la Tabla 23.

Tabla 23. Interpretación de matriz de confusión.

TN	FP	TN
FN	TP	FN
TN	FP	TN

Fuente: (Rouse, 2016)

Se implementa la matriz de confusión basado en los datos obtenidos en RStudio, como se presenta a continuación en la Tabla 24.

Tabla 24. Matriz de confusión

	SINTOMATICO	ASINTOMATICO
SINTOMATICO	168	11
ASINTOMATICO	7	165

Fuente: Elaboración propia

A continuación se calcula el porcentaje de exactitud a través de la ecuación 4:

$$\%Exactitud = \frac{TN + TP + TN}{\#TOTAL DE DATOS} \times 100$$

Ecuación 4. Ecuación de porcentaje de exactitud

Fuente: (Alibaba, 2018)

$$\%Exactitud = \frac{168 + 165}{351} \times 100$$

$$\%Exactitud=94,87\%$$

Mediante el cálculo del porcentaje de exactitud del sistema se puede deducir que se aproxima al 93,34%, de esta manera dando un 6,7% de error del mismo.

4.5. Valores máximo y mínimos obtenidos de la toma de datos

Durante el proceso de recopilación y comprobación de datos, se localizan ciertos datos que pudiesen ser los límites tanto inferior como superior del sistema, es por esto que se ve la necesidad de analizarlos en la siguiente tabla que se muestra a continuación.

En la Tabla 25 se pueden observar los valores máximos y mínimos establecidos por el sistema.

Tabla 25. Valores máximos y mínimos del sistema

	Mínimo(Min)	Máximo(Max)
Oxígeno en la sangre(SPO2)	88%	100%
Pulso Cardíaco(Bpm)	64Bpm	85Bpm
Respiración por minuto	10	24
Ronquido	0	1
Posición del cuerpo	1	3

Fuente: Elaboración Propia

4.6 Test de pruebas basadas en el modelo en V

En este apartado se describen en detalle cada uno de los casos de pruebas que se han identificado como necesarios para comprobar la funcionalidad del sistema, se identifica las pruebas que deben realizarse para asegurar el correcto funcionamiento del prototipo

<i>4.6.1. Test Eléctrico</i>	Subsistema prueba	Eléctrico
	Prueba de despliegue	Si/No
Descripción:		
Prueba de diseño del prototipo basado en el subsistema eléctrico.		
Prerrequisitos:		
<ol style="list-style-type: none"> 1. Conexiones de Shield y distribución hacia y desde las placas programables a los sensores 2. Verificar ubicación de dispositivos 		
Pasos:		
<ul style="list-style-type: none"> • Verificación de encendido de placa programable NodeMCU • Verificación de encendido de dispositivos electrónicos: sensores y módulos de comunicación. • Verificación de niveles de voltaje. • Comprobación de suministro de energía continua hacia dispositivos electrónicos. 		
Resultado esperado:		
Tener todos los elementos necesarios para el suministro y distribución de energía en el sistema embebido, con un consumo de energía ideal, para evitar que los sensores no tengan la suficiente energía para funcionar y evitar que el sistema tenga fallas durante la realización de la toma de datos.		

Resultado obtenido:

Se verificó que tanto la placa de procesamiento NodeMCU y la placa de red de sensores funciona correctamente, los niveles de voltaje son adecuados para su correcto funcionamiento.

<i>4.6.2. Test Subsistema pasivo</i>	Subsistema prueba	Eléctrico
	Prueba de despliegue	Si/No
<p>Descripción:</p> <p>Prueba de diseño del prototipo basado en el subsistema pasivo.</p>		
<p>Prerrequisitos:</p> <ol style="list-style-type: none"> 1. Adquisición de cables conectores para el prototipo móvil 2. Ubicación adecuada en el usuario. 3. Seguridad ante los movimientos durante el sueño. 		
<p>Pasos:</p> <ul style="list-style-type: none"> • Comprobación de estado de cables para el prototipo • Verificación de conexiones a pines de placa programable • Posición del sistema embebido en el usuario. • Comprobación de estabilidad del sistema embebido en el usuario. 		
<p>Resultado esperado:</p> <p>Las conexiones con todos los elementos electrónicos desde la placa NodeMCU debe estar funcional y no sufrir alteraciones o daños con el movimiento del usuario durante la toma de datos, permitiendo estabilidad a los sensores.</p>		

Resultado obtenido:

Las conexiones realizadas en la placa se encuentran en estado correcto y seguro, las soldaduras realizadas están en buen estado; también se verificó que la placa central NodeMCU está estable y

no se producirá daños en los pines. La caja en la que se encuentra el sistema embebido se sujeta adecuadamente al usuario sin riesgo de que se caiga o se averíe.

4.6.3. Test Subsistema activo	Subsistema prueba	Eléctrico
	Prueba de despliegue	Si/No
<p>Descripción:</p> <p>Prueba de diseño del prototipo basado en el subsistema activo.</p>		
<p>Prerrequisitos:</p> <ol style="list-style-type: none"> 1. Adquisición de módulos WiFi. 2. Revisión de conectividad 		
<p>Pasos:</p> <ul style="list-style-type: none"> • Comprobación de conexión de módulo WiFi con la web. • Verificación de comunicación de sistema con plataforma de representación. • Comprobación de envío y recepción de datos. 		
<p>Resultado esperado:</p> <p>Mantener una adquisición de datos con un margen mínimo de errores, almacenando los datos censados a todo momento y de todos los sensores establecidos en el sistema.</p>		

Resultado obtenido:

Se comprobó que el NodeMCU se acopla correctamente con todos los bloques, y se pudo establecer la comunicación WiFi, permitiendo el envío de los datos procesados por el sistema.

<i>4.6.4. Test Subsistema Hardware</i>	Subsistema prueba	Eléctrico
	Prueba de despliegue	Si/No
Descripción:		
Prueba de los dispositivos electrónicos utilizados para el desarrollo del prototipo		
Prerrequisitos:		
<ol style="list-style-type: none"> 1. Adquisición de materiales 2. Estudio de funcionamiento de cada dispositivo 		
Pasos:		
<ul style="list-style-type: none"> • Verificación de encendido de los dispositivos electrónicos • Verificación de encendido y funcionamiento de sensores • Revisión del funcionamiento de la placa NodeMCU 		
Resultado esperado:		
Los dispositivos electrónicos funcionan correctamente, las placas programables alimentan con 5v y 3v a los módulos y sensores. El módulo WiFi debe ser capaz de establecer correctamente conectividad para enviar los datos.		

Resultados obtenidos:

Una vez realizada el análisis del hardware se pudo determinar que los equipos están en correcto estado, el NodeMCU tiene los pines necesarios para conectar todos los sensores y módulos que se necesitan. Lo primero que se realizó es el encendido de las placas Shield para verificar que el voltaje sea suficiente para el correcto funcionamiento de los sensores establecidos.

<i>4.6.5. Test Subsistema Software</i>	Subsistema prueba	Eléctrico
	Prueba de despliegue	Si/No
<p>Descripción:</p> <p>Prueba de funciones y características del software para el uso del sistema.</p>		
<p>Prerrequisitos:</p> <ol style="list-style-type: none"> 1. Verificación de funcionamiento de filtros lógicos 2. Verificación de funcionamiento del algoritmo de clasificación. 3. Verificación de código y librerías de módulo WiFi. 		
<p>Pasos:</p> <ul style="list-style-type: none"> • Transmisión y recepción de datos. • Página web para la muestra de información. 		
<p>Resultado esperado:</p> <p>Se espera que el sistema embebido tome datos biomédicos de la anatomía del usuario, se envíe la información y finalmente se representan los datos en la plataforma.</p>		

CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones

- Durante el estudio para la realización del proyecto, se pudo obtener información necesaria y determinar que dicho trastorno es un problema difícil de detectar y estudiar a corto plazo. Por lo cual se ve la necesidad del desarrollo de un sistema de medición y análisis de la apnea a través de la visualización de bioseñales monitoreadas en un tiempo relativamente corto.
- Con la implementación del presente sistema embebido se facilita el estudio y análisis del trastorno debido a que el tiempo de monitoreo es mucho más corto y más real, ya que el estudio se realiza en un ambiente totalmente natural, sin provocar alteraciones que pudiesen afectar el resultado final.
- Dentro del grupo de algoritmos seleccionados, se determina que k-NN es uno de los algoritmos más eficientes que se adapta al sistema ya que el número de variables que se monitorean permiten una mayor exactitud en la clasificación establecida por distancia más corta entre dos grupos de prueba.
- Durante el proceso de diseño y pruebas del sistema se observan diferentes variaciones de las bioseñales monitoreadas en cada uno de los sujetos de prueba, demostrando con esto que el sistema no se basa en una sola variable sino en todas las que el prototipo le brinda.

5.2. Recomendaciones

- Es importante evaluar cada valor obtenido por los diferentes sensores, ya que cada uno monitorea un sector diferente del cuerpo. Es por esto que el proyecto está abierto a mejoramientos, tanto a nivel de software como también lo relacionado a hardware; es decir, aumento o reemplazo de sensores para tener un mayor porcentaje de efectividad en cuanto a la predicción brindada por el sistema.
- Se debe tener en cuenta la correcta ubicación de cada uno de los sensores antes de realizar el monitoreo pertinente, evitando de esta manera obtener datos erróneos por la mala ubicación de los mismos y por ende un mal resultado final.
- El proyecto permite la integración de sistemas informáticos, dando lugar a futuras investigaciones referentes al tema tratado e incluso al monitoreo simple tales como el ritmo cardiaco y porcentaje oxígeno en la sangre entre los más importantes.
- Es necesario tomar en cuenta el tiempo mínimo establecido de monitoreo para obtener un resultado más acertado, ya que caso contrario los resultados brindados por el sistema no garantizan un pronóstico altamente confiable.

BIBLIOGRAFIA

- Aguayo P, A., & Lagos T., A. P. (2012). *Guia Técnica de control de signos vitales*. Retrieved from [http://academico.upv.cl/doctos/KINE-4068/%7B328B1B37-2C2A-4747-8B38-169806A27753%7D/2012/S1/GUIA TECNICA DE CONTROL DE SIGNOS VITALES KINE.pdf](http://academico.upv.cl/doctos/KINE-4068/%7B328B1B37-2C2A-4747-8B38-169806A27753%7D/2012/S1/GUIA%20TECNICA%20DE%20CONTROL%20DE%20SIGNOS%20VITALES%20KINE.pdf)
- Alberto, I., & Farfán, C. (2010). Clasificación de los Sensores Remotos, 1–9. Retrieved from <http://thelastlabproject.blogspot.com/2010/12/clasificacion-de-los-sensores.html>
- Alfaro, R. (2012). Biopotenciales - Biomedica e instrumentación. Retrieved April 4, 2019, from <https://sites.google.com/site/biomedicaeinstrumentacion/biopotenciales>
- Ángel M. Rayo. (2016). Análisis de Datos en Big Data: tipos y fases del análisis - Bit. Retrieved April 3, 2019, from <https://www.bit.es/knowledge-center/analisis-de-datos-en-big-data/>
- Arduino Company. (2015). *Arduino UNO*. Retrieved from <https://www.farnell.com/datasheets/1682209.pdf>
- Carrillo, J., Arredondo, F., Reyes, M., Castorena, A., Vázquez, J., & Torre-Bouscoulet, L. (2010). *Síndrome de apnea obstructiva del sueño en población adulta. Clínica de Trastornos Respiratorios del Dormir, INER; Subdirección de Enseñanza, INER; Departamento de Fisiología Respiratoria, INER* (Vol. 69). Retrieved from www.medigraphic.org.mx
- Chio Cho, N., Tibaduiza Burgos, D. A., Aparicio Zafra, L. C., & Caro Ortiz, L. M. . (2013). *Redes De Sensores Inalámbricas*. Retrieved from <http://bibing.us.es/proyectos/abreproy/11750/fichero/Carpeta+3%252FCapitulo3.pdf>
- D.A,Handbook. (2012). *A Reference For DAQ And Analog & Digital Signal Conditioning. Measurement Computing Corporation (2004-2012)*. Retrieved from [https://utneduec.sharepoint.com/sites/TESSISTEMASINTELIGENTES/Documentos compartidos/LIBROS/Data-Acquisition-Handbook.pdf](https://utneduec.sharepoint.com/sites/TESSISTEMASINTELIGENTES/Documentos%20compartidos/LIBROS/Data-Acquisition-Handbook.pdf)
- Dibujes, A. (2018). *SISTEMA ELECTRÓNICO DE EVALUACIÓN DE POSTURA DE LA*

ESPALDA AL REALIZAR EL EJERCICIO DE SENTADILLAS CON BARRA LIBRE USANDO ALGORITMOS DE APRENDIZAJE AUTÓNOMO EN MIEMBROS DEL CLUB DE FÍSICO CULTURISMO DE LA UNIVERSIDAD TÉCNICA DEL NORTE. Retrieved from [http://repositorio.utn.edu.ec/bitstream/123456789/8275/1/04 RED 194 TRABAJO DE GRADO.pdf](http://repositorio.utn.edu.ec/bitstream/123456789/8275/1/04_RED_194_TRABAJO_DE_GRADO.pdf)

Gállego Pérez-Larraya, J., Toledo, J. B., Urrestarazu, E., & Iriarte, J. (2007). *Clasificación de los trastornos del sueño Classification of sleep disorders.* *An. Sist. Sanit. Navar* (Vol. 30). Retrieved from <http://scielo.isciii.es/pdf/asisna/v30s1/03.pdf>

Galmés, A. (2016). Seguridad Del Almacenamiento En La Nube, 91. Retrieved from http://openaccess.uoc.edu/webapps/o2/bitstream/10609/45887/1/Agalmesh_TFM_0116.pdf

Gómez Díaz, M. ., & Lara Rendón, C. (2008). *Análisis de bioseñales: Enfoque técnico de la adquisición, procesamiento y sus aplicaciones.* The Dial Press. Retrieved from https://www.researchgate.net/publication/327560974_Analisis_de_biosenales_Enfoque_tecnico_de_la_adquisicion_procesamiento_y_sus_aplicaciones

Guerrero Martínez, J. F. (2010). *INGENIERÍA BIOMÉDICA Tema 2 Bioseñales 2.1. Introducción.* Retrieved from http://ocw.uv.es/ingenieria-y-arquitectura/1-5/ib_material/IB_T2_OCW.pdf

Handsontec. (2017). *Handson Technology User Manual V1.2.* Hanson Technology. Retrieved from http://www.handsontec.com/pdf_learn/esp8266-V10.pdf

Haro, S., Zúñiga, L., Vera Rojas, L., & Villa, A. E. (2018). *mÉTODOs DE ClasIfICaCIÓN EN mINERía DE DaTOs mETEOrOlÓgICOs Methods of Classification in Mining of Meteorological Data.* Retrieved from <http://ceaa.esPOCH.edu.ec:8080/revista.perfiles/Articulos/Perfiles20Art13.pdf>

Kalaithasan, K., Radzi, N. A. M., & Abidin, H. Z. (2018). Internet of things application in monitoring sick building syndrome. *Indonesian Journal of Electrical Engineering and Computer Science*, 12(2), 505–512. <https://doi.org/10.11591/ijeecs.v12.i2.pp505-512>

Libelium Comunicaciones, D. (2017). MySignals SW eHealth and Medical IoT Development

Platform Technical Guide. <https://doi.org/10.5382/econgeo.2018.4561>

Luque, C. M. (2003). *El algoritmo Naïve Bayes Resumen*. Retrieved from https://www.nebrija.es/~cmalagon/inco/Apuntes/bayesian_learning.pdf

Maldonado, Li. (2018). PROTOTIPO DE UN SISTEMA DE SENSORES DE EVALUACIÓN DEL ÁNGULO DE INCLINACIÓN FEMOROTIBIAL EN EL EJERCICIO DE SENTADILLA CON BARRA LIBRE TRADICIONAL APLICADO A DEPORTISTAS DE FÍSICO CULTURISMO EN LA UTN.

Marín Agudelo, H., Franco A., Vinaccia Alpi S., Tobón S., S. B. (2008). Trastornos Del Sueño, Salud Y Calidad De Vida: Una Perspectiva Desde La Medicina Comportamental Del Sueño. *Suma Psicológica Bogotá (Col.) Suma Psicológica Suma Psicológica*, 15(1), 217–239. Retrieved from <http://www.redalyc.org/articulo.oa?id=134212604009>

Márquez Flores, J. (2016). Introducción a Instrumentación y Señales-AMPLIFICADORES OPERACIONALES. *Procede Del Padre y Del Hijo*, 1–51. <https://doi.org/10.3726/978-3-0352-0094-2/1>

Martin Aldana, H. S., Calderon Rivas, J. D., & Vargas Hidalgo, J. M. (2018). Big data , el futuro de las predicciones certeras. *Revista Avenir*, 2(2), 10–16.

Merino Monge, M. (2015). Procesamiento y caracterización de bioseñales para su uso en interfaces de control y afectividad, (June). <https://doi.org/10.13140/RG.2.1.2088.8407>

Meza, L., Llamosa, L., & Ceballos, S. (2007). Diseño de procedimientos para la calibración de pulsioxímetros. *Scientia et Technica Año XIII. Universidad Tecnológica de Pereira.*, (37), 491–496.

Moreno, L. (2015). Diseño e Implementación de un Dispositivo para la Detección de la Oxigenación en Sangre, 107. Retrieved from <https://bit.ly/2xtPecI>

Mostafa, S. S., Mendonça, F., Morgado-Dias, F., & Ravelo-García, A. (2017). SpO2 based sleep apnea detection using deep learning. *INES 2017 - IEEE 21st International Conference on Intelligent Engineering Systems, Proceedings, 2017–Janua*, 91–96.

<https://doi.org/10.1109/INES.2017.8118534>

Noguerol, M., & Seco, A. (2017). *Pulsioximetría. Upeu*. Retrieved from www.fisterra.com

Pallás Areny, R. (1993). *Adquisición y distribución de señales*. Marcombo Boixareu. Retrieved from

[https://books.google.com.mx/books?id=sYavwLA12k4C&pg=PA27&dq=Sensores+analógicos+y+digitales&hl=es&ei=51v9TJzvHI-qsAOc4vGBBQ&sa=X&oi=book_result&ct=result#v=onepage&q=Sensores analógicos y digitales&f=false](https://books.google.com.mx/books?id=sYavwLA12k4C&pg=PA27&dq=Sensores+analógicos+y+digitales&hl=es&ei=51v9TJzvHI-qsAOc4vGBBQ&sa=X&oi=book_result&ct=result#v=onepage&q=Sensores+analógicos+y+digitales&f=false)

Pérez García, E., Hernández Echevarría, O., Rodríguez Martín, Y., & Gómez González, J. (2016). *Obtención, interpretación y usos del electroretinograma multifocal Obtaining, interpreting and using the multifocal electroretinogram. Revista Cubana de Oftalmología* (Vol. 29). Retrieved from <http://scielo.sld.cu>

Portilla, L. G. (2018). DETECTOR DE ALCOHOLEMIA PARA CONDUCTORES QUE ANALIZA VARIABLES FACIALES Y AMBIENTALES DEL AUTOMÓVIL MEDIANTE EL APRENDIZAJE AUTOMÁTICO SUPERVISADO PARA LA REDUCCIÓN DE ACCIDENTES DE TRÁNSITO. <https://doi.org/1541-8561>

Ramirez Silva, T. A. (2005). *El quinto paradigma. Gestión de la inteligencia colectiva*. Retrieved from <http://pedrobeltrancanessa-biblioteca.weebly.com/uploads/1/2/4/0/12405072/sisinteli.pdf>

Rengifo García, E. (2013). *Computaciòn en la nube. La propiedad Inmaterial* (Vol. 1).

Ruíz, F., & González, M. (2014). Procesos de Ingeniería del Software. Retrieved from <http://www.ctr.unican.es/ asignaturas/is1/is1-t02-trans.pdf>

Sáinz Menéndez, B. (2006). *ALTERACIONES DEL EQUILIBRIO ACIDO BÁSICO*. Retrieved from <http://scielo.sld.cu/pdf/cir/v45n1/cir11106.pdf>

Salazar, R. (2017). Redes Neuronales. *Historia*, 55. Retrieved from https://hugepdf.com/downloadFile/redes-neuronales-4_pdf

- Selvaraj, N., & Narasimhan, R. (2013). Detection of sleep apnea on a per-second basis using respiratory signals. In *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS* (pp. 2124–2127). IEEE. <https://doi.org/10.1109/EMBC.2013.6609953>
- Vallejos, S. (2006). *Universidad Nacional del Nordeste Facultad de Ciencias Exactas, Naturales y Agrimensura*. Retrieved from http://exa.unne.edu.ar/informatica/SO/Mineria_Datos_Vallejos.pdf
- Vázquez-Moctezuma, S. E. (2015). Tecnologías de almacenamiento de información en el ambiente digital. *E-Ciencias de La Información*, 5(2), 1. <https://doi.org/10.15517/eci.v5i2.19762>
- Villamar, L., Rabadán, A., Martínez Pérez, L., & Garay, L. (2008). Diseño e Implementación de un Electrogastrógrafo para uso Educativo. Retrieved from https://www.researchgate.net/publication/309430390_Disenio_e_Implementacion_de_un_Electrogastrografo_para_uso_Educativo
- Watanabe, K., Watanabe, T., Watanabe, H., Ando, H., Ishikawa, T., & Kobayashi, K. (2005). Noninvasive measurement of heartbeat, respiration, snoring and body movements of a subject in bed via a pneumatic method. *IEEE Transactions on Biomedical Engineering*, 52(12), 2100–2107. <https://doi.org/10.1109/TBME.2005.857637>
- Zhang, Z., & Yang, G. Z. (2015). Monitoring cardio-respiratory and posture movements during sleep: What can be achieved by a single motion sensor. *2015 IEEE 12th International Conference on Wearable and Implantable Body Sensor Networks, BSN 2015*. <https://doi.org/10.1109/BSN.2015.7299409>

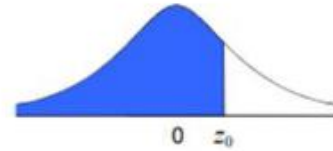
ANEXOS

ANEXO 1. Tabla de nivel de confianza de encuesta (distribución normal)

μ = Media

σ = Desviación típica

$$P(z \leq z_0) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{z_0} e^{-\frac{z^2}{2}} dz$$



Tipificación: $z_0 = \frac{x - \mu}{\sigma}$

z_0	0,00	0,01	0,02	0,03	0,04	0,05	0,06	0,07	0,08	0,09	z_0
0,0	0,5000	0,5040	0,5080	0,5120	0,5160	0,5199	0,5239	0,5279	0,5319	0,5359	0,0
0,1	0,5398	0,5438	0,5478	0,5517	0,5557	0,5596	0,5636	0,5675	0,5714	0,5753	0,1
0,2	0,5793	0,5832	0,5871	0,5910	0,5948	0,5987	0,6026	0,6064	0,6103	0,6141	0,2
0,3	0,6179	0,6217	0,6255	0,6293	0,6331	0,6368	0,6406	0,6443	0,6480	0,6517	0,3
0,4	0,6554	0,6591	0,6628	0,6664	0,6700	0,6736	0,6772	0,6808	0,6844	0,6879	0,4
0,5	0,6915	0,6950	0,6985	0,7019	0,7054	0,7088	0,7123	0,7157	0,7190	0,7224	0,5
0,6	0,7257	0,7291	0,7324	0,7357	0,7389	0,7422	0,7454	0,7486	0,7517	0,7549	0,6
0,7	0,7580	0,7611	0,7642	0,7673	0,7704	0,7734	0,7764	0,7794	0,7823	0,7852	0,7
0,8	0,7881	0,7910	0,7939	0,7967	0,7995	0,8023	0,8051	0,8078	0,8106	0,8133	0,8
0,9	0,8159	0,8186	0,8212	0,8238	0,8264	0,8289	0,8315	0,8340	0,8365	0,8389	0,9
1,0	0,8413	0,8438	0,8461	0,8485	0,8508	0,8531	0,8554	0,8577	0,8599	0,8621	1,0
1,1	0,8643	0,8665	0,8686	0,8708	0,8729	0,8749	0,8770	0,8790	0,8810	0,8830	1,1
1,2	0,8849	0,8869	0,8888	0,8907	0,8925	0,8944	0,8962	0,8980	0,8997	0,9015	1,2
1,3	0,9032	0,9049	0,9066	0,9082	0,9099	0,9115	0,9131	0,9147	0,9162	0,9177	1,3
1,4	0,9192	0,9207	0,9222	0,9236	0,9251	0,9265	0,9279	0,9292	0,9306	0,9319	1,4
1,5	0,9332	0,9345	0,9357	0,9370	0,9382	0,9394	0,9406	0,9418	0,9429	0,9441	1,5
1,6	0,9452	0,9463	0,9474	0,9484	0,9495	0,9505	0,9515	0,9525	0,9535	0,9545	1,6
1,7	0,9554	0,9564	0,9573	0,9582	0,9591	0,9599	0,9608	0,9616	0,9625	0,9633	1,7
1,8	0,9641	0,9649	0,9656	0,9664	0,9671	0,9678	0,9686	0,9693	0,9699	0,9706	1,8
1,9	0,9713	0,9719	0,9726	0,9732	0,9738	0,9744	0,9750	0,9756	0,9761	0,9767	1,9
2,0	0,9772	0,9778	0,9783	0,9788	0,9793	0,9798	0,9803	0,9808	0,9812	0,9817	2,0
2,1	0,9821	0,9826	0,9830	0,9834	0,9838	0,9842	0,9846	0,9850	0,9854	0,9857	2,1
2,2	0,9861	0,9864	0,9868	0,9871	0,9875	0,9878	0,9881	0,9884	0,9887	0,9890	2,2
2,3	0,9893	0,9896	0,9898	0,9901	0,9904	0,9906	0,9909	0,9911	0,9913	0,9916	2,3
2,4	0,9918	0,9920	0,9922	0,9925	0,9927	0,9929	0,9931	0,9932	0,9934	0,9936	2,4
2,5	0,9938	0,9940	0,9941	0,9943	0,9945	0,9946	0,9948	0,9949	0,9951	0,9952	2,5
2,6	0,9953	0,9955	0,9956	0,9957	0,9959	0,9960	0,9961	0,9962	0,9963	0,9964	2,6
2,7	0,9965	0,9966	0,9967	0,9968	0,9969	0,9970	0,9971	0,9972	0,9973	0,9974	2,7
2,8	0,9974	0,9975	0,9976	0,9977	0,9977	0,9978	0,9979	0,9979	0,9980	0,9981	2,8
2,9	0,9981	0,9982	0,9982	0,9983	0,9984	0,9984	0,9985	0,9985	0,9986	0,9986	2,9
3,0	0,99865	0,99869	0,99874	0,99878	0,99882	0,99886	0,99889	0,99893	0,99896	0,99900	3,0
3,1	0,99903	0,99906	0,99910	0,99913	0,99916	0,99918	0,99921	0,99924	0,99926	0,99929	3,1
3,2	0,99931	0,99934	0,99936	0,99938	0,99940	0,99942	0,99944	0,99946	0,99948	0,99950	3,2
3,3	0,99952	0,99953	0,99955	0,99957	0,99958	0,99960	0,99961	0,99962	0,99964	0,99965	3,3
3,4	0,99966	0,99968	0,99969	0,99970	0,99971	0,99972	0,99973	0,99974	0,99975	0,99976	3,4
3,5	0,99977	0,99978	0,99978	0,99979	0,99980	0,99981	0,99981	0,99982	0,99983	0,99983	3,5
3,6	0,99984	0,99985	0,99985	0,99986	0,99986	0,99987	0,99987	0,99988	0,99988	0,99989	3,6
3,7	0,99989	0,99990	0,99990	0,99990	0,99991	0,99991	0,99992	0,99992	0,99992	0,99992	3,7
3,8	0,99993	0,99993	0,99993	0,99994	0,99994	0,99994	0,99994	0,99995	0,99995	0,99995	3,8
3,9	0,99995	0,99995	0,99996	0,99996	0,99996	0,99996	0,99996	0,99996	0,99997	0,99997	3,9

$1-\alpha$	90%	92%	94%	95%	96%	97%	98%	99%
α	10%	8%	6%	5%	4%	3%	2%	1%
$z_{\alpha/2}$	1,645	1,751	1,881	1,960	2,054	2,170	2,326	2,576
z_{α}	1,282	1,405	1,555	1,645	1,751	1,881	2,054	2,326

Siendo:
 $1-\alpha$ = Nivel de confianza
 α = Nivel de significación

ANEXO 2. Formato de encuesta para requerimientos de usuarios.



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERIA EN CIENCIAS
APLICADAS CIERCOM



Esta encuesta va dirigida a las personas en general con la finalidad de levantar información sobre los requerimientos del usuario del sistema de representación de bioseñales de la apnea.

1. ¿Estaría usted de acuerdo utilizar durante su descanso nocturno un dispositivo capaz de detectar trastorno del sueño-apnea con la finalidad de medir posibles problemas durante su etapa del sueño?

- a. Si
- b. No

2. Para el monitoreo de sus bioseñales ¿Dónde preferiría ser evaluado?

- a. Hogar
- b. Laboratorio u hospital
- c. Otros:.....

3. ¿Le molestaría tomarse un pequeño tiempo para colocarse el prototipo de manera cómoda y adecuada antes de descansar en las noches?

- a. Si
- b. No

4. Para su comodidad ¿cómo le gustaría el diseño de los sensores de monitoreo?

Medidor de respiración		Medidor de ritmo cardiaco	
a. Diseño intrusivo a la nariz		a. Diseño ajustable al dedo	
b. Mascarilla		b. Diseño ajustable a la mano(guante)	
c. Otros:.....		c. Otros:.....	

5. ¿Cómo le gustaría recibir el resultado de su diagnóstico?

a. Mensaje de texto

b. Correo electrónico

d. Otros:.....

6. ¿Le gustaría que su información recogida en el periodo del sueño, sea representada a través de gráficos en una pantalla por computador?

a. Si

b. No

ANEXO 3. Programación script ide-arduino para la lectura de sensor de oximetría (spo2)

```
void loop() {
for(CICLOS=0; CICLOS<=100; CICLOS++){
delayMicroseconds(980);           //delay en microsegundos
////////////////////////////////////Lectura del pin analógico del sensor infrarojo //////////////////////////////////////
digitalWrite(A, HIGH);
digitalWrite(B, LOW);
digitalWrite(C, LOW);
digitalWrite(D, LOW);
Entrada[1] = analogRead(A0);       //Asigna valor obtenido a arreglo
SENSOR_VALOR_IR = Entrada[1];     //Asigna valor obtenido a variable
delayMicroseconds(1000);         // Espera para estabilización de señal
////////////////////////////////////Lectura del pin analógico del sensor rojo //////////////////////////////////////
digitalWrite(A, LOW);
digitalWrite(B, HIGH);
digitalWrite(C, LOW);
digitalWrite(D, LOW);
Entrada[2] = analogRead(A0);       //Asigna valor obtenido a arreglo
SENSOR_VALOR_R = Entrada[2];     //Asigna valor obtenido a variable
delayMicroseconds(20);           // delay en microsegundos
}
CICLOS =0;                        // Reset de ciclos
NUM_MUESTRAS=NUM_MUESTRAS+1;     //contador de muestras cada 200
miliseg.
if (NUM_MUESTRAS<=35){
```

```

MEDIA=(MEDIA*(NUM_MUESTRAS-1)+SENSOR_VALOR_R)/NUM_MUESTRAS;

// Durante los primeros 3 segundos sólo se ha calculado la media de la señal roja

// Si han pasado los 3 primeros segundos, empezamos a buscar los cruces

if(NUM_MUESTRAS>15){

if(SENSOR_VALOR_R_ANT>MEDIA&&SENSOR_VALOR_R<MEDIA){ //Se detecta un
cruce

TIEMPO_INI_PULSO = millis(); // se captura el tiempo en el que se produce el cruce

if(NUM_CRUCES ==0){ // Si es el primer cruce encontrado

TIEMPO_INI_PULSO_ANT = TIEMPO_INI_PULSO;

}

if(NUM_CRUCES >= 1){ // Apartir del segundo cruce encontrado

// Se calcula el periodo entre este cruce y el anterior

PERIODO = TIEMPO_INI_PULSO-TIEMPO_INI_PULSO_ANT;

TIEMPO_INI_PULSO_ANT = TIEMPO_INI_PULSO;

// Se calcula una media entre los periodos encontrados hasta el momento

MEDIA_PERIODOS=(MEDIA_PERIODOS*(NUM_CRUCES-
1)+PERIODO)/NUM_CRUCES;

}

NUM_CRUCES =NUM_CRUCES+1;

}

//////////////////////////////////////////SAO2//////////////////////////////////////////
//////////////////////////////////////////

//Utilización del método de absorción delta para el cálculo del ratio

if(SENSOR_VALOR_R!=0&&SENSOR_VALOR_R_ANT!=0){

dA_R=(SENSOR_VALOR_R-
SENSOR_VALOR_R_ANT)*2/(SENSOR_VALOR_R+SENSOR_VALOR_R_ANT);

```

```

if(SENSOR_VALOR_IR!=0&&SENSOR_VALOR_IR_ANT!=0){

dA_IR=(SENSOR_VALOR_IR-
SENSOR_VALOR_IR_ANT)*2/(SENSOR_VALOR_IR+SENSOR_VALOR_IR_ANT);

RATIO_PROV=dA_IR/dA_R;

}

}

// Si el ratio provisional es infinito, no sirve y se usa el ratio anterior para la media
if(RATIO_PROV != INFINITY){
if(RATIO_PROV!=NAN){
RATIO=RATIO_PROV;
}
}

// Se calcula una media entre los ratios calculados hasta el momento
MEDIA_RATIO=(MEDIA_RATIO*(NUM_MUESTRAS-16)+RATIO)/(NUM_MUESTRAS-
15);

SENSOR_VALOR_R_ANT=SENSOR_VALOR_R;

SENSOR_VALOR_IR_ANT=SENSOR_VALOR_IR;

}

}

else{

// Calculamos y mostramos el valor del pulso cardiaco
PULSO_CARD = (60000/MEDIA_PERIODOS);

SAO2=115-(30*MEDIA_RATIO);

Serial.print("Pulso cardiaco =");

Serial.print(PULSO_CARD);

```

```
Serial.print(",");  
Serial.print("Saturacion de oxigeno=");  
Serial.println(SAO2);  
  
NUM_MUESTRAS=0;  
MEDIA=0;  
SENSOR_VALOR_R_ANT=0;  
SENSOR_VALOR_IR_ANT=0;  
TIEMPO_INI_PULSO=0;  
TIEMPO_INI_PULSO_ANT=0;  
NUM_CRUCES=0;  
MEDIA_PERIODOS=0;  
PERIODO=0;  
MEDIA_RATIO=0;  
RATIO=0;  
RATIO_PROV=0;  
}  
}
```

ANEXO 4. Programación script ide-arduino para la lectura de sensor acelerómetro (posición).

```
////////////////////////////////////DECLARACION VARIABLES////////////////////////////////////
```

```
float Entrada[15];
```

```
float valsx[N];
```

```
float valsy[N];
```

```
float valsz[N];
```

```
float filteredx;
```

```
float filteredy;
```

```
float filteredz;
```

```
int posicion;
```

```
////////////////////////////////////
```

```
void setup() {
```

```
  Serial.begin(9600);
```

```
  pinMode(A,OUTPUT);// salida conectado a A/S0 pin 11 multiplexor
```

```
  pinMode(B,OUTPUT);// salida conectado a B/S1 pin 10 multiplexor
```

```
  pinMode(C,OUTPUT);// salida conectado a C/S2 pin 09 multiplexor
```

```
  pinMode(D,OUTPUT); // salida conectado a D/S3 pin 08 multiplexor
```

```
  pinMode(A0,INPUT); //Salida analógica
```

```
}
```

```
void loop() {
```

```
  digitalWrite(A,HIGH);//Encendido de PIN
```

```
  digitalWrite(B,HIGH); //Encendido de PIN
```

```
  digitalWrite(C,LOW); //Apagado de PIN
```



```

digitalWrite(D,LOW); //Apagado de PIN

Entrada[3] = analogRead(A0); //Guarda valor en arreglo

//////////////////////////////////FILTRO AVERAGE MOVING//////////////////////////////////

for(int i =N-1; i>0; i--){

    valsx[i]=valsx[i-1];

}

valsx[0]=Entrada[3];

long sumx=0;

for(int i=0;i<N;i++){

    sumx=sumx+valsx[i];

}

    filteredx=sumx/N;

//////////////////////////////////

// Lectura de X4/Y4

digitalWrite(A,LOW); //Apagado de PIN

digitalWrite(B,LOW); //Apagado de PIN

digitalWrite(C,HIGH); // Encendido de PIN

digitalWrite(D,LOW); //Apagado de PIN

Entrada[4] = analogRead(A0); // Guarda valor en arreglo

//////////////////////////////////FILTRO AVERAGE MOVING//////////////////////////////////

for(int i =N-1; i>0; i--){

    valsy[i]=valsy[i-1];

}

    valsy[0]=Entrada[4];

    long sumy=0;

```

```

for(int i=0;i<N;i++){
    sumy=sumy+valsy[i];
}
filteredy=sumy/N;

/////////////////////////////////////////////////////////////////

// Lectura de X5/Y5

digitalWrite(A,HIGH); // Encendido de PIN
digitalWrite(B,LOW); // Apagado de PIN
digitalWrite(C,HIGH); // Encendido de PIN
digitalWrite(D,LOW); // Apagado de PIN

Entrada[5] = analogRead(A0); // Guarda valor en arreglo

/////////////////////////////////////////////////////////////////

/////////////////////////////////////////////////////////////////
//FILTRO AVERAGE MOVING
/////////////////////////////////////////////////////////////////

for(int i =N-1; i>0; i--){
    valsz[i]=valsz[i-1];
}

valsz[0]=Entrada[5];

long sumz=0;

for(int i=0;i<N;i++){
    sumz=sumz+valsz[i];
}

filteredz=sumz/N;

/////////////////////////////////////////////////////////////////

if(filteredx>=460&&filteredx<=550){ //Sentencia de comparación de valores en eje x

```

```

if(filteredz>=570&&filteredz<=660){ // Sentencia de comparación de valores en eje z
    //Serial.println("BOCA ARRIBA");
    posicion=1; //Asignar valor a POSICION
}
if(filteredz<460){ // Sentencia de comparación de valores en eje z
    //Serial.println("BOCA ABAJO");
    posicion=4; //Asignar valor a POSICION
}
}

if(filteredz>=500&&filteredz<=550){ // Sentencia de comparación de valores en eje z
    if(filteredx>=425&&filteredx<=440){ //Sentencia de comparación de valores en eje x
        //Serial.println("IZQUIERDA");
        posicion=3; //Asignar valor a POSICION
    }
    if(filteredx>630&&filteredx<640){ //Sentencia de comparación de valores en eje x
        //Serial.println("DERECHA");
        posicion=2; //Asignar valor a POSICION
    }
}
}
}

```

ANEXO 3. Programación script ide-arduino para la lectura de sensor respiración.

```
////////////////////////////////////DECLARACION VARIABLES////////////////////////////////////  
  
float Entrada[15];  
  
float valresp[N];  
  
float filteredresp;  
  
float PT;  
  
int cont;  
  
int a;  
  
int b;  
  
float ant=0;  
  
float total;  
  
////////////////////////////////////  
  
void setup() {  
  Serial.begin(9600);  
  
  pinMode(A,OUTPUT);// salida conectado a A/S0 pin 11 multiplexor  
  pinMode(B,OUTPUT);// salida conectado a B/S1 pin 10 multiplexor  
  pinMode(C,OUTPUT);// salida conectado a C/S2 pin 09 multiplexor  
  pinMode(D,OUTPUT); // salida conectado a D/S0 pin 08 multiplexor  
  pinMode(A0,INPUT); //Salida análoga  
}  
  
void loop() {  
  for(;b<1000;b++){  
    digitalWrite(A, HIGH); // Encendido de PIN  
    digitalWrite(B, LOW); // Apagado de PIN
```

```
digitalWrite(C, HIGH); // Encendido de PIN
digitalWrite(D, HIGH); // Encendido de PIN
Entrada[13] = analogRead(A0); //Asignación de valor a arreglo
```

```
////////////////////////////////////FILTRO AVERAGE MOVING////////////////////////////////////
```

```
for(int i =N-1; i>0; i--){
    valresp[i]=valresp[i-1];
}
valresp[0]=Entrada[13];
long sumresp=0;
for(int i=0;i<N;i++){
    sumresp=sumresp+valresp[i];
}
filteredresp=sumresp/N;
```

```
////////////////////////////////////
```

```
if(filteredresp<=380&&ant>380){ //Sentencia de cruce
    a=a+1; // contador
}
ant=filteredresp; //guarda valor anterior en variable
delay(10);
}
PT=PT+a; //Sumatoria de valores
cont=cont+1; //contador
```

```
if(cont==4){ //condicional para imprimir cada minuto
total=PT*1.5;
Serial.print("RESPUESTA=");
Serial.println(total); //imprime respuesta
cont=0;// reinicio de variables
PT=0; // reinicio de variables
}
a=0; // reinicio de variables
b=0; // reinicio de variables
}
```

ANEXO 4. Programación script ide-arduino para la lectura de sensor ronquido.

```
////////////////////////////////////DECLARACION VARIABLES////////////////////////////////////  
  
float Entrada[15];  
  
float valsno1[N];  
  
float filteredsno1;  
  
float PT;  
  
int cont;  
  
int a;  
  
int b;  
  
float ant=0;  
  
float total;  
  
////////////////////////////////////  
  
void setup() {  
  Serial.begin(9600);  
  
  pinMode(A,OUTPUT);// salida conectado a A/S0 pin 11 multiplexor  
  pinMode(B,OUTPUT);// salida conectado a B/S1 pin 10 multiplexor  
  pinMode(C,OUTPUT);// salida conectado a C/S2 pin 09 multiplexor  
  pinMode(D,OUTPUT);// salida conectado a D/S3 pin 08 multiplexor  
  pinMode(A0,INPUT); //Salida analoga  
}  
  
void loop() {  
  for(;b<1000;b++){  
    digitalWrite(A, LOW); // Apagado de PIN  
    digitalWrite(B, LOW); // Apagado de PIN
```

```

digitalWrite(C, LOW); // Apagado de PIN
digitalWrite(D, LOW); // Apagado de PIN
Entrada[0] = analogRead(A0); //Asignación de valor a arreglo

////////////////////////////////////FILTRO AVERAGE MOVING////////////////////////////////////

for(int i =N-1; i>0; i--){
    valsno1[i]=valsno1[i-1];
}
valsno1[0]=Entrada[0];
long sumsno1=0;
for(int i=0;i<N;i++){
    sumsno1=sumsno1+valsno1[i];
}
filtered1=sumsno1/N;

////////////////////////////////////

if(filtered1>=290&&ant<290||filtered1<=250&&ant>250){ //sentencia para asignar
conteo
    a=a+1; //contador
}
ant=filtered1;// guarda valor anterior
delay(10);
}
PT=PT+a; //sumatoria total
cont=cont+1;//contador

```



```
if(cont==4){//sentencia para imprimir cada minuto
total=(PT)*1.5;
Serial.print("=====si ronca");
Serial.println(ant);
}
else{
  Serial.print("=====no ronca");
  Serial.println(ant);
}
cont=0;
PT=0;
}
a=0;
}
```

ANEXO 5. Señal obtenida por serial plotter de arduino del prototipo de oximetría.

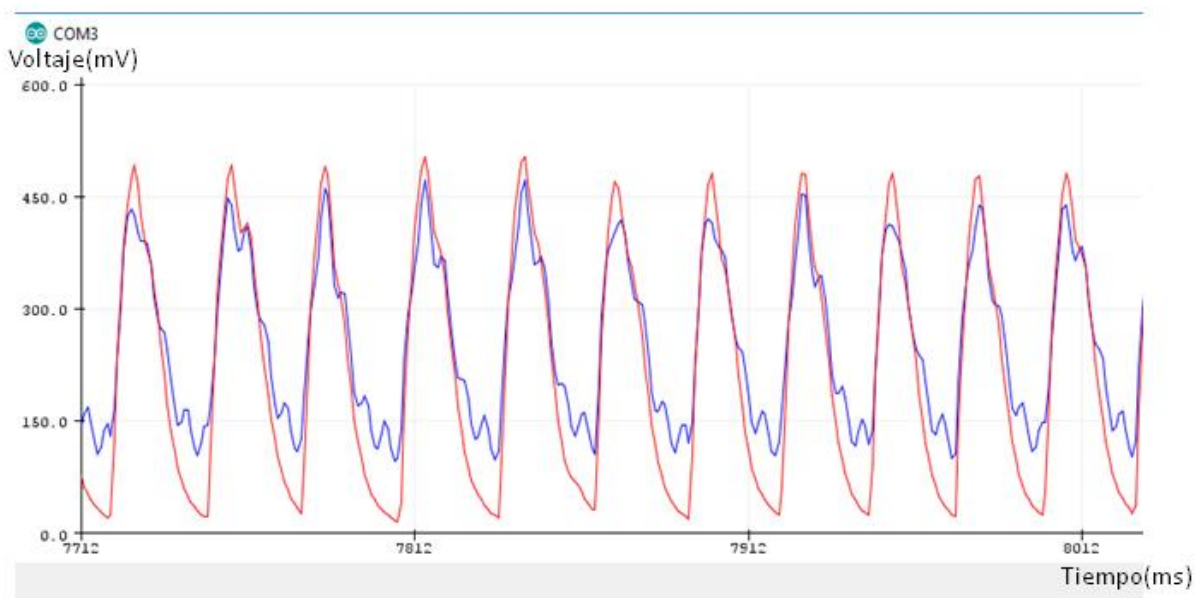


Figura 60: Señal final obtenida por el sistema

Fuente: Elaboración propia

ANEXO 6. Manual de usuario funcionamiento del sistema inteligente.



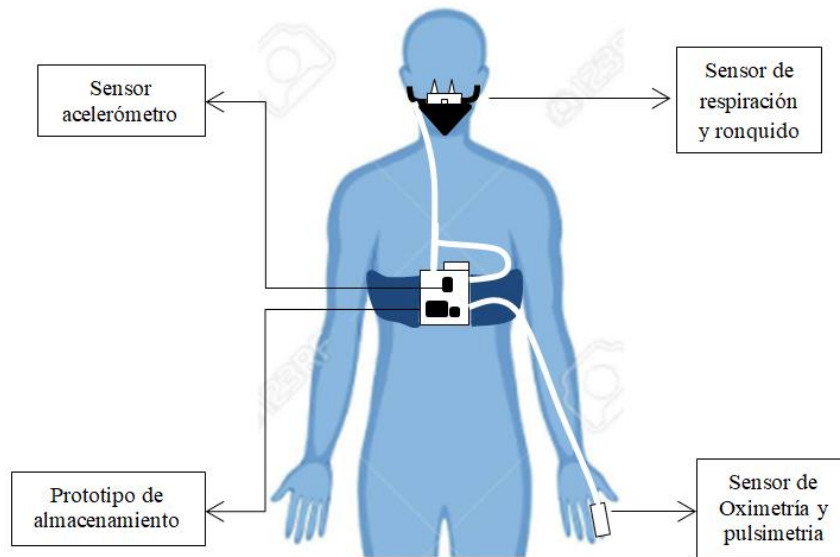
UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERIA EN CIENCIAS APLICADAS CIERCOM

MANUAL DE USUARIO

Para el correcto funcionamiento del sistema es necesario tomar en cuenta ciertos pasos necesarios los cuales se detallan a continuación.

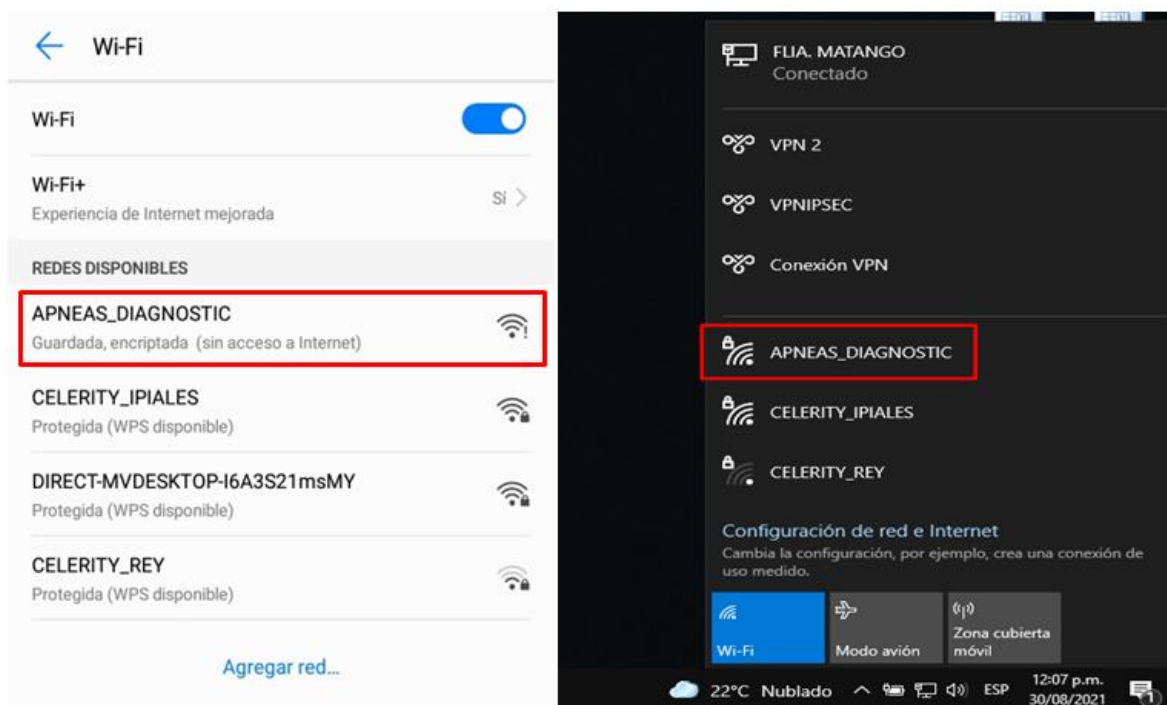
El sistema posee un sistema eléctrico interno recargable mediante el cual puede ser más fácil su utilización y movilidad ya que no sería obligatorio utilizar conexión directa a una fuente de energía. Sin embargo, también existe la posibilidad de utilizar un cable de conexión eléctrica mientras se utiliza el sistema.

1. Asegúrese que el sistema se encuentre correctamente ubicado para ello es necesario guiarse de la figura que se muestra a continuación.

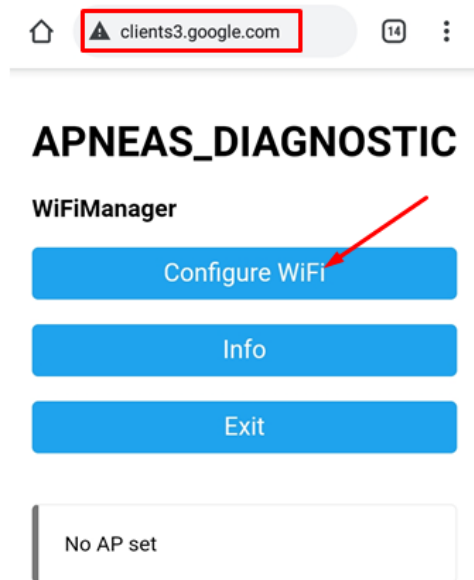


2. Es importante encender el sistema cuando esté totalmente seguro que el usuario va descansar (dormir), ya que de no ser así el sistema arrojará resultados poco acertados.

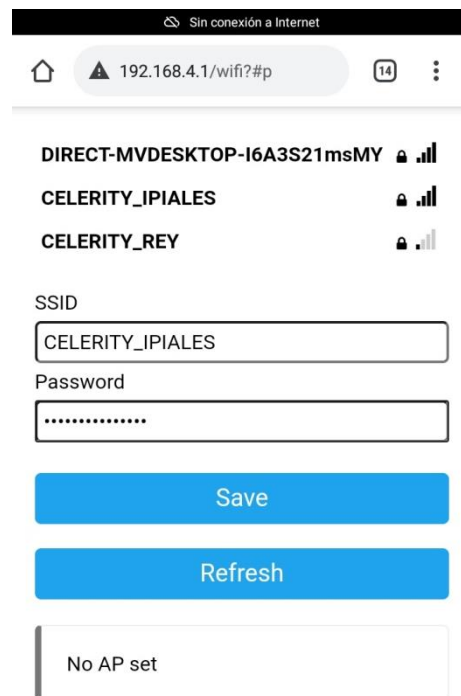
3. Luego de encender, en el caso de que el dispositivo sea utilizado por primera vez en una determinada red WiFi, este creará una red temporal a la cual accederemos con las credenciales por defecto. Usuario: APNEAS_DIAGNOSTIC, Contraseña: 12345678



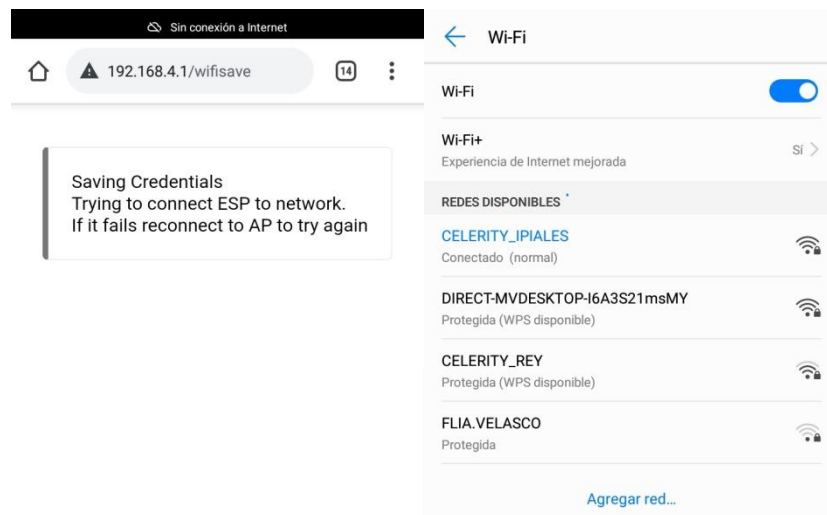
4. luego de conectarnos automáticamente el dispositivo será re direccionado a la página creada por el sistema. En caso de no ser re direccionado se puede hacerlo manualmente a través de un navegador, en el cual nos dirigiremos al siguiente enlace: clients3.google.com y nos aparecerá la siguiente página:



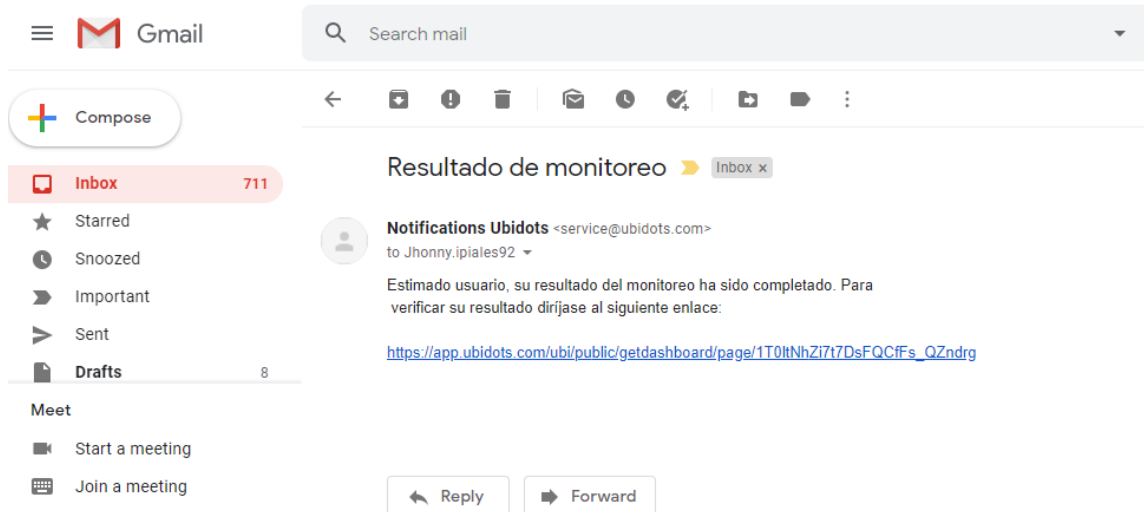
5. A continuación, nos dirigimos a “Configure WiFi”, en el cual vamos a dirigirnos a una red inalámbrica con acceso a internet y brindamos las credenciales necesarias para conectarse a esta nueva red.



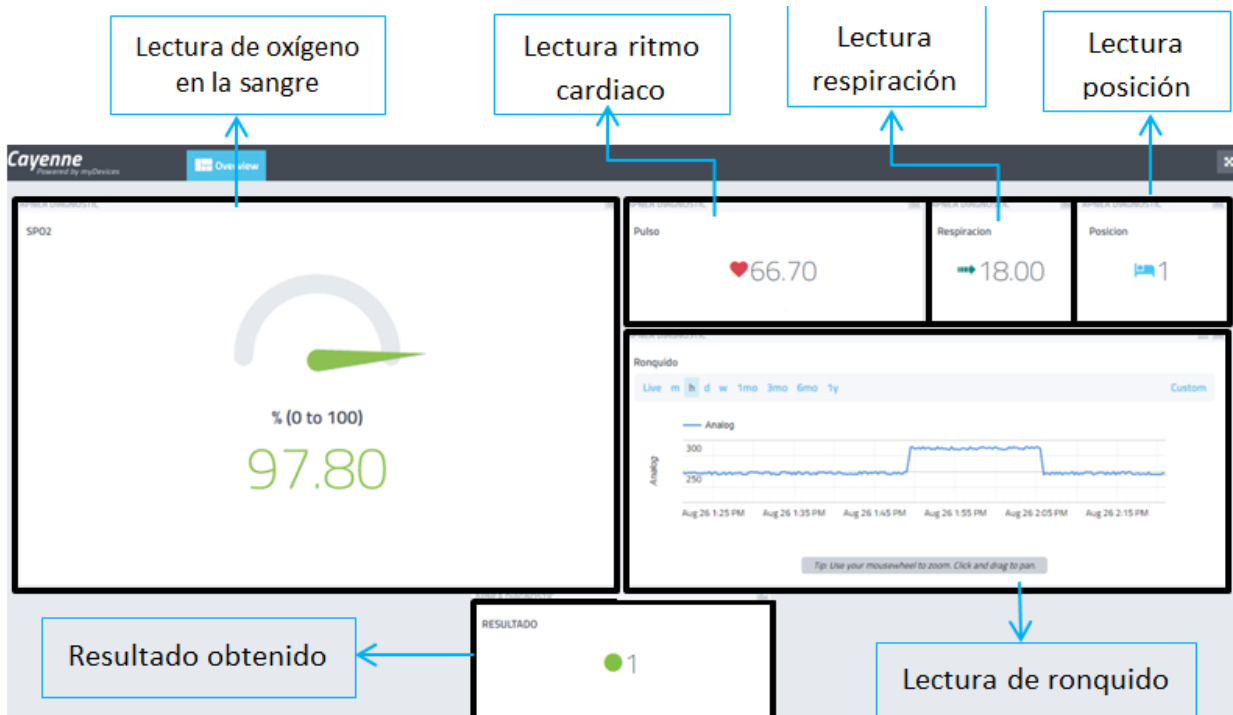
6. Con lo ya realizado el sistema ya tiene conexión a internet, por lo cual la red antes creada por el mismo desaparece ya que ya no es necesaria. A continuación se muestra el mensaje de confirmación en cuanto a las credenciales brindadas al sistema, además se observa que la red a desaparecido ya que su objetivo ha sido completado.



7. Es necesario tomara en consideración que el sistema debe monitorear al usuario al menos por una hora durante el sueño del mismo, de no ser así el sistema no tendrá datos suficientes para brindar un correcto pronóstico. Una vez transcurrido el tiempo necesario de monitoreo el sistema enviara los resultados a la plataforma. Además, se enviara un correo electrónico al usuario en el cual consta el enlace que lo dirigirá a la página de visualización de datos.



5. El usuario recibirá un correo electrónico con un enlace, mismo que dirige a la página, en la cual se observa los valores biomédicos los mismo que se representan de la siguiente manera.



ANEXO 7. Especificaciones técnicas de Node MCU

Tabla 26. Especificaciones técnicas de Node MCU



Voltaje de operación	5V
Wi-Fi	Direct (P2P), soft-AP.
Current consumption:	10uA~170mA.
Flash memory attachable:	16MB max (512K normal).
Processor:	Tensilica L106 32-bit.
Processor speed:	80~160MHz.
RAM:	32K + 80K.
GPIOs:	17 (multiplexed with other functions).
Analog to Digital:	1 input with 1024 step resolution.
Maximum concurrent TCP connections:	5

Fuente: Adaptado de (Handsontec, 2017)

7.1 Current Consumption

The following current consumption is based on 3.3V supply, and 25°C ambient, using internal regulators. Measurements are done at antenna port without SAW filter. All the transmitter's measurements are based on 90% duty cycle, continuous transmit mode.

Mode	Min	Typ	Max	Unit
Transmit 802.11b, CCK 1Mbps, $P_{OUT}=+19.5\text{dBm}$		215		mA
Transmit 802.11b, CCK 11Mbps, $P_{OUT}=+18.5\text{dBm}$		197		mA
Transmit 802.11g, OFDM 54Mbps, $P_{OUT}=+16\text{dBm}$		145		mA
Transmit 802.11n, MCS7, $P_{OUT}=+14\text{dBm}$		135		mA
Receive 802.11b, packet length=1024 byte, -80dBm		60		mA
Receive 802.11g, packet length=1024 byte, -70dBm		60		mA
Receive 802.11n, packet length=1024 byte, -65dBm		62		mA
Standby		0.9		mA
Deep sleep		10		uA
Power save mode DTIM 1		1.2		mA
Power save mode DTIM 3		0.86		mA
Total shutdown		0.5		uA

7.2 RF Performance

The following are measured under room temperature conditions with 3.3V and 1.1V power supplies.

Description	Min	Typical	Max	Unit
Input frequency	2412		2484	MHz
Input impedance		50		Ω
Input reflection			-10	dB
Output power of PA for 72.2Mbps	14	15	16	dBm
Output power of PA for 11b mode	17.5	18.5	19.5	dBm
Sensitivity				
CCK, 1Mbps		-98		dBm
CCK, 11Mbps		-91		dBm
6Mbps (1/2 BPSK)		-93		dBm
54Mbps (3/4 64-QAM)		-75		dBm
HT20, MCS7 (65Mbps, 72.2Mbps)		-71		dBm
Adjacent Channel Rejection				
OFDM, 6Mbps		37		dB
OFDM, 54Mbps		21		dB
HT20, MCS0		37		dB
HT20, MCS7		20		dB

ANEXO 8. Características de sensor SPO2

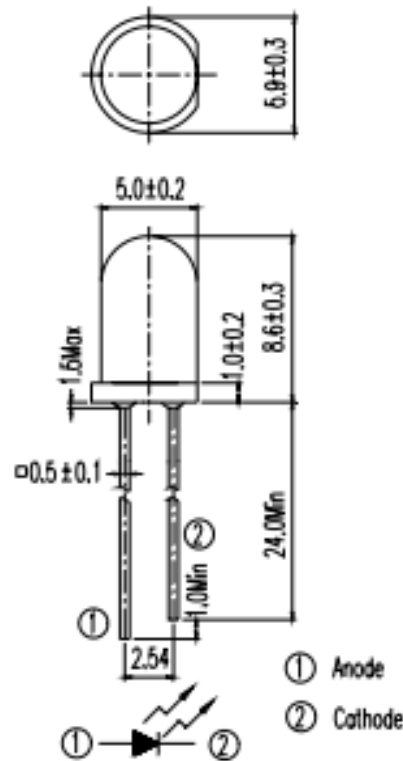
Tabla 27. Características de sensor SPO2



Voltaje de operación	3.3V
Frecuencia	500Hz
emisión máxima infrarroja	950nm
emisión máxima-roja	660nm
Detector	2 KB
Longitud de onda de max. sensibilidad	920nm
Rango de sensibilidad	400nm-1100nm
Área radiante sensibilidad	1.3x1.3 (mm)
Capacidad típica	900-1155 <u>mAh</u>

Fuente: Obtenido de (Moreno, 2015)

Package Dimensions



- Notes:** 1.All dimensions are in millimeters
 2.Tolerances unless dimensions $\pm 0.25\text{mm}$

Absolute Maximum Ratings (Ta=25°C)

Parameter	Symbol	Rating	Units
Continuous Forward Current	I_F	100	mA
Peak Forward Current	I_{FP}	1.0	A
Reverse Voltage	V_R	5	V
Operating Temperature	T_{opr}	-40 ~ +85	°C
Storage Temperature	T_{stg}	-40 ~ +85	°C
Soldering Temperature	T_{sol}	260	°C
Power Dissipation at(or below) 25°C Free Air Temperature	P_d	150	mW

- Notes:** *1: I_{FP} Conditions--Pulse Width $\leq 100 \mu s$ and Duty $\leq 1\%$.
 *2:Soldering time ≤ 5 seconds.

Electro-Optical Characteristics (Ta=25°C)

Parameter	Symbol	Condition	Min.	Typ.	Max.	Units
Radiant Intensity	Ee	I _F =20mA	11	12	--	mW/sr
		I _F =100mA Pulse Width ≤ 100 μs ,Duty ≤ 1%	--	45	--	
		I _F =1A Pulse Width ≤ 100 μs ,Duty ≤ 1%.	--	400	--	
Peak Wavelength	λ _p	I _F =20mA	--	940	--	nm
Spectral Bandwidth	Δλ	I _F =20mA	--	45	--	nm
Forward Voltage	V _F	I _F =20mA		1.2	1.5	V
		I _F =100mA Pulse Width ≤ 100 μs ,Duty ≤ 1%	--	1.4	1.8	
		I _F =1A Pulse Width ≤ 100 μs ,Duty ≤ 1%.	--	2.6	4.0	
Reverse Current	I _R	V _R =5V	--	--	10	μA
View Angle	2θ _{1/2}	I _F =20mA	--	40	--	deg

Typical Electro-Optical Characteristics Curves

Fig.1 Forward Current vs. Ambient Temperature

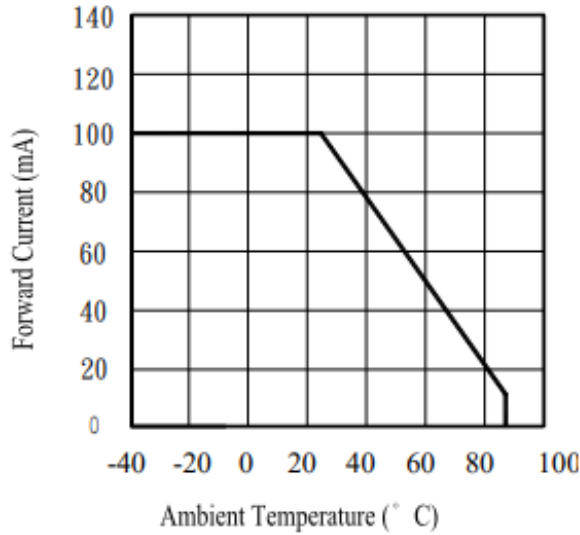


Fig.2 Spectral Distribution

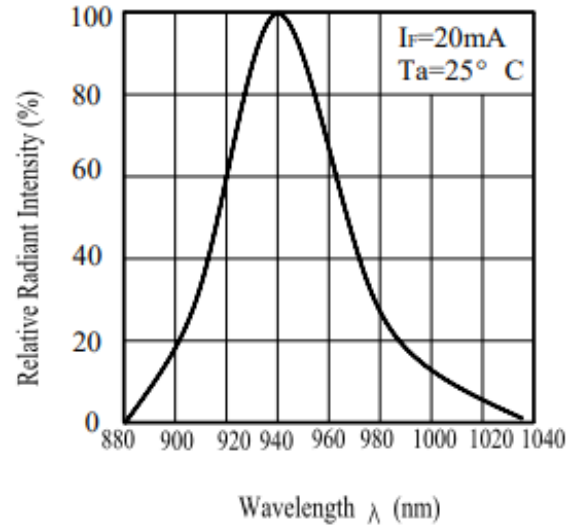


Fig.3 Peak Emission Wavelength vs. Ambient Temperature

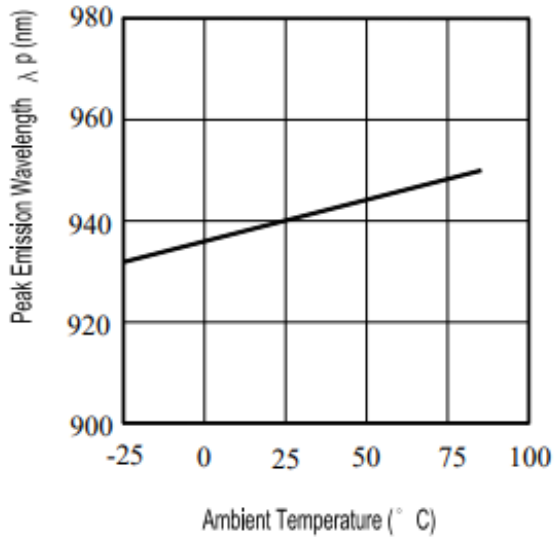
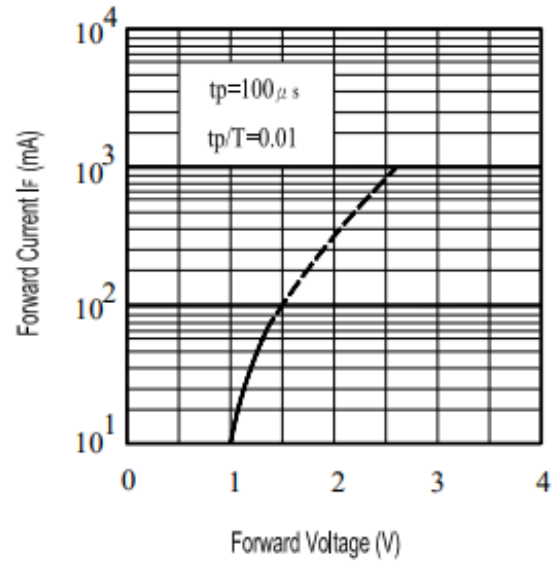
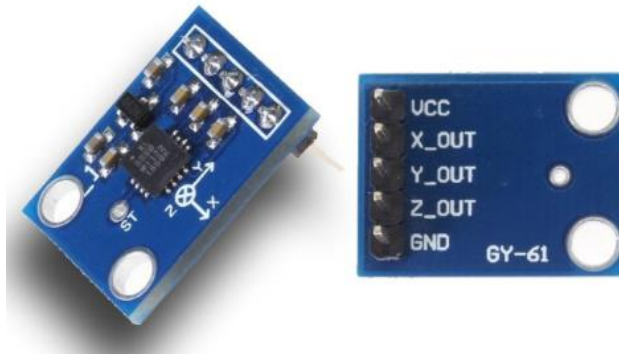


Fig.4 Forward Current vs. Forward Voltage



ANEXO 9. Características de sensor de posición del cuerpo

Tabla 28. Características de sensor de posición del cuerpo



Voltaje de operación	3.3V
Rango de rotación:	0° to 360° rotación continua
Temperatura de operación estándar °C	-40 a +85
Sensibilidad de XOUT, YOUT, ZOUT	$V_s=3V$
No linealidad	$\pm 0.3\%$
Sensor de frecuencia de resonancia	5.5kHz
Ancho de banda XOUT, YOUT	1.6kHz
Ancho de banda ZOUT	550 Hz

Fuente: Obtenido de (Libelium Comunicaciones, 2017)

SPECIFICATIONS

$T_A = 25^\circ\text{C}$, $V_S = 3\text{ V}$, $C_X = C_Y = C_Z = 0.1\ \mu\text{F}$, acceleration = 0 g, unless otherwise noted. All minimum and maximum specifications are guaranteed. Typical specifications are not guaranteed.

Table 1.

Parameter	Conditions	Min	Typ	Max	Unit
SENSOR INPUT					
Measurement Range	Each axis	± 3	± 3.6		g
Nonlinearity	% of full scale		± 0.3		%
Package Alignment Error			± 1		Degrees
Interaxis Alignment Error			± 0.1		Degrees
Cross-Axis Sensitivity ¹			± 1		%
SENSITIVITY (RATIOMETRIC)²					
Sensitivity at X_{OUT} , Y_{OUT} , Z_{OUT}	$V_S = 3\text{ V}$	270	300	330	mV/g
Sensitivity Change Due to Temperature ³	$V_S = 3\text{ V}$		± 0.01		%/ $^\circ\text{C}$
ZERO g BIAS LEVEL (RATIOMETRIC)					
0 g Voltage at X_{OUT} , Y_{OUT}	$V_S = 3\text{ V}$	1.35	1.5	1.65	V
0 g Voltage at Z_{OUT}	$V_S = 3\text{ V}$	1.2	1.5	1.8	V
0 g Offset vs. Temperature			± 1		mg/ $^\circ\text{C}$
NOISE PERFORMANCE					
Noise Density X_{OUT} , Y_{OUT}			150		$\mu\text{g}/\sqrt{\text{Hz}}$ rms
Noise Density Z_{OUT}			300		$\mu\text{g}/\sqrt{\text{Hz}}$ rms
FREQUENCY RESPONSE⁴					
Bandwidth X_{OUT} , Y_{OUT} ⁵	No external filter		1600		Hz
Bandwidth Z_{OUT} ⁵	No external filter		550		Hz
R_{FLT} Tolerance			$32 \pm 15\%$		k Ω
Sensor Resonant Frequency			5.5		kHz
SELF-TEST⁶					
Logic Input Low			+0.6		V
Logic Input High			+2.4		V
ST Actuation Current			+60		μA
Output Change at X_{OUT}	Self-Test 0 to Self-Test 1	-150	-325	-600	mV
Output Change at Y_{OUT}	Self-Test 0 to Self-Test 1	+150	+325	+600	mV
Output Change at Z_{OUT}	Self-Test 0 to Self-Test 1	+150	+550	+1000	mV
OUTPUT AMPLIFIER					
Output Swing Low	No load		0.1		V
Output Swing High	No load		2.8		V
POWER SUPPLY					
Operating Voltage Range		1.8		3.6	V
Supply Current	$V_S = 3\text{ V}$		350		μA
Turn-On Time ⁷	No external filter		1		ms
TEMPERATURE					
Operating Temperature Range		-40		+85	$^\circ\text{C}$

ANEXO 10. Características del sensor de medición de ronquido

Tabla 29. Características del sensor de medición de ronquido.

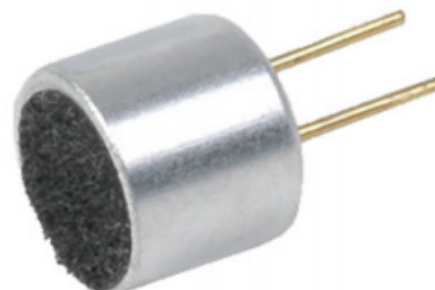


Voltaje de operación	3.3V
Atmospheric pressure	10 - 15 ps
Typical Signal Amplitude	1mV
Low frequency filter	10 Hz
High frequency filter	100 Hz
Sensibilidad	20 a 75 $\mu\text{V}/\text{mm}$
Temperatura de operación	0 - 40°C
Humedad	0 a 93%

Fuente: Obtenido de (Libelium Comunicaciones, 2017)

MODEL: CMC-5044PF-A | **DESCRIPTION:** ELECTRET CONDENSER MICROPHONE**FEATURES**

- omnidirectional
- 6.0 mm diameter
- solder pins

**SPECIFICATIONS**

parameter	conditions/description	min	typ	max	units
directivity	omnidirectional				
sensitivity (S)	f = 1 kHz, 1 Pa, 0 dB = 1 V/Pa	-47	-44	-41	dB
standard operating voltage (Vs)			2		Vdc
max operating voltage				10	Vdc
output impedance (Zout)	f = 1 kHz, 1 Pa		2.2		kΩ
sensitivity reduction (ΔS-Vs)	f = 1 kHz, 1 Pa, Vs = 2.0 ~ 1.5 Vdc		-3		dB
frequency (f)		100		20,000	Hz
current consumption (IDSS)	Vs = 2.0 Vdc, RL = 2.2 kΩ			0.5	mA
signal to noise ratio (S/N)	f = 1 kHz, 1 Pa, A-weighted		56		dBA
dimensions	Ø6.0 x 5.0				mm
material	AL				
terminal	solder pins (gold plating)				
weight				0.30	g
operating temperature		-20		70	°C
storage temperature		-20		70	°C
RoHS	yes				

Notes: 1. We use the "Pascal (Pa)" indication of sensitivity as per the recommendation of I.E.C. (International Electrotechnical Commission). The sensitivity of "Pa" will increase 20 dB compared to the "ubar" indication. Example: -60 dB (0 dB = 1 V/ubar) = -40 dB (1 V/Pa)
 2. All specifications measured at 5~35°C, humidity at 45~85%, under 86~106 kPa pressure, unless otherwise noted.

SOLDERABILITY

parameter	conditions/description	min	typ	max	units
hand soldering	for 2 ± 0.5 seconds	270	280	290	°C